diagnostics

Special Issue Reprint

Medical Data Processing and Analysis

Edited by
Wan Azani Mustafa and Hiam Alquran

www.mdpi.com/journal/diagnostics

MDPI

Medical Data Processing and Analysis

Medical Data Processing and Analysis

Editors

Wan Azani Mustafa
Hiam Alquran

MDPI • Basel • Beijing • Wuhan • Barcelona • Belgrade • Manchester • Tokyo • Cluj • Tianjin

Editors
Wan Azani Mustafa
Universiti Malaysia Per
Arau, Malaysia

Hiam Alquran
Yarmouk University
Irbid, Jordan

Editorial Office
MDPI
St. Alban-Anlage 66
4052 Basel, Switzerland

This is a reprint of articles from the Special Issue published online in the open access journal *Diagnostics* (ISSN 2075-4418) (available at: https://www.mdpi.com/journal/diagnostics/special_issues/7S600N389J).

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

LastName, A.A.; LastName, B.B.; LastName, C.C. Article Title. *Journal Name* **Year**, *Volume Number*, Page Range.

ISBN 978-3-0365-8068-5 (Hbk)
ISBN 978-3-0365-8069-2 (PDF)

Contents

About the Editors . . . . . **vii**

Wan Azani Mustafa and Hiam Alquran
Editorial on Special Issue "Medical Data Processing and Analysis"
Reprinted from: *Diagnostics* **2023**, *13*, 2081, doi:10.3390/diagnostics13122081 . . . . . **1**

Naseha Wafa Qammar, Vaiva Šiaučiūnaitė, Vytautas Zabiela, Alfonsas Vainoras and Minvydas Ragulskis
Detection of Atrial Fibrillation Episodes based on 3D Algebraic Relationships between Cardiac Intervals
Reprinted from: *Diagnostics* **2022**, *12*, 2919, doi:10.3390/diagnostics12122919 . . . . . **5**

Keijiro Nakamura, Xue Zhou, Naohiko Sahara, Yasutake Toyoda, Yoshinari Enomoto, Hidehiko Hara, et al.
Risk of Mortality Prediction Involving Time-Varying Covariates for Patients with Heart Failure Using Deep Learning
Reprinted from: *Diagnostics* **2022**, *12*, 2947, doi:10.3390/diagnostics12122947 . . . . . **25**

Abdelkader Dairi, Nabil Zerrouki, Fouzi Harrou and Ying Sun
EEG-Based Mental Tasks Recognition via a Deep Learning-Driven Anomaly Detector
Reprinted from: *Diagnostics* **2022**, *12*, 2984, doi:10.3390/diagnostics12122984 . . . . . **39**

Parvathaneni Naga Srinivasu, Jana Shafi, T Balamurali Krishna, Canavoy Narahari Sujatha, S Phani Praveen and Muhammad Fazal Ijaz
Using Recurrent Neural Networks for Predicting Type-2 Diabetes from Genomic and Tabular Data
Reprinted from: *Diagnostics* **2022**, *12*, 3067, doi:10.3390/diagnostics12123067 . . . . . **59**

Mehmet Erten, Madhav R. Acharya, Aditya P. Kamath, Niranjana Sampathila, G. Muralidhar Bairy, Emrah Aydemir, et al.
Hamlet-Pattern-Based Automated COVID-19 and Influenza Detection Model Using Protein Sequences
Reprinted from: *Diagnostics* **2022**, *12*, 3181, doi:10.3390/diagnostics12123181 . . . . . **89**

Ahmed Almulihi, Hager Saleh, Ali Mohamed Hussien, Sherif Mostafa, Shaker El-Sappagh, Khaled Alnowaiser, et al.
Ensemble Learning Based on Hybrid Deep Learning Model for Heart Disease Early Prediction
Reprinted from: *Diagnostics* **2022**, *12*, 3215, doi:10.3390/diagnostics12123215 . . . . . **105**

Ala'a Zyout, Hiam Alquran, Wan Azani Mustafa and Ali Mohammad Alqudah
Advanced Time-Frequency Methods for ECG Waves Recognition
Reprinted from: *Diagnostics* **2023**, *13*, 308, doi:10.3390/diagnostics13020308 . . . . . **123**

Yasmin Mohd Yacob, Hiam Alquran, Wan Azani Mustafa, Mohammed Alsalatie, Harsa Amylia Mat Sakim and Muhamad Safiih Lola
H. pylori Related Atrophic Gastritis Detection Using Enhanced Convolution Neural Network (CNN) Learner
Reprinted from: *Diagnostics* **2023**, *13*, 336, doi:10.3390/diagnostics13030336 . . . . . **143**

Hatim Butt, Ikramullah Khosa and Muhammad Aksam Iftikhar
Feature Transformation for Efficient Blood Glucose Prediction in Type 1 Diabetes Mellitus Patients
Reprinted from: *Diagnostics* **2023**, *13*, 340, doi:10.3390/diagnostics13030340 . . . . . **159**

Riaz Ahmad, Muhammad Awais, Nabeela Kausar and Tallha Akram
White Blood Cells Classification Using Entropy-Controlled Deep Features Optimization
Reprinted from: *Diagnostics* **2023**, *13*, 352, doi:10.3390/diagnostics13030352 . . . . . **179**

Marina Yusoff, Toto Haryanto, Heru Suhartanto, Wan Azani Mustafa, Jasni Mohamad Zain and Kusmardi Kusmardi
Accuracy Analysis of Deep Learning Methods in Breast Cancer Classification: A Structured Review
Reprinted from: *Diagnostics* **2023**, *13*, 683, doi:10.3390/diagnostics13040683 . . . . . **197**

Abdul Aziz K Abdul Hamid, Wan Imanul Aisyah Wan Mohamad Nawi, Muhamad Safiih Lola, Wan Azani Mustafa, Siti Madhihah Abdul Malik, Syerrina Zakaria, et al.
Improvement of Time Forecasting Models Using Machine Learning for Future Pandemic Applications Based on COVID-19 Data 2020–2022
Reprinted from: *Diagnostics* **2023**, *13*, 1121, doi:10.3390/diagnostics13061121 . . . . . **213**

About the Editors

Wan Azani Mustafa

Dr. Wan Azani Mustafa received a degree in biomedical electronic engineering (2013) and a PhD in mechatronic engineering (2017) from University Malaysia Perlis (UniMAP). Mustafa was a member of the Board of Engineers Malaysia (BEM) 2014 and the Malaysia Board of Technologists (MBOT) (2017), as well as a senior member of IEEE. Now, Mustafa works as a senior lecturer at University Malaysia Perlis, Malaysia. Mustafa has published more than 300 academic articles with Scopus H-Index 17. Current interests include image processing, biomechanics, intelligence systems, and computer science.

Hiam Hamad Alquran

Dr. Hiam Hamad Alquran is an associate professor in the Department of Biomedical Systems and Informatics Engineering, Yarmouk University, Jordan. Alquran received her PhD (2014) in Biomedical and Biotechnology Engineering from the University of Massachusetts Lowell, USA. She received an M.Sc. (2008) in Automation Engineering from Yarmouk University and her B.S.c in Biomedical Engineering form JUST-Jordan (2005). Research interests include Medical Image Processing, Digital Signal Processing, Pattern Recognition, and Deep Learning. Scholar H. index is 14, Scopus Index is 12.

diagnostics

MDPI

Editorial

Editorial on Special Issue "Medical Data Processing and Analysis"

Wan Azani Mustafa [1,2,*] and Hiam Alquran [3]

[1] Faculty of Electrical Engineering Technology, Campus Pauh Putra, Universiti Malaysia Perlis, Arau 02600, Malaysia
[2] Advanced Computing (AdvComp), Centre of Excellence (CoE), Universiti Malaysia Perlis, Arau 02600, Malaysia
[3] Department of Biomedical Systems and Informatics Engineering, Yarmouk University, Irbid 21163, Jordan; heyam.q@yu.edu.jo
* Correspondence: wanazani@unimap.edu.my

Citation: Mustafa, W.A.; Alquran, H. Editorial on Special Issue "Medical Data Processing and Analysis". *Diagnostics* **2023**, *13*, 2081. https://doi.org/10.3390/diagnostics13122081

Received: 14 June 2023
Accepted: 14 June 2023
Published: 16 June 2023

Medical data plays an essential role in several applications in the medical field. Medical data can be raw data, signals, images, and several observations. The processing and analysis of the data lead to inferring many vital points that help in diagnosis, treatment, progression, and decision-making. Various research handles the medical data in many aspects, such as improving time forecasting models using machine learning for future pandemic applications based on COVID-19 data 2020–2022. The validated proposed method outperformed existing models in both accuracy and efficiency. The authors highlighted the significance of machine learning techniques in predicting the spread of pandemics, in which the proposed model can be exploited for future pandemic applications. The study emphasizes the importance of accurate forecasting models in managing pandemics and highlights the limitations of traditional statistical models in capturing the complex dynamics of pandemics. The designed model overcomes limitations and is considered a standalone public health system [1]. However, it requires the employment of deep learning to extract deep features from white blood cells to classify them into different categories [2]. The optimized entropy-controlled deep features are utilized to build a prominent model that performs success over all existing models in various terms of sensitivity, specificity, and accuracy [2]. The medical data can be transformed using algorithms such as features transformation to predict blood glucose in type 1 diabetes mellitus patients. H. Butt et al. [3] proposed technique outperforms existing accuracy, sensitivity, and specificity methods. They also discussed the potential implications of the proposed technique for improving the management of T1DM and developing more accurate and efficient medical diagnostic tools. Valuable insights were also provided into the current state of research in the field and identified potential avenues for future research [3]. Moreover, using deep learning with medical images improves the detection of *H. pylori* atrophic gastritis. Apart from that, Yasmeen Yacoob et al. [4] proposed a deep learning-based approach for detecting atrophic gastritis caused by Helicobacter pylori (*H. pylori*) infection. The proposed method utilizes an enhanced convolutional neural network (CNN) learner to extract features from endoscopic images of the gastric mucosa and classify them into normal or abnormal categories. The proposed method aims to improve the accuracy of existing detection methods and reduce the need for invasive diagnostic procedures such as biopsy. They explore the impact of accurate detection of *H. pylori*-related atrophic gastritis for preventing and treating gastric cancer, a common complication of chronic *H. pylori* infection. The proposed method is evaluated using a dataset of endoscopic images from patients with *H. pylori* infection. It is compared with other state-of-the-art methods for atrophic gastritis detection. The results show that the proposed method outperforms existing methods in accuracy, sensitivity, and specificity [4].

ECG signals can be represented as images using various transformation techniques. A. Zyout et al. employed a combination of wavelet and Hilbert–Huang transforms to extract relevant features from ECG signals and apply a support vector machine (SVM) classifier to diagnose them. They evaluate the proposed method on a publicly available MIT–BIH arrhythmia database and compare the results with existing ECG wave recognition methods. The proposed method achieved an overall accuracy of 99.25% in recognizing ECG waves, significantly higher than the existing methods. The authors further analyze the performance of the proposed method in identifying individual ECG waves, such as P-waves, QRS complexes, and T-waves. The results show that the proposed method achieves high accuracy in recognizing all the ECG waves, with an average accuracy of 99.4% for P-waves, 99.9% for QRS complexes, and 99.6% for T-waves. The high accuracy achieved in recognizing individual ECG waves is crucial for diagnosing and treating specific cardiac conditions [5]. On the other hand, A. Almulihi et al. [6] proposed an ensemble learning approach based on a hybrid deep learning model for early heart disease prediction. The authors combine a CNN and a long short-term memory (LSTM) network to capture both spatial and temporal features of electrocardiogram (ECG) signals. The proposed method is evaluated on the publicly available PTB diagnostic ECG database and compared with existing ECG-based heart disease prediction models. The results show that the proposed hybrid deep learning model outperforms existing models concerning the accuracy, sensitivity, specificity, and area under the receiver operating characteristic curve (AUC-ROC). The proposed model achieves an accuracy of 95.96%, a sensitivity of 95.85%, a specificity of 96.07%, and an AUC-ROC of 0.9909, which are significantly higher than the existing models. Their proposed method has the potential for clinical use in the early diagnosis and treatment of heart disease [6].

The protein sequence data are exploited to detect COVID-19 and influenza viruses. M. Erten et al. used a hamlet-pattern-based method to extract features from the protein sequences and then apply an SVM classifier for detection. The proposed model is evaluated on several datasets, including the SARS-CoV-2, influenza virus, and non-viral protein datasets. The proposed model achieves an accuracy of 99.61% and an AUC-ROC of 1.00 in detecting the COVID-19 virus and an accuracy of 99.66% and an AUC-ROC of 1.00 in detecting the influenza virus. The model also shows high specificity in detecting viral proteins, with a specificity of 99.99% for COVID-19 and 99.96% for influenza viruses. The proposed model can be used in clinical settings for early detection and diagnosis of COVID-19 and influenza viruses [7]. However, the genomic and tabular data with recurrent neural networks are employed to predict type-2 diabetes. P. N. Srinivasu et al. [8] proposed the RNN model utilizing DNA sequence (as genetic data) and tabular data, such as patient characteristics and medical history. The evaluation is performed on a dataset of patients with and without type-2 diabetes, and the results show that it outperforms other state-of-the-art models in terms of accuracy, sensitivity, specificity, and AUC-ROC. The model achieves an accuracy of 88.1%, a sensitivity of 84.8%, and a specificity of 91.4% in predicting the risk of type-2 diabetes. The AUC-ROC score of the proposed model is 0.935, indicating a high discriminative power in distinguishing between patients with and without type-2 diabetes. The results indicated that the proposed model is effective in clinical settings for type-2 diabetes, either early prediction or early prevention [8].

A. Dairi et al. [9] proposed an anomaly detection method for recognizing mental tasks using EEG signals. The authors exploited a deep learning model to detect normal and abnormal brain patterns during mental activities. The evaluation of the proposed model is performed on a publicly available EEG dataset, which achieved an overall accuracy of 98.31% above traditional classification methods. The proposed model is robust to any variation in noise levels and signal strength. This study suggests that the proposed method could be a practical tool for real-time recognition of mental tasks, potentially enabling applications, such as brain–computer interfaces for controlling external devices with mental commands [9]. Further, the medical data can be used for risk mortality prediction. Moreover, K. Nakamura et al. [10] proposed a deep learning model for predicting the risk of

mortality due to heart failure. The authors utilized a time-varying covariate approach to account for changes in patient health over time, which leads to enhanced accuracy. The evaluation is performed on a large real-world heart failure dataset. The outperforming C-index is 0.741. Here, the model undergoes various conditions to demonstrate its robustness for various input features. It shows that the inclusion of time-varying covariates significantly improved the accuracy of the predictions. The proposed model could be valuable for clinical decision-making concerning heart failure patient management. It enables clinicians to identify high-risk patients and develop targeted treatment plans. The study suggests that deep learning algorithms can potentially improve risk stratification and decision-making in clinical practice for heart failure patients [10].

On the other hand, some researchers detect atrial fibrillation episodes based on 3D algebraic relationships between cardiac intervals. N. W. Qammar et al. [11] proposed a strategy based on the 3D phase space of the cardiac intervals and their algebraic relationships, which are modeled by the Grassmann manifold. The evaluation is carried out on 250 electrocardiogram (ECG) recordings with varying lengths and compared with two state-of-the-art methods for AF detection. The results show that the proposed method outperformed the two state-of-the-art methods in sensitivity, specificity, accuracy, and F1 score. The model is robust against noise and baseline drift, achieving a sensitivity of 97.56%, specificity of 99.15%, accuracy of 98.97%, and F1-score of 0.9804, which are higher than the two state-of-the-art methods. Their approach can be utilized for the early detection of AF episodes, leading to timely interventions and preventing complications associated with AF [11].

The use of medical data to enhance the accuracy of breast cancer diagnosis is widely recognized, as breast cancer remains the most prevalent cancer among women globally. For example, M. Yusoff et al. [12] systematically analyzed 25 papers published from 2018 to 2020 and evaluated their methods and performance metrics. The results showed that CNNs were the most used deep learning approach for breast cancer classification, followed by ensemble methods and recurrent neural networks (RNNs). The review also identified various data types used for classification, such as mammography images, histopathology images, and gene expression profiles. The evaluation uses different metrics, such as accuracy, sensitivity, specificity, F1-score, and area under the curve (AUC). The average accuracy of the reviewed methods ranged from 70% to 99%, with most achieving accuracy above 90%. The study also highlighted limitations and challenges in the reviewed papers, such as limited sample sizes, lack of standardized datasets, and difficulty interpreting deep learning models' decision-making process. In conclusion, the article provides a comprehensive overview of the current state of deep learning methods in breast cancer classification and identifies potential areas for future research [12].

Funding: This research received no external funding.

Conflicts of Interest: The writers certify they have no conflicting interest in this research.

References

1. KAbdul Hamid, A.A.; Wan Mohamad Nawi WI, A.; Lola, M.S.; Mustafa, W.A.; Abdul Malik, S.M.; Zakaria, S.; Abdullah, M.T. Improvement of Time Forecasting Models Using Machine Learning for Future Pandemic Applications Based on COVID-19 Data 2020–2022. *Diagnostics* **2023**, *13*, 1121. [CrossRef] [PubMed]
2. Ahmad, R.; Awais, M.; Kausar, N.; Akram, T. White Blood Cells Classification Using Entropy-Controlled Deep Features Optimization. *Diagnostics* **2023**, *13*, 352. [CrossRef] [PubMed]
3. Butt, H.; Khosa, I.; Iftikhar, M.A. Feature Transformation for Efficient Blood Glucose Prediction in Type 1 Diabetes Mellitus Patients. *Diagnostics* **2023**, *13*, 340. [CrossRef] [PubMed]
4. Yacob, Y.M.; Alquran, H.; Mustafa, W.A.; Alsalatie, M.; Sakim, H.A.M.; Lola, M.S.H. pylori Related Atrophic Gastritis Detection Using Enhanced Convolution Neural Network (CNN) Learner. *Diagnostics* **2023**, *13*, 336. [CrossRef] [PubMed]
5. Zyout, A.; Alquran, H.; Mustafa, W.A.; Alqudah, A.M. Advanced Time-Frequency Methods for ECG Waves Recognition. *Diagnostics* **2023**, *13*, 308. [CrossRef] [PubMed]
6. Almulihi, A.; Saleh, H.; Hussien, A.M.; Mostafa, S.; El-Sappagh, S.; Alnowaiser, K.; Ali, A.A.; Hassan, M.R. Ensemble Learning Based on Hybrid Deep Learning Model for Heart Disease Early Prediction. *Diagnostics* **2022**, *12*, 3215. [CrossRef] [PubMed]

7. Erten, M.; Acharya, M.R.; Kamath, A.P.; Sampathila, N.; Bairy, G.M.; Aydemir, E.; Barua, P.D.; Baygin, M.; Tuncer, I.; Dogan, S.; et al. Hamlet-Pattern-Based Automated COVID-19 and Influenza Detection Model Using Protein Sequences. *Diagnostics* **2022**, *12*, 3181. [CrossRef] [PubMed]
8. Srinivasu, P.N.; Shafi, J.; Krishna, T.B.; Sujatha, C.N.; Praveen, S.P.; Ijaz, M.F. Using Recurrent Neural Networks for Predicting Type-2 Diabetes from Genomic and Tabular Data. *Diagnostics* **2022**, *12*, 3067. [CrossRef] [PubMed]
9. Dairi, A.; Zerrouki, N.; Harrou, F.; Sun, Y. EEG-Based Mental Tasks Recognition via a Deep Learning-Driven Anomaly Detector. *Diagnostics* **2022**, *12*, 2984. [CrossRef] [PubMed]
10. Nakamura, K.; Zhou, X.; Sahara, N.; Toyoda, Y.; Enomoto, Y.; Hara, H.; Noro, M.; Sugi, K.; Huang, M.; Moroi, M.; et al. Risk of Mortality Prediction Involving Time-Varying Covariates for Patients with Heart Failure Using Deep Learning. *Diagnostics* **2022**, *12*, 2947. [CrossRef] [PubMed]
11. Qammar, N.W.; Šiaučiūnaitė, V.; Zabiela, V.; Vainoras, A.; Ragulskis, M. Detection of Atrial Fibrillation Episodes based on 3D Algebraic Relationships between Cardiac Intervals. *Diagnostics* **2022**, *12*, 2919. [CrossRef] [PubMed]
12. Yusoff, M.; Haryanto, T.; Suhartanto, H.; Mustafa, W.A.; Zain, J.M.; Kusmardi, K. Accuracy Analysis of Deep Learning Methods in Breast Cancer Classification: A Structured Review. *Diagnostics* **2023**, *13*, 683. [CrossRef] [PubMed]

diagnostics

MDPI

Article

Detection of Atrial Fibrillation Episodes based on 3D Algebraic Relationships between Cardiac Intervals

Naseha Wafa Qammar [1], Vaiva Šiaučiūnaitė [1], Vytautas Zabiela [2], Alfonsas Vainoras [2,*] and Minvydas Ragulskis [1]

[1] Department of Mathematical Modelling, Kaunas University of Technology, LT-51368 Kaunas, Lithuania
[2] Cardiology Institute, The Lithuanian University of Health Sciences, Mickeviciaus g.9, LT-44307 Kaunas, Lithuania
* Correspondence: alfonsas.vainoras@lsmu.lt

Abstract: In this study, the notion of perfect matrices of Lagrange differences is employed to detect atrial fibrillation episodes based on three ECG parameters (JT interval, QRS interval, RR interval). The case study comprised 8 healthy individuals and 7 unhealthy individuals, and the mean and standard deviation of age was 65.84 ± 1.4 years, height was 1.75 ± 0.12 m, and weight was 79.4 ± 0.9 kg. Initially, it was demonstrated that the sensitivity of algebraic relationships between cardiac intervals increases when the dimension of the perfect matrices of Lagrange differences is extended from two to three. The baseline dataset was established using statistical algorithms for classification by means of the developed decision support system. The classification helps to determine whether the new incoming candidate has indications of atrial fibrillation or not. The application of probability distribution graphs and semi-gauge indicator techniques aided in visualizing the categorization of the new candidates. Though the study's data are limited, this work provides a strong foundation for (1) validating the sensitivity of the perfect matrices of Lagrange differences, (2) establishing a robust baseline dataset for supervised classification, and (3) classifying new incoming candidates within the classification framework. From a clinical standpoint, the developed approach assists in the early detection of atrial fibrillation in an individual.

Keywords: atrial fibrillation; perfect matrix of Lagrange differences; statistical indicator; decision support system

Citation: Qammar, N.W.; Šiaučiūnaitė, V.; Zabiela, V.; Vainoras, A.; Ragulskis, M. Detection of Atrial Fibrillation Episodes based on 3D Algebraic Relationships between Cardiac Intervals. *Diagnostics* **2022**, *12*, 2919. https://doi.org/10.3390/diagnostics12122919

Academic Editors: Wan Azani Mustafa and Hiam Alquran

Received: 21 October 2022
Accepted: 18 November 2022
Published: 23 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

1.1. Existing Diagnostics Techniques for Atrial Fibrillation

Atrial fibrillation (AF) is linked to an increased risk of cardiovascular events such as cardiovascular mortality, major cardiovascular events, heart failure, ischemic heart disease, sudden cardiac death, and stroke [1,2]. Numerous symptoms, such as fatigue, palpitations, shortness of breath, and chest pain, are experienced by AF patients. Some patients have no symptoms, which is known as asymptomatic or "silent" AF. Asymptomatic AF has serious clinical consequences. Patients with undiagnosed AF may develop life-threatening thromboembolic complications or tachycardia-mediated cardiomyopathy [3,4]. People with AF disorders often have no symptoms or vague signs because AF is frequently intermittent. This combination consequently makes detection and diagnosis challenging. Most of the time, AF is not noticeable until a person experiences a serious health issue and seeks therapy [2]. One of the key tactics for enhancing AF identification and possibly lowering AF-related stroke, mortality, and healthcare expenditures is screening for AF [4]. The rhythm disturbance's frequent silence makes it difficult to detect AF early, which is a serious concern. The so-called asymptomatic AF is unknown to patients in around one-third of those who have this arrhythmia. Much earlier detection of the arrhythmia may enable the prompt introduction of medicines to protect patients not only from the consequences of the arrhythmia, but also from the progression of AF from a treatable illness into an untreatable issue [5].

There are various heart health monitoring tools available on the market, such as AliveCor [2], that show promise, particularly for patients with intermittent AF. However, this equipment is expensive and unable to identify those with AF condition symptoms. Today, a variety of methods, including Single Lead, Holter Monitoring, Mobile Telemetry Monitoring, and Implantable Loop Recorders, are available for use in AF detection outside of the hospital setting. Spot single-lead ECG monitors, such as the AliveCor, need the patient to identify symptoms, document them, and discuss them with their clinician [2]. It can be concluded that current heart health monitoring methods are either excessive or call for specialized, pricey equipment that must be used in conjunction with substantial signal processing to produce better quality findings. People with known or potential cardiovascular problems must have access to a low-cost heart health monitoring system in order to lower the expense of medical examinations, lessen their worry, and receive timely care.

The electrocardiogram (ECG) visual examination is the gold standard for identifying AF. An ECG is required to identify AF; nevertheless, an erratic pulse can increase the suspicion that it may be present [6,7]. A 12-lead ECG can be used to confirm the presence of AF, a frequent persistent arrhythmia that is frequently asymptomatic [2]. In addition to the severe irregularity of the RR interval, P waves vanish and are replaced by irregular fibrillation waves (f waves) of various sizes and forms [8,9].

1.2. Perfect Matrices of Lagrange Differences as a Method for ECG Signal Analysis

ECG signals have been examined over time to detect and examine various cardiovascular diseases [9]. Before delving into the analysis techniques of the ECG signal, it is important to recall that the ECG signal is a complex signal in its behavior [10,11]. The complexity explicates how the various ECG parameters [12] have significant variation between them, which is caused by various physiological and pathological factors, so therefore, no definite mathematical model can characterize the relationship between ECG parameters, even for a specific individual [11,13].

There is always an attempt to analyze the ECG signal in such a way that the data with the vast bulk of the extraction of clinically relevant features are available, which contain all the important information of the original ECG signal and thus operate as the signal's substitute for further analysis. Several computational approaches including deep learning methods, such as feature extraction and dimensionality reduction, are well known for conducting the analysis of ECG. These techniques are helpful for increasing clinical research by gaining a better understanding of medical challenges [14,15]. Feature extraction technique can occur in the frequency domain, time domain, or frequency–time domain analysis [16]. Noujaim et al. [17–19] propose that the behavior of the cardiovascular system is "fractal-like", attempting to retain adaptive variability demonstrating non-linearity in the cardiovascular system rather than stability. As a result, it makes more sense to seek a solution to dynamic processes that are complicated in nature [11]. A range of well-known nonlinear analytic techniques for analyzing ECG data have been used due to their robustness. For instance, reconstructed phase space analysis [20], Lyapunov exponents [21,22], correlation dimension [23–25], detrended fluctuation analysis (DFA) [26], recurrence plot and Poincaré plots [27], and possibly additional nonlinear analytic approaches for ECG analysis are becoming increasingly popular.

It is essential to surpass the boundaries of conventional nonlinear computation techniques to determine potentially accurate, faster, and more reliable solutions for the analysis of the ECG parameter. The idea of the perfect matrices of Lagrange differences has been proposed and proved for the different nonlinear time series analysis applications in several studies [10,11,28]. One of the objectives of this paper is to show that the expansion of the matrix dimension does result in the increased sensitivity of the algebraic relationships between cardiac intervals, which provides a more accurate platform for the early detection of atrial fibrillation. First, the architecture of the third-order perfect matrices of Lagrange differences is introduced. Then, the sensitivity of the proposed architecture is demonstrated

by comparing the proposed architecture to the second-order perfect matrices of Lagrange differences used in previous studies. The rest of the structure of the paper is as follows. The notion of the perfect matrices of Lagrange differences is explained (this part is basically a recall from the work [10,11] and is hence named "*Preliminary Synopsis*"). The perfect matrices of Lagrange differences are used to explain how to interpret the relationships between the three cardiac intervals (JT interval, QRS interval, RR interval). It is demonstrated (and validated) that the expansion of the dimension of the matrix serves as an accurate platform for assessing the ECG signal. Subsequently, using the one sigma rule, statistical operations are performed on the dataset to establish an observation window for the classification purposes. In addition to the statistical analysis, it is shown how the generated probability index and the semi-gauge representation help to serve as the decision support instrument for the detection of early AF episodes.

2. Methods

2.1. The Description of the Experimental Setup

The three cardiac intervals, (i) JT interval, (ii) QRS interval, and (iii) RR interval, are registered as three time series, as shown in Figure 2. The ECG signal is recorded under the "No Load" condition, which indicates that the participants did not exercise or perform any stress tests while the ECG recordings were registered.

2.2. Participants

Participants in this study were divided into two groups: healthy and unhealthy individuals. It is worth mentioning that the ECG data collected for both groups showed a proclivity for atrial fibrillation (AF). Antiarrhythmic medication was not reported to be used by either group of subjects; therefore, the cohorts were divided and categorized as healthy or unhealthy. The cohort of healthy individuals comprised 8 people, whereas the cohort of unhealthy individuals comprised 7 people. The mean and standard deviation of age was 65.84 $\pm$ 1.4 years; of height, 1.75 $\pm$ 0.12 m; and of weight, 79.4 $\pm$ 0.9 kg. For purposes of confidentiality, the candidates' names were labeled as H1–H8, which indicate healthy candidates, and UH1-UH7, which denote unhealthy candidates (see Tables 3–6).

2.3. Ethics Statement

The research met all applicable standards for the ethics of experimentation in accordance with the Declaration of Helsinki as reflected in prior approval by the Regional Biomedical Research Ethics Committee of the Lithuanian University of Health Sciences (ID No. BE-2-4, 17 March 2016). The permit to perform biomedical investigation was granted by the LUHS Bioethics Committee (see the Institutional Review Board Statement). Participants provided written informed consent prior to the experiment.

2.4. The Description of the Proposed Algorithm

2.4.1. Preliminary Synopsis

Let us begin by presenting a brief overview of the three cardiac intervals examined in this study (JT interval, QRS interval, and RR interval). Figure 2 depicts a time series of the three cardiac intervals for one of the candidates. In Figure 2, the x-axis represents the time in minutes, whereas the y-axis represents the time series recorded for the each of the three cardiac intervals. Let us denote each of these three time series as $x = (x_1, x_2, x_3, \cdots\cdots, x_i)$, $y = (y_1, y_2, y_3, \cdots\cdots, y_i)$, and $z = (z_1, z_2, z_3, \cdots\cdots, z_i)$, where i is the total number of heartbeats recorded throughout the experiment.

The key concept, notion, and underlying conditions for the generation of 2-by-2 perfect matrices of Lagrange differences and the corresponding algebraic relationship have been previously thoroughly explained and verified [10,11,28]. Furthermore, the fact that derivatives tend to amplify noise, as well as the implications of reasonable mathematical manipulations for the extraction of the most relevant information from the ECG signal, have already been addressed in [10,11,28] and will not be explained in this work. What

makes this research distinct from previous approaches is that (1) the order of the matrices is increased, thus enhancing the sensitivity of the proposed algorithm, and (2) the idea of employing this method for early detection of AF from a cohort of healthy and unhealthy individuals. The next section discusses the architecture of the 3-by-3 perfect matrices of Lagrange differences and then reflects on its potential sensitivity.

2.4.2. The Architecture of Third-Order Square Matrices of Lagrange Differences

The definition of 2-by-2 perfect matrices of Lagrange differences is given in [10,11,28]. Each element of the matrix can be either a zeroth-order difference or a first-order difference. The following conditions must hold true for the matrix in order be a perfect matrix of Lagrange differences [10,11,28]:

1. All elements of the matrix must be different.
2. Zeroth-order differences are located on the main diagonal.
3. First-order differences are located on the secondary diagonal.
4. The matrix is balanced with respect to time (the sum of all time lags is equal to zero).
5. The matrix is balanced with respect to lexicographic variables (the number of different symbols must be the same).

Let us assume that index n denotes the current moment and δ represents the time lag ($\delta \epsilon \mathbb{N}$). Then, the structure of the 2-by-2 perfect matrix of Lagrange differences reads [10,11,28]:

$$\begin{bmatrix} x_n & x_{n+\delta} - y_{n+\delta} \\ x_{n-\delta} - y_{n-\delta} & y_n \end{bmatrix} \quad (1)$$

The schematic representation of this 2-by-2 perfect matrix of Lagrange differences can be illustrated by the diagram in Figure 1. The diagonal elements of the matrix are shown in circles. The first-order Lagrange differences are depicted by arrows connecting respective elements.

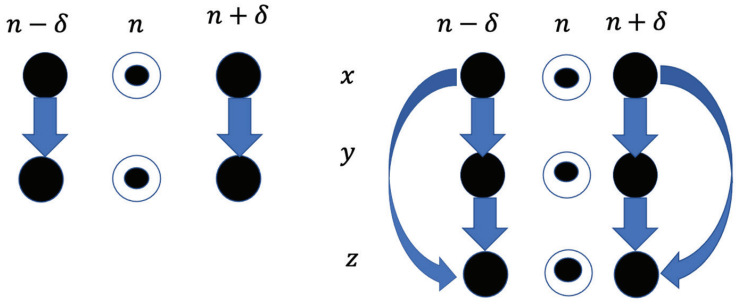


Figure 1. A schematic diagram illustrating the architecture of 2-by-2 and 3-by-3 perfect matrices of Lagrange differences. Zeroth-order differences (the diagonal elements of the matrices) are depicted in circles. First-order differences are depicted by arrows connecting respective elements.

Let us consider the three different time series x, y, and z (Table 1) also shown in Figure 2. The architecture of the 3-by-3 matrix can be naturally expanded by scaling the structure of Equation (2).

$$\begin{bmatrix} x_n & y_{n+\delta} - x_{n+\delta} & z_{n+\delta} - x_{n+\delta} \\ y_{n-\delta} - x_{n-\delta} & y_n & z_{n+\delta} - y_{n+\delta} \\ z_{n-\delta} - x_{n-\delta} & z_{n-\delta} - y_{n-\delta} & z_n \end{bmatrix} \quad (2)$$

Table 1. The nine different elements of the time series x, y, and z with current, time-backward, and time-forward indexes.

$x_{n-\delta}$	x_n	$x_{n+\delta}$
$y_{n-\delta}$	y_n	$y_{n+\delta}$
$z_{n-\delta}$	z_n	$z_{n+\delta}$

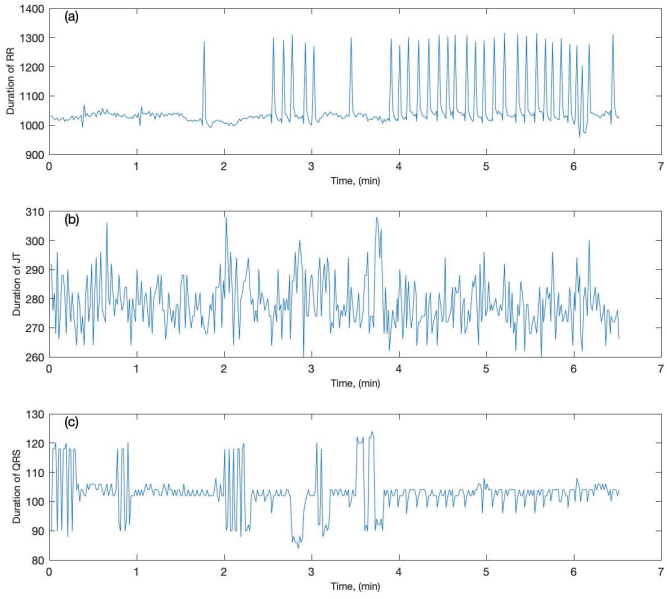


Figure 2. The three recorded cardiac intervals are plotted as time series (the time is in minutes): (**a**) the RR interval; (**b**) the JT interval; (**c**) the QRS interval.

It can be observed that all requirements for perfect matrices of Lagrange differences hold true for Equation (2). Therefore, the matrix defined by Equation (2) is also a perfect matrix of Lagrange differences. The schematic representation of the 3-by-3 perfect matrix of Lagrange differences is depicted in Figure 1. Again, the diagonal elements (the zeroth-order Lagrange differences) are shown in circles; all first-order Lagrange differences are depicted by arrows connecting respective elements.

The algebraic relationship between two cardiac intervals (sequences x and y) is defined by a mapping function $\mathcal{F} : \mathbb{R}^{(2*2)} \rightarrow \mathbb{R}^1$, which transforms each perfect matrix of Lagrange differences to a scalar variable. Different mapping functions are used in [10,11,28] (the discriminant of the matrix, the modulus of the maximal eigenvalue of the matrix, the norm of the matrix). The same mapping function $\mathcal{F} : \mathbb{R}^{(3*3)} \rightarrow \mathbb{R}^1$ should be defined for the 3-by-3 perfect matrix of Lagrange differences. The norm of the matrix is set at the mapping function in all further computations. Internal and external smoothing of the mapped scalar algebraic relationship between cardiac intervals are executed after the mapping procedure [28]; the same radiuses of internal and external smoothing are adapted from [28].

2.4.3. The Sensitivity of the Proposed Algorithm

After illustrating the structure of the 3-by-3 perfect matrices of Lagrange differences, the sensitivity of the proposed architecture is explored. To this end, one of the candidates from the cohort of individuals is employed as an example. The ECG signal is recorded for the candidate (Figure 2). Initially, all computations are performed with 2-by-2 perfect matrices of Lagrange differences, as described in Section 2. The algebraic relationship

between JT and QRS intervals is shown in Figure 3a; the algebraic relationship between JT and RR intervals is shown in Figure 3b; the algebraic relationship between QRS and RR intervals is shown in Figure 3c.

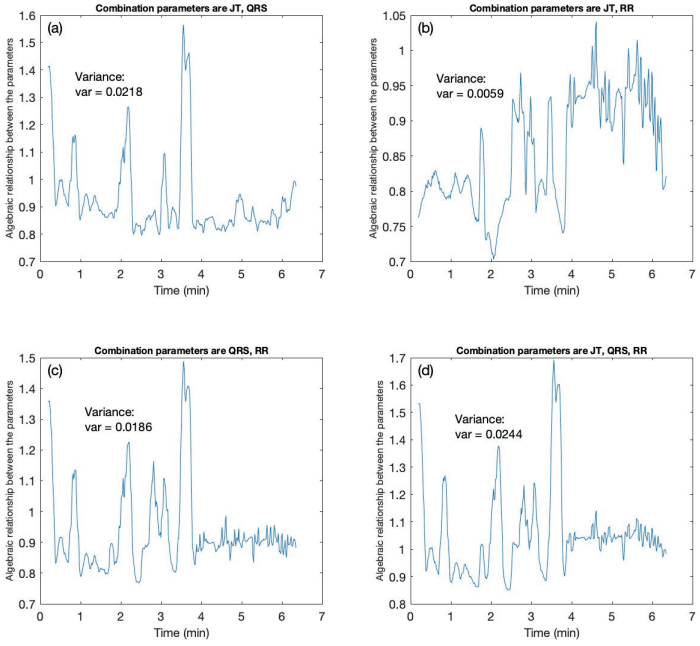


Figure 3. The sensitivity of 3-by-3 matrices vs 2-by-2 matrices is demonstrated for one of the candidates. The x-axis denotes the time in minutes. The y-axis demonstrates the algebraic relationship for the 2-by-2 (**a**–**c**) and 3-by-3 (**d**) matrix combination for the cardiac intervals JT, QRS, RR. (**a**) The 2-by-2 matrix algebraic relation is computed for the combination of cardiac intervals JT, QRS. (**b**) The 2-by-2 matrix algebraic relation is computed for the combination of cardiac intervals JT, RR. (**c**) The 2-by-2 matrix algebraic relationship is computed for the cardiac intervals QRS, RR. (**d**) The 3-by-3 matrix algebraic relationship is computed for the combination of cardiac intervals JT, QRS, RR.

These computations are repeated with all recorded cardiac intervals (JT, QRS, RR intervals). Three scalar time series are mapped into a sequence of 3-by-3 matrices, as shown in Figure 1. It is to be noted that the order of the parameter also has an important role in defining the sensitivity of the proposed algorithm. The order of the parameters in this study is fixed with JT interval as x, QRS interval as y, and RR interval as z (see Section 3 for more details). Once the three parameters are according to the proposed architecture, they are evaluated by using computational techniques discussed above. (1) The recorded time series are mapped into the trajectory matrices. (2) With the mapping $(\mathcal{F})$ technique, the trajectory of the matrices is transformed into a scalar time series $(\mathcal{F} : \mathbb{R}^{(3*3)} \rightarrow \mathbb{R}^1)$. As noted previously, the mapping function $\mathcal{F}$ is defined as the norm of the 3-by-3 perfect matrix of Lagrange differences. Finally, the internal and external smoothing techniques are applied to smooth the mapped scalar sequence.

The sensitivity of the algebraic relationship revealed by the 2-by-2 and 3-by-3 perfect matrices of Lagrange differences is measured by the variability (the variance) of the scalar mapped signal depicted in Figure 3. The algebraic relationship between JT and QRS intervals yields the variance value of 0.0218 (Figure 3a). Analogously, the JT–RR relationship yields 0.0059 (Figure 3b); the QRS–RR relationship yields 0.0186 (Figure 3c). The 3-by-3 algebraic relationship between JT, QRS, and RR intervals yields the variance value of 0.0244 (Figure 3d). This increase in the sensitivity of the algebraic relationship can be explained by

the following reasoning. Only two of four terms represented differences in the structure of a 2-by-2 perfect matrix of Lagrange differences. However, six of nine terms represented differences in the structure of a 3-by-3 perfect matrix of Lagrange differences. As mentioned previously, the differences better reflect the changes in the inter-connected system represented by related synchronized cardiac intervals representing every consecutive heart contraction. Moreover, the information fed into the algorithm is much larger (three cardiac intervals instead of two). Therefore, a higher sensitivity of the 3-by-3 matrix architecture is not surprising.

The values of the variances obtained from the second- and third-order perfect matrices of Lagrange differences are tabulated in Table 2. With the comparison between the variance values for both second- and third-order matrices, the sensitivity of the proposed algorithm is highly exhibited by the 3-by-3 perfect matrices of Lagrange differences. The results tabulated in Table 2 and shown in Figure 3 confirm the hypothesis of this study. Before we delve into the results and discussions, we review the number of computations performed on our dataset to determine (1) which combination patterns exhibit more sensitivity from the 3-by-3 perfect matrices of Lagrange differences and (2) which combination patterns produced the most likely statistical outcomes for the statistical computations.

Table 2. The comparison between the sensitivity of 2-by-2 and 3-by-3 matrices. The variance value for each of the combination is tabulated for one of the candidates (see also Figure 2).

Candidate	Variance Values			
	Combination of JT–QRS Interval	**Combination of JT–RR Interval**	**Combination of QRS–RR Interval**	**Combination of JT–QRS–RR Interval**
	0.0218	0.0059	0.0186	0.0244

2.4.4. Trials Computed upon the Dataset with Different Combination Patterns

This section explores different combinations of the parameters to determine which combination is more sensitive in characterizing the algebraic relationship between the parameters and which statistical parameters should be chosen for the subsequent computations. With the number of vectors as three, $k = 3$, the possible number of combinations that could be made is $2^k = 2^3 = 8$. The computations are performed on all the possible combinations (i.e., $2^k = 2^3$); a few of them are highlighted in Tables 3–6. As mentioned before, the computations were conducted with two main objectives: (1) by altering the parameters' order, it is intended to assess the sensitivity of the proposed algorithm, and (2) to look for the most suitable statistical analysis parameter for further computations, whether it is variance (σ^2), mean (σ), median ($x \sim$), or standard deviation (σ_x). In Table 3, the computational techniques are performed for the combination parameters (QRS interval, RR interval, and JT interval). First, the variance (σ^2), mean (σ), median ($x \sim$), and standard deviation (σ_x) are computed for the sequence representing the algebraic relationship between cardiac intervals (see Table 3). To obtain more statistically significant results, the one sigma rule is applied to the smoothed sequence for the cohort of both healthy and unhealthy people. Note that the one sigma rule approach applied here must not be confused with the one sigma rule used subsequently in the study for the variance data. At this point in the study, the goal of employing the one sigma rule is to identify the variability in the algebraic relationship between the various combinations of parameters. In addition, the one sigma rule is applied to the median and standard deviation of the smoothed algebraic sequence, but later in the study, the one sigma rule will be applied to the variance data when the statistical manipulations and computer-generated procedures are applied to it.

Table 3. Statistical analysis outcome for the combination of parameters QRS interval, RR interval, and JT interval.

Healthy Candidates (H)							Unhealthy Candidates (UH)						
Variance σ^2_H	Mean μ_H	Median $x\sim_H$	Standard Deviation σ_H	Upper Limit $(x\sim_H+\sigma_H)$	Lower Limit $(x\sim_H-\sigma_H)$	Area A_H	Variance σ^2_{UH}	Mean μ_{UH}	Median $x\sim_{UH}$	Standard Deviation σ_{UH}	Upper Limit $(x\sim_{UH}+\sigma_{UH})$	Lower Limit $(x\sim_{UH}-\sigma_{UH})$	Area A_{UH}
0.00080583	0.736	0.734	0.028	0.763	0.707	0.359	**0.0003**	1.168	1.164	0.018	1.182	0.004	0.769
0.0016751	0.827	0.821	0.041	0.863	0.781	1.100	**0.0002**	1.281	1.279	0.015	1.294	1.264	0.875
0.0015197	0.917	0.916	0.039	0.956	0.878	−0.376	**0.0061**	1.018	1.012	0.078	1.091	0.935	2.062
0.00073806	1.031	1.031	0.027	1.059	1.004	0.254	**0.0006**	0.932	0.936	0.025	0.962	0.912	−1.217
0.0015323	1.024	1.027	0.039	1.067	0.989	−1.307	**0.0141**	1.422	1.409	0.119	1.528	1.291	1.461
0.0010183	0.875	0.875	0.032	0.907	0.843	0.148	**0.0027**	0.822	0.811	0.052	0.864	0.760	2.422
0.010428	1.317	1.307	0.102	1.41	1.206	1.508	**0.0015**	1.011	1.006	0.039	1.045	0.967	1.466
0.0060534	0.899	0.881	0.078	0.959	0.804	4.437							

Table 4. Statistical analysis outcome for the combination of parameters RR interval, JT interval, and QRS interval.

Healthy Candidates (H)							Unhealthy Candidates (UH)						
Variance σ^2_H	Mean μ_H	Median $x\sim_H$	Standard Deviation σ_H	Upper Limit $(x\sim_{UH}+\sigma_{UH})$	Lower Limit $(x\sim_{UH}-\sigma_{UH})$	Area A_H	Variance σ^2_{UH}	Mean μ_{UH}	Median $x\sim_{UH}$	Standard Deviation σ_{UH}	Upper Limit $(x\sim_{UH}+\sigma_{UH})$	Lower Limit $(x\sim_{UH}-\sigma_{UH})$	Area A_{UH}
0.000792	0.736	0.736	0.028	0.764	0.708	0.242	**0.0003**	1.168	1.164	0.018	1.182	1.146	0.775
0.001677	0.827	0.821	0.041	0.862	0.780	1.147	**0.0002**	1.282	1.279	0.015	1.294	1.264	0.896
0.001713	0.921	0.922	0.041	0.963	0.880	−0.494	**0.0061**	1.016	1.011	0.078	1.089	0.932	2.051
0.000767	1.033	1.033	0.026	1.061	1.005	0.310	**0.0006**	0.933	0.937	0.025	0.962	0.911	−1.264
0.001689	1.031	1.034	0.041	1.075	0.999	−1.086	**0.0141**	1.424	1.411	0.119	1.529	1.292	1.477
0.001000	0.881	0.879	0.032	0.911	0.848	0.231	**0.0027**	0.823	0.812	0.052	0.864	0.759	2.434
0.01042	1.319	1.310	0.102	1.412	1.208	1.525	**0.0015**	1.012	1.006	0.039	1.046	0.967	1.478
0.006422	0.906	0.891	0.080	0.976	0.810	4.392							

Table 5. Statistical analysis outcome for the combination of parameters JT interval, RR interval, and QRS interval.

Healthy Candidates (H)							Unhealthy Candidates (UH)						
Variance σ_H^2	Mean μ_H	Median $x\sim_H$	Standard Deviation σ_H	Upper Limit $(x\sim_H+\sigma_H)$	Lower Limit$(x\sim_H-\sigma_H)$	Area A_H	Variance σ_{UH}^2	Mean μ_{UH}	Median $x\sim_{UH}$	Standard Deviation σ_{UH}	Upper Limit $(x\sim_{UH}+\sigma_{UH})$	Lower Limit $(x\sim_{UH}-\sigma_{UH})$	Area A_{UH}
0.0013	0.698	0.694	0.037	0.731	0.658	1.565	**0.0006**	1.397	1.393	0.025	1.418	1.367	0.9204
0.0030	0.882	0.880	0.055	0.934	0.825	0.527	**0.0004**	1.539	1.535	0.019	1.554	1.516	1.1994
0.0025	1.039	1.032	0.050	1.082	0.982	1.427	**0.0090**	1.051	1.043	0.095	1.138	0.948	2.2968
0.0014	1.165	1.168	0.037	1.206	1.131	−0.806	**0.0014**	1.070	1.077	0.037	1.114	1.040	−2.2801
0.0043	1.169	1.184	0.066	1.250	1.119	−5.280	**0.0067**	1.293	1.283	0.082	1.365	1.201	1.1756
0.0029	0.905	0.909	0.054	0.963	0.856	−0.469	**0.0031**	0.900	0.900	0.056	0.956	0.845	0.6442
0.0031	1.441	1.446	0.055	1.502	1.391	−1.105	**0.0014**	1.140	1.139	0.038	1.177	1.101	0.5454
0.0021	1.003	0.998	0.046	1.044	0.953	1.300							

Table 6. Statistical computations for the cohort of healthy and unhealthy individuals for the combination of parameters JT interval, QRS interval, and RR interval. The variance, mean, median, and standard deviation values are tabulated.

Healthy Candidates (H)							Unhealthy Candidates (UH)						
Variance σ_H^2	Mean μ_H	Median $x\sim_H$	Standard Deviation σ_H	Upper Limit $(x\sim_H+\sigma_H)$	Lower Limit $(x\sim_H-\sigma_H)$	Area A_H	Variance σ_{UH}^2	Mean μ_{UH}	Median $x\sim_{UH}$	Standard Deviation σ_{UH}	Upper Limit $(x\sim_{UH}+\sigma_{UH})$	Lower Limit $(x\sim_{UH}-\sigma_{UH})$	Area A_{UH}
0.0012	0.692	0.689	0.035	0.724	0.654	1.370	**0.0006424**	1.397	1.393	0.025	1.418	1.367	0.904
0.0030	0.881	0.880	0.055	0.934	0.825	0.435	**0.0003502**	1.538	1.535	0.019	1.553	1.516	1.219
0.0023	1.032	1.025	0.048	1.073	0.978	1.565	**0.0082**	1.038	1.031	0.091	1.122	0.940	2.147
0.0014	1.161	1.164	0.037	1.201	1.127	−0.773	**0.0015**	1.065	1.073	0.039	1.112	1.034	−2.483
0.0040	1.162	1.176	0.064	1.240	1.113	−5.041	**0.0064**	1.285	1.274	0.080	1.354	1.194	1.532
0.0025	0.894	0.895	0.050	0.945	0.845	0.042	**0.0030**	0.897	0.900	0.055	0.955	0.844	0.355
0.0031	1.431	1.437	0.056	1.493	1.382	−1.224	**0.0014**	1.135	1.134	0.038	1.172	1.096	0.548
0.0018	0.991	0.987	0.042	1.029	0.945	1.159							

As the implication of the one sigma rule, the upper boundary ($x \sim_H + \sigma_H$) and lower boundary ($x \sim_H - \sigma_H$) are defined. Subsequently, the areas (A) above and below the boundaries are computed. With the boundaries defined, statistical conditions are applied to the smoothed sequence. If the smooth sequence is greater than the upper limit, then the area above the upper limit is computed, and if the smoothed sequence is less than the lower limit, the area below the lower limit is computed. The final outcome of the overall area computed is denoted by A (see Tables 3–5). It makes sense to conduct computational trials to determine the extent of the variability that exists while defining the algebraic relationship between the three parameters. For the combination (QRS interval, RR interval, JT interval), we do see the variability, as shown by the statistical outcomes tabulated in Table 3. The same statistical techniques have been performed for the other three sequences (RR interval, JT interval, and QRS interval; JT interval, RR interval, and QRS interval; JT interval, QRS interval, and RR interval—see Tables 4–6). It is worth noting that while variability exists in all of these combinations, upon close inspection of the statistical data, it is particularly significant for the parameter combination of JT interval, QRS interval, and RR interval (see Tables 3–6).

Furthermore, upon the comprehensive review of the mean (σ), median ($x \sim$), and standard deviation (σ_x) data for the cohort of healthy and unhealthy individuals, it is revealed that the variance (σ^2) data are considerably distinctive and sensitive in their conclusions for both cohorts in all combinations (see Tables 3–6) and can be utilized to generate a reliable baseline dataset for the classification, which can then be used to classify new incoming candidates. Therefore, the variance (σ^2) data are chosen for generating the Gaussian distribution graphs and producing the baseline dataset. In the next section, the discussion on the statistical analysis techniques and the subsequent outcomes are elaborated in detail.

2.4.5. The Development of the Decision Support System

Following the sensitivity assessment of the 3-by-3 matrices based on the variance values (highlighted and underlined in Table 6), the statistical computations are performed for the cohort of healthy and unhealthy individuals. The retrieved knowledge from the algebraic relationship is employed to conduct the statistical analysis. For example, the variance values are used to perform the statistical analysis for further computations. The variance values will be employed to estimate the Gaussian distribution plot for both cohorts (see Figure 4). The distribution will help to determine how evenly the data are distributed along the x-axis. Since the distribution graph will assist in establishing the foundations for classification, the one sigma rule is applied to the Gaussian distribution plot for classification (see Figure 5a). The data fitted to the classification serve as the foundation for classifying the new incoming individual in order to determine whether he/she falls into the category of a healthy or unhealthy individual based on the classification rules established in Table 7. To strengthen the visualization of classification of the new candidate, a graphical representation of the probability distribution graph is proposed, followed by a gauge representation with coloring indication for generating the warning system. If an incoming person within the classification demonstrates either healthy or unhealthy behavior, the classification approach will single out the individual according to his/her ECG registering, and the warning gauge will provide a graphical depiction via a color on the gauge that might be green, yellow, or red (based on the individual's behavior). For example, if a person enters the classification and exhibits the behavior of a healthy person, the gauge arrow will point towards the green color, indicating that the incoming person is healthy. On the other hand, if the incoming person indicates that he/she falls even slightly towards the unhealthy category, the indication gauge arrow will point towards yellow, which means that the incoming person may have indications of atrial fibrillation based on his/her ECG registering. Finally, if the approaching individual falls far towards the right of the distribution graph, it strongly indicates that the incoming person has a higher risk

of having atrial fibrillation episodes and will be indicated by the color red. Each of these scenarios are discussed more extensively in Section 3.

Table 7. Classification rules specified for the new incoming candidate into the variation interval.

	Condition 1	**Condition 2**	**Condition 3**
If	$C \leq \mu_h - \sigma_h$	$C \geq \mu_{uh} + \sigma_{uh}$	$\mu_{uh} - \sigma_{uh} \leq C \leq \mu_h + \sigma_h$
Then, indicator is	$IND = 0$	$IND = 1$	$IND = \frac{C-(\mu_{uh}-\sigma_{uh})}{(\mu_h+\sigma_h)-(\mu_{uh}-\sigma_{uh})}$

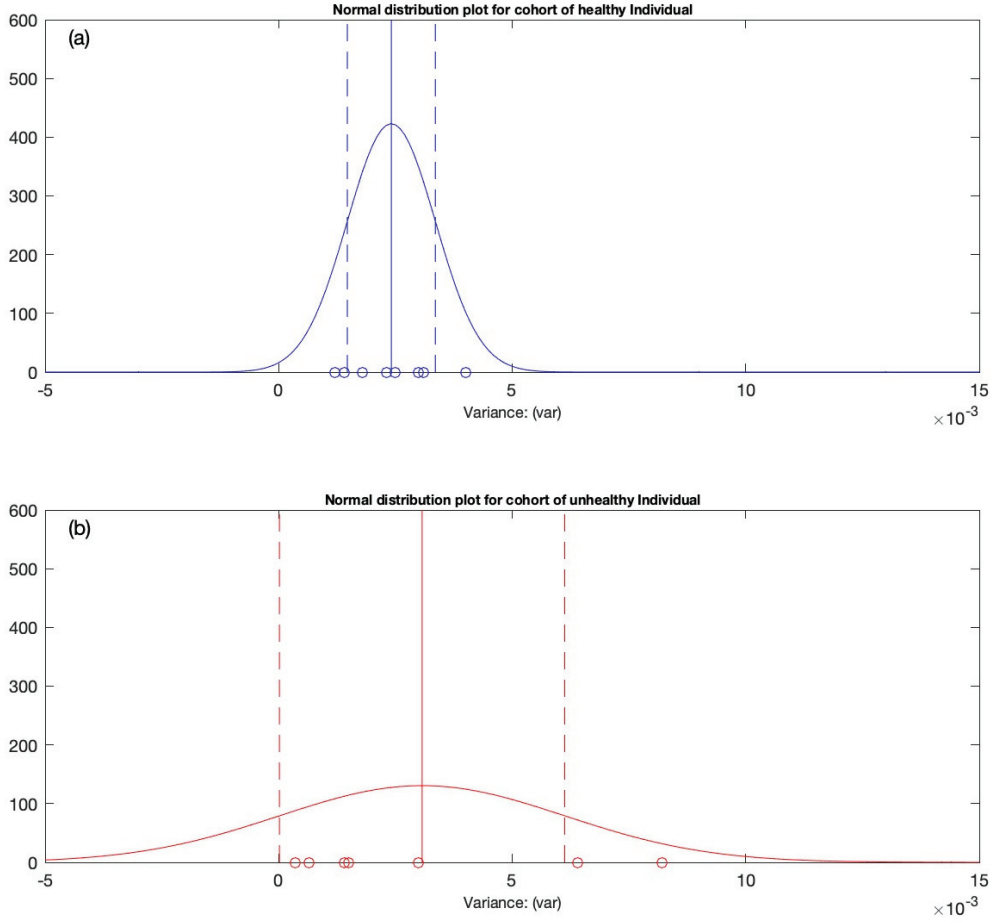


Figure 4. The distribution of individuals in each cohort is approximated by the Gaussian distribut-ion. (**a**) The Gaussian distribution plot computed for the cohort of healthy individuals. (**b**) The Gaussian distribution plot computed for the cohort of unhealthy individuals.

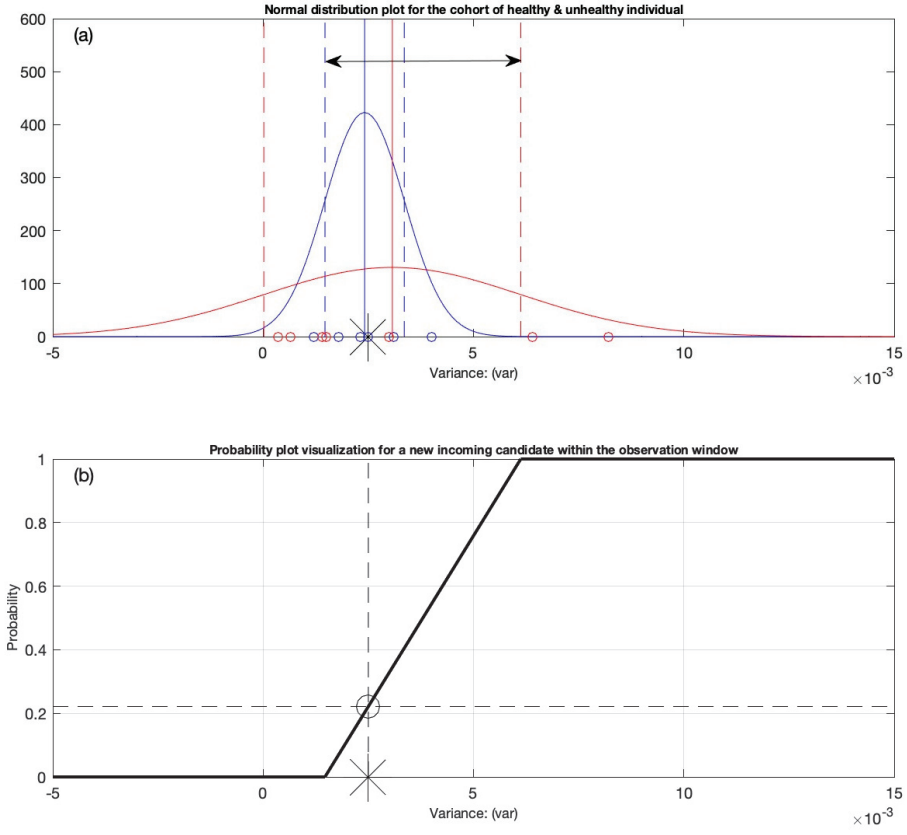


Figure 5. (**a**) The construction of the variation interval between the mean of the healthy distribution minus the standard deviation of the healthy distribution, and the mean of the unhealthy distribution plus the standard deviation of the unhealthy distribution. The variation interval is indicated by a double arrow and is used for the classification of a new incoming candidate. (**b**) The decision support system is depicted as a probability distribution plot with a candidate's likelihood (0.22) of having AF, indicated by an asterisk sign.

3. Results and Discussion

First, the sensitivity of the proposed architecture is analyzed to assess the three ECG parameters (JT interval, QRS interval, RR interval). As said earlier, the dataset comprises a cohort of eight healthy individuals and seven unhealthy individuals. For each candidate, the ECG signal is presented as a time series and is characterized as scalar vectors; x, y, and z. According to the architecture shown in Figure 1 and tabulated in Table 1, each of these scalar vectors is passed through the routine of perfect matrices of Lagrange differences. In other words, the three scalar vectors are transferred onto a trajectory of matrices. Figure 3 depicts the algebraic relationship between the vectors x, y, and z for one of the candidates, but the computations were performed for the entire cohort of 15 people (see Table 6).

The matrix trajectory for each individual in the cohort is translated into a scalar time series using the norm of the matrix, where mapping is represented as $\mathcal{F} : \mathbb{R}^{(3*3)} \rightarrow \mathbb{R}^1$. Finally, the signal is smoothed using internal and external smoothing techniques. The entire computation resulted in the final smooth singular scalar vector, which contains most of the original signal's information. Table 6 illustrates the outcomes of the matrix's sensitivity analysis (as mentioned in Section 2.4.3, our attention is entirely focused on the variance data for both cohorts). Up to this point, mathematical computations are performed for each

individual in the cohort in order to determine the algebraic relationship between the JT, QRS, and RR intervals according to the architecture proposed earlier. In the next section, the statistical computational techniques are discussed in detail.

3.1. Performing the Statistical Analysis

The statistical analysis is performed on the variance dataset presented Table 6. To begin with, the Anderson–Darling goodness-of-fit hypothesis test is conducted to determine whether the data values for the healthy and unhealthy individuals are of a normal distribution. Once the Anderson–Darling goodness-of-fit hypothesis test confirms the normal distribution of the dataset, the estimates of the mean (mu), standard deviation (sigma), and 95% confidence intervals are determined using the computer-generated built-in algorithms. Finally, the distribution graphs are created using the machine-provided probability density function. In Figure 4, the x-axis represents the distributed data (which are variance values produced from the proposed matrix architecture) for the cohort of healthy and unhealthy individual, whereas the y-axis shows the probability density of the dataset. To distinguish between the healthy and unhealthy cohorts, different color schemes are employed, such as blue for healthy individuals and red for unhealthy people. In the next section, the criterion for generating the observation interval is employed, which is used to classify the new incoming individual.

3.2. Generation of the Variation Interval

Figure 4 indicates that the distribution has an optimal fit for the cohort of healthy and unhealthy individuals. The observation interval is defined for classification purposes. The one sigma rule is applied on the normally distributed data shown in Figure 4. For this, the variation interval's boundaries must first be defined. The boundaries are indicated as right $\mu_{uh} + \sigma_{uh}$ and left $\mu_h - \sigma_h$ ends of the interval. This is indicated in Figure 5a by a double arrow linking the two boundaries. The new entrants into the observation window will be classified using this variation interval. After defining the observation interval, it is essential to build the probability distribution graphical representation and develop the semi-gauge indication tool. A set of statistical conditions is required for this purpose, as shown in Table 7. Before defining the conditions, a few variables must be defined. For example, the new candidate is denoted by C, and the variable IND (indicator) is introduced to represent the new person's location within the variation interval. The indices "h" and "uh" denote healthy individuals and unhealthy individuals, respectively. The right side of the variation interval is denoted as $\mu_{uh} + \sigma_{uh}$ and the left end is denoted as $\mu_h - \sigma_h$. The location of the new arriving candidate may be identified by the asterisk sign located along the x-axis depending on its ECG registration (see Figure 5). The conditions defined for the classification of the new incoming candidate are discussed below.

3.2.1. Condition 1

If the incoming candidate's ECG registering value is less than or equal to the left side of the variation interval, which is denoted as $C \leq \mu_h - \sigma_h$ (see Table 7), the indicator (IND) will be zero, and the person will fall towards the left side of the variation interval, indicating that he/she is a healthy candidate.

3.2.2. Condition 2

If the incoming candidate's ECG registering value is larger than or equal to the right side of the variation interval, which is referred to as $C \geq \mu_{uh} + \sigma_{uh}$ (see Table 7), then the person will fall towards the right side of the variation interval, indicating that he/she is a unhealthy candidate.

3.2.3. Condition 3

If the incoming candidate's value falls between the variation interval, which is codified as $u_h - \sigma_h \leq C \leq \mu_{uh} + \sigma_{uh}$ (see Table 7), the arriving candidate C falls in between the variation interval.

Because the effect of atrial fibrillation is not expressed during ECG registration, the distribution function is wide and the standard deviation is high. This implies that the subjects may have experienced asymptomatic atrial fibrillation. There could have been some "silent" AF candidates even among the healthy people. As a result, we created an algorithm that could aid in their classification.

To supplement the visual representation, we incorporate a semi-gauge indication tool for designing a warning system. This is accomplished by adopting a color scheme that is solely dependent on the conditions specified in Table 7. Depending on the ECG registering of the new candidate within the variation interval, the gauge arrow will point towards either green, yellow, or red (see Figures 6d and 7d). In the next section, the two test applicants are analyzed to determine where their ECG recordings fall after classification within the variation interval.

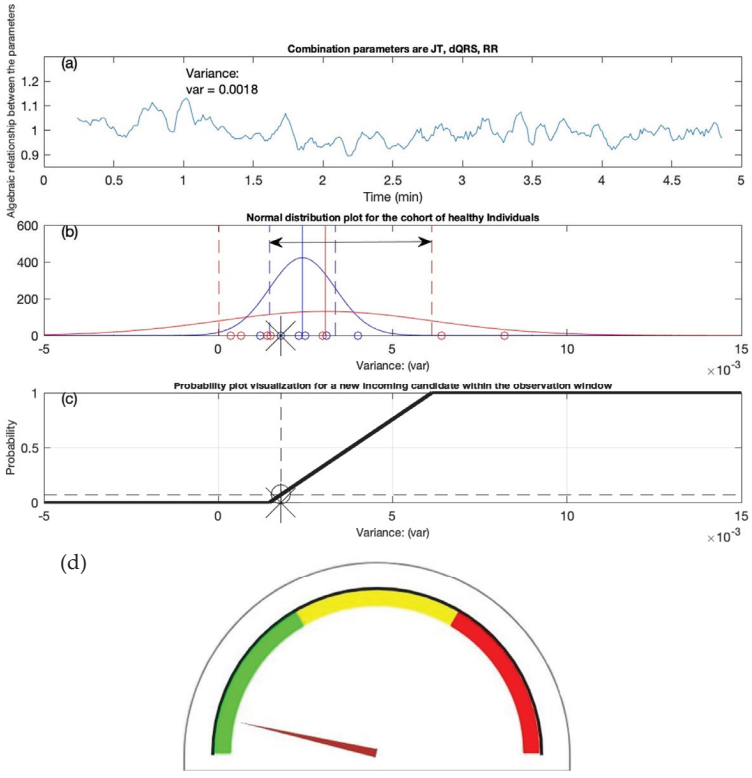


Figure 6. Test candidate #1. (**a**) The algebraic relation is shown for the combination of JT, RR, and QRS intervals. (**b**) The classification of the candidate is marked with an asterisk indicator. (**c**) The decision support system recommends the probability of the AF equal to 0.07. (**d**) The semi-gauge representation is exhibited for the candidate with the arrow of the gauge pointing towards green, indicating that the person is classified as a healthy individual.

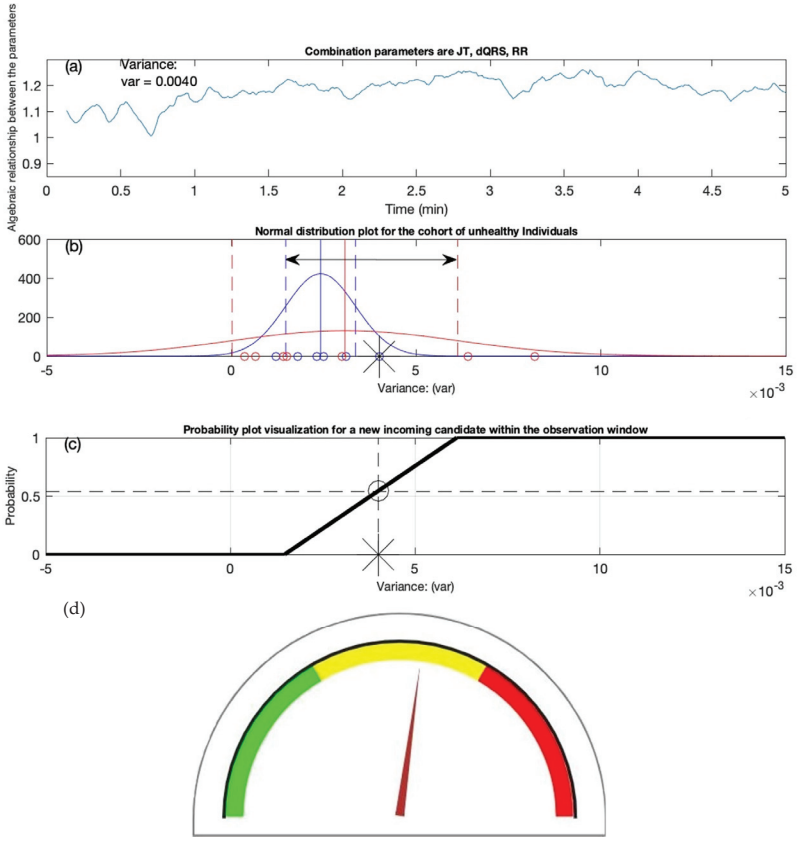


Figure 7. Test candidate #2. (**a**) The algebraic relation is shown for the combination of JT, QRS, and RR intervals. (**b**) The classification of the candidate is marked with an asterisk indicator. (**c**) The decision support system recommends the probability of the AF equal to 0.54. (**d**) The semi-gauge representation is exhibited for the candidate with the arrow of the gauge pointing towards yellow, indicating that the person is classified as an unhealthy individual.

3.3. First Test Candidate

Figure 6 depicts the example of one of the test candidates. In Figure 6a, the x-axis represents the time (minutes) and the y-axis represents the algebraic relationship between the three parameters (JT interval, QRS interval, RR interval). The computational techniques resulted in a variance value of 0.0018, as shown in Figure 6a. The resulting variance value is subjected to the classification conditions described in Table 7, and it is noted that the candidate appears to fall on the left side $C \leq \mu_h - \sigma_h$ of the variation interval. The location of the test candidate is indicated by the asterisk sign in Figure 6b. Furthermore, when the candidate's value is passed through the routine of the probability distribution graph, it is found that the candidate had the lowest likelihood (0.07) of having an atrial fibrillation prediction, as this can be seen in Figure 6c. In Figure 6c, the x-axis represents the variance values, and the y-axis represents the probability (0–1). Finally, by using a semi-gauge visualization tool, the color indication scheme is used to indicate where the gauge's arrow points out. It can be seen that arrow direction is towards the green, indicating that the individual is healthy (see Figure 6d). In the next section, we look at another test candidate's scenario.

3.4. Second Test Candidate

Figure 7 illustrates the example of another test candidate subjected to the proposed techniques described in the previous sections. In Figure 7a, the x-axis represents the time in minutes, and the y-axis shows the algebraic relationship between the parameters (JT interval, QRS interval, and RR interval). The mathematical computations generated a variance value of 0.0040 after passing through the proposed algorithm. When the variance value is compared to the data in Table 7, it is noticed that the test applicant falls into the category of healthy individuals. However, when the variance value is subjected to the classification conditions, it appears that the person is falling towards the extreme right $C \geq \mu_{uh} + \sigma_{uh}$ of the observation interval (see Table 7). The probability distribution graphical representation revealed that the test candidate has a likelihood of 0.54 to have the indication atrial fibrillation, indicated by the asterisk sign (Figure 7c). To demonstrate the severity of the warning, the semi-gauge arrow directed towards the yellow indicates the risk that the individual has an indication of having atrial fibrillation (see Figure 7d).

In the examples of test candidates presented above, it is seen that the candidates initially exhibit the behavior of being a healthy individual, but when their ECG recordings are passed through the entire computation techniques proposed, it is discovered that one of the candidates, despite having a normal ECG, may have indications of atrial fibrillation. As mentioned in the introduction section, some people may not exhibit symptoms of AF, which is known as asymptomatic or "silent" AF and has major clinical repercussions. This was discovered to be the case for our study's second test candidate.

AF is a pathology, and the identification of an arrhythmia on an ECG is necessary for the diagnosis. Unfortunately, prompt diagnosis of PV might be challenging due to the brief arrhythmic paroxysms and frequent asymptomatic state. To distinguish between paroxysmal and persistent AF in the ECG, signal-averaged parameters were not used when looking for unhealthy and healthy people. Electrocardiography signal-averaged parameters and changes in their relationships between AF patients and healthy people were used to identify significant ECG parameters. Due to the fact that this electrocardiography technology is being used to analyze ECG parameters in patients with AF for the first time, there is no way to compare the study's findings to those of previous investigations. Bright ECGs were seen throughout the examination, but after more research using this methodology, the trends in the disparities between the parameters and their links suggest that the signal-averaged electrocardiography parameters may be useful for screening in patients with risk factors for stroke, for whom a more thorough examination for asymptomatic AF is appropriate. The surface electrocardiogram (ECG) is the primary tool used for the clinical diagnosis of atrial fibrillation (AF), and AF is distinguished by the absence of a P wave due to the electrical activity being disorganized. The RR interval, which reflects the ventricular interbeat, was proposed as a significant biomarker for AF detection, despite the P wave's relatively low amplitude and challenging baseline that make its detection difficult [29]. Several publications investigate how the complexity of RR intervals changes as different cardiovascular disorders progress [30,31]. A suitable level of RR interval fluctuations within an organism denotes both healthy function and natural self-regulatory flexibility or resilience. JT interval and myocardial metabolic rate are related (when heart activity is the highest, JT interval is the shortest, and vice versa). The JT range, which is divided into the JTa interval (from point J to the peak of the T wave) and the Te interval, regulates certain electrophysiological events (from the T wave peak to the end of T) [32]. ECG leads with a shorter JT interval, showing earlier repolarization and quicker metabolic changes in particular cardiac areas. The JT interval was found to be the more significant indicator of AF risk than the QT interval, which has been linked to the development of AF. This suggests that ventricular repolarization rather than ventricular depolarization is a better predictor of future AF events (e.g., QRS duration). As previous reports have mainly concentrated on the AF risk associated with the entire QT interval, which includes components of both ventricular depolarization and repolarization, the results of this analysis offer significant insight into the pathophysiology of AF [29]. The interventricular synchronization aspects of the QRS duration parameter

may be linked to the intrinsic regulatory system of the heart. The QRS complex, which is a component of the heart's regulatory system, depicts the spreading of depolarization inside the ventricle and the synchronization of the spreading of depolarization between ventricles. Slower conduction is visible in the heart ventricle in the larger QRS complex. In normal conditions, the duration of the QRS complex can range from 80 to 120 ms. This index is sensitive to changes in sympathetic and parasympathetic nervous system tone. This parameter could be observed shortening in a healthy heart during sympathetic activation, as well as during load. It is prolonged in some heart diseases, such as ischemic heart disease. Patients with prolonged QRS had a higher prevalence of AF, according to the authors of a cross-sectional study involving 25,000 people with left ventricular dysfunction [29]. The majority of methods employed by physicians rely on the analysis of discrete values such as heart rate, blood pressure, or the duration of a single cardiac interval. Comparing ECG features in relation to one another can provide more clinical data, which is oftentimes more insightful than evaluating features alone.

4. Limitations

The current research had some limitations. For instance, the study sample was rather modest to start with. In order to evaluate atrial shape and potential connections with P-wave ECG parameters, participants were not subjected to cardiac imaging. Additionally, it makes sense to assess the dynamics of ECG parameter and interface values in addition to their instantaneous values when assessing each subject as a separate complex system.

5. Conclusions

In this novel work, it is hypothesized, demonstrated, and proved that increasing the dimensions of the perfect matrices of Lagrange differences from two to three enhances the sensitivity of the proposed architecture and provides sufficient grounds for analyzing the algebraic relationship between the ECG parameters. The idea of perfect matrices of Lagrange differences is exploited to determine the early detection of atrial fibrillation episodes using the ECG parameters JT, QRS, and RR. The computations were carried out on 8 healthy individuals and 7 unhealthy individuals. Despite the modest sample size, the analysis of this study showed solid evidence for defining the algebraic relationship between the parameters. Moreover, several mathematical computational tests were conducted to determine if the mean, median standard deviation, or variance could be employed for classification, and we identified that the variance parameter stood out the most. We decided to choose the variance as a computing tool for generating a baseline dataset. The observation window was then built using the one sigma rule. Furthermore, a set of conditions was also developed based on machine learning techniques to classify the new arriving candidate inside the variation interval. The idea for probability distribution graphs is proposed to indicate whether the new incoming person has the least or greatest likelihood of developing atrial fibrillation symptoms. In addition, a semi-gauge indicator tool was developed in order to construct a warning system based on the classification conditions. We evaluated two test candidates, both of whom initially displayed the behavior of a healthy individual, but when categorized using the classification, it was discovered that one of the test candidates stood out as having an indication of atrial fibrillation.

6. Future Work

Since the proposed method is sensitive to describing the algebraic relationship between the ECG parameters, changing the parameters' order may provide significantly different results. As a result, different ECG parameter combinations are encouraged for further research and experimentation. Nonetheless, this work has the potential to establish a reliable foundation for future research investigating ECG parameters using 3-by-3 perfect matrices of Lagrange differences. As the sample size of the proposed study was modest to begin with, it is encouraged to expand the sample size of the experiment and perform the computations.

Author Contributions: Conceptualization, A.V. and M.R.; methodology, M.R., A.V. and V.Z.; software, N.W.Q. and M.R.; validation, V.Z., M.R. and A.V.; formal analysis, N.W.Q.; investigation, V.Š.; resources, V.Z.; data curation, V.Š. and A.V.; writing—original draft preparation, N.W.Q. and V.Š.; writing—review and editing, A.V., M.R., N.W.Q. and V.Š.; visualization, M.R. and N.W.Q.; supervision, M.R.; project administration, A.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: The research was conducted with the approval of the LUHS Bioethics Committee (permission number BE-2-4 15 March 2016).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study. Written informed consent was obtained from the patients to publish this paper.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cunha, S.; Antunes, E.; Antoniou, S.; Tiago, S.; Relvas, R.; Fernandez-Llimós, F.; da Costa, F.A. Raising awareness and early detection of atrial fibrillation, an experience resorting to mobile technology centred on informed individuals. *Res. Soc. Adm. Pharm.* **2020**, *16*, 787–792. [CrossRef] [PubMed]
2. Odutayo, A.; Wong, C.X.; Hsiao, A.J.; Hopewell, S.; Altman, D.G.; Emdin, C.A. Atrial fibrillation and risks of cardiovascular disease, renal disease, and death: Systematic review and meta-analysis. *BMJ* **2016**, *354*, i4482. [CrossRef]
3. Rho, R.W.; Page, R.L. Asymptomatic atrial fibrillation. *Prog. Cardiovasc. Dis.* **2005**, *48*, 79–87. [CrossRef] [PubMed]
4. Savelieva, I.; Camm, A.J. Clinical relevance of silent atrial fibrillation: Prevalence, prognosis, quality of life, and management. *J. Interv. Card. Electrophysiol.* **2000**, *4*, 369–382. [CrossRef] [PubMed]
5. Camm, A.J.; Corbucci, G.; Padeletti, L. Usefulness of continuous electrocardiographic monitoring for atrial fibrillation. *Am. J. Cardiol.* **2012**, *110*, 270–276. [CrossRef]
6. Developed with the Special Contribution of the European Heart Rhythm Association (EHRA); Camm, A.J.; Kirchhof, P.; Lip, G.Y.; Schotten, U.; Savelieva, I.; Ernst, S.; Van Gelder, I.C.; Al-Attar, N.; Hindricks, G.; et al. Guidelines for the management of atrial fibrillation: The Task Force for the Management of Atrial Fibrillation of the European Society of Cardiology (ESC). *Eur. Heart J.* **2010**, *31*, 2369–2429.
7. Members, W.G.; Wann, L.S.; Curtis, A.B.; January, C.T.; Ellenbogen, K.A.; Lowe, J.E.; Estes, N.M., III; Page, R.L.; Ezekowitz, M.D.; Slotwiner, D.J. 2011 ACCF/AHA/HRS focused update on the management of patients with atrial fibrillation (updating the 2006 guideline) a report of the American College of Cardiology Foundation/American Heart Association Task Force on Practice Guidelines. *Circulation* **2011**, *123*, 104–123. [CrossRef]
8. Guidera, S.A.; Steinberg, J.S. The signal-averaged P wave duration: A rapid and noninvasive marker of risk of atrial fibrillation. *J. Am. Coll. Cardiol.* **1993**, *21*, 1645–1651. [CrossRef]
9. Mehta, S.; Lingayat, N.; Sanghvi, S. Detection and delineation of P and T waves in 12-lead electrocardiograms. *Expert Syst.* **2009**, *26*, 125–143. [CrossRef]
10. Qammar, N.W.; Orinaitė, U.; Šiaučiūnaitė, V.; Vainoras, A.; Šakalytė, G.; Ragulskis, M. The Complexity of the Arterial Blood Pressure Regulation during the Stress Test. *Diagnostics* **2022**, *12*, 1256. [CrossRef]
11. Ziaukas, P.; Alabdulgader, A.; Vainoras, A.; Navickas, Z.; Ragulskis, M. New approach for visualization of relationships between RR and JT intervals. *PLoS ONE* **2017**, *12*, e0174279. [CrossRef] [PubMed]
12. Houssein, E.H.; Kilany, M.; Hassanien, A.E. ECG signals classification: A review. *Int. J. Intell. Eng. Inform.* **2017**, *5*, 376–396. [CrossRef]
13. Malik, M.; Färbom, P.; Batchvarov, V.; Hnatkova, K.; Camm, A. Relation between QT and RR intervals is highly individual among healthy subjects: Implications for heart rate correction of the QT interval. *Heart* **2002**, *87*, 220–228. [CrossRef] [PubMed]
14. Lyon, A.; Mincholé, A.; Martínez, J.P.; Laguna, P.; Rodriguez, B. Computational techniques for ECG analysis and interpretation in light of their contribution to medical advances. *J. R. Soc. Interface* **2018**, *15*, 20170821. [CrossRef] [PubMed]
15. Ebrahimi, Z.; Loni, M.; Daneshtalab, M.; Gharehbaghi, A. A review on deep learning methods for ECG arrhythmia classification. *Expert Syst. Appl. X* **2020**, *7*, 100033. [CrossRef]
16. Gupta, V.; Mittal, M.; Mittal, V.; Saxena, N.K. A critical review of feature extraction techniques for ECG signal analysis. *J. Inst. Eng. Ser. B* **2021**, *102*, 1049–1060. [CrossRef]
17. Noujaim, S.F.; Lucca, E.; Muñoz, V.; Persaud, D.; Berenfeld, O.; Meijler, F.L.; Jalife, J. From mouse to whale: A universal scaling relation for the PR Interval of the electrocardiogram of mammals. *Circulation* **2004**, *110*, 2802–2808. [CrossRef]
18. Bonomini, M.P.; Arini, P.D.; Gonzalez, G.E.; Buchholz, B.; Valentinuzzi, M.E. The allometric model in chronic myocardial infarction. *Theor. Biol. Med. Model.* **2012**, *9*, 15. [CrossRef]

19. Captur, G.; Karperien, A.L.; Hughes, A.D.; Francis, D.P.; Moon, J.C. The fractal heart—Embracing mathematics in the cardiology clinic. *Nat. Rev. Cardiol.* **2017**, *14*, 56–64. [CrossRef]
20. Jafari, A. Sleep apnoea detection from ECG using features extracted from reconstructed phase space and frequency domain. *Biomed. Signal Process. Control.* **2013**, *8*, 551–558. [CrossRef]
21. Casaleggio, A.; Braiotta, S.; Corana, A. Study of the Lyapunov exponents of ECG signals from MIT-BIH database. In Proceedings of the Computers in Cardiology 1995, Vienna, Austria, 10–13 September 1995; pp. 697–700.
22. Übeyli, E.D. Detecting variabilities of ECG signals by Lyapunov exponents. *Neural Comput. Appl.* **2009**, *18*, 653–662. [CrossRef]
23. Casaleggio, A.; Corana, A.; Ridella, S. Correlation dimension estimation from electrocardiograms. *Chaos Solitons Fractals* **1995**, *5*, 713–726. [CrossRef]
24. Acharya, R.; Lim, C.; Joseph, P. Heart rate variability analysis using correlation dimension and detrended fluctuation analysis. *Itbm-Rbm* **2002**, *23*, 333–339. [CrossRef]
25. Fojt, O.; Holcik, J. Applying nonlinear dynamics to ECG signal processing. *IEEE Eng. Med. Biol. Mag.* **1998**, *17*, 96–101. [CrossRef] [PubMed]
26. Lee, J.-M.; Kim, D.-J.; Kim, I.-Y.; Park, K.-S.; Kim, S.I. Detrended fluctuation analysis of EEG in sleep apnea using MIT/BIH polysomnography data. *Comput. Biol. Med.* **2002**, *32*, 37–47. [CrossRef] [PubMed]
27. Houshyarifar, V.; Amirani, M.C. Early detection of sudden cardiac death using Poincaré plots and recurrence plot-based features from HRV signals. *Turk. J. Electr. Eng. Comput. Sci.* **2017**, *25*, 1541–1553. [CrossRef]
28. Saunoriene, L.; Siauciunaite, V.; Vainoras, A.; Bertasiute, V.; Navickas, Z.; Ragulskis, M. The characterization of the transit through the anaerobic threshold based on relationships between RR and QRS cardiac intervals. *PLoS ONE* **2019**, *14*, e0216938. [CrossRef]
29. Duan, J.; Wang, Q.; Zhang, B.; Liu, C.; Li, C.; Wang, L. Accurate detection of atrial fibrillation events with RR intervals from ECG signals. *PLoS ONE* **2022**, *17*, e0271596. [CrossRef]
30. de Godoy, M.F. Nonlinear analysis of heart rate variability: A comprehensive review. *J. Cardiol. Ther.* **2016**, *3*, 528–533. [CrossRef]
31. Skinner, J.E.; Anchin, J.M.; Weiss, D.N. Nonlinear analysis of the heartbeats in public patient ECGs using an automated PD2i algorithm for risk stratification of arrhythmic death. *Ther. Clin. Risk Manag.* **2008**, *4*, 549. [CrossRef]
32. Vainoras, A. Kardiovaskulinė sistema ir sportinė veikla. *Kardiovaskulinė Sist. Sport. Veikla. Vilnius* **1996**, *3*, 8.

diagnostics

MDPI

Article

Risk of Mortality Prediction Involving Time-Varying Covariates for Patients with Heart Failure Using Deep Learning

Keijiro Nakamura [1,*], Xue Zhou [2,*], Naohiko Sahara [1], Yasutake Toyoda [1], Yoshinari Enomoto [1], Hidehiko Hara [1], Mahito Noro [3], Kaoru Sugi [3], Ming Huang [2], Masao Moroi [1], Masato Nakamura [1] and Xin Zhu [4,*]

[1] Division of Cardiovascular Medicine, Toho University Ohashi Medical Center, Tokyo 153-8515, Japan
[2] Graduate School of Science and Technology, Nara Institute of Science and Technology, Ikoma 630-0192, Japan
[3] Division of Cardiovascular Medicine, Odawara Cardiovascular Hospital, Odawara 250-0873, Japan
[4] Graduate Department of Computer and Information Systems, The University of Aizu, Aizuwakamatsu 965-8580, Japan
* Correspondence: keijiro.nakamura@med.toho-u.ac.jp (K.N.); x.zhou@is.naist.jp (X.Z.); zhuxin@u-aizu.ac.jp (X.Z.); Tel.: +81-3-468-1251 (K.N.); +81-242-37-2771 (Xin Zhu)

Abstract: Heart failure (HF) is challenging public medical and healthcare systems. This study aimed to develop and validate a novel deep learning-based prognostic model to predict the risk of all-cause mortality for patients with HF. We also compared the performance of the proposed model with those of classical deep learning- and traditional statistical-based models. The present study enrolled 730 patients with HF hospitalized at Toho University Ohashi Medical Center between April 2016 and March 2020. A recurrent neural network-based model (RNNSurv) involving time-varying covariates was developed and validated. The proposed RNNSurv showed better prediction performance than those of a deep feed-forward neural network-based model (referred as "DeepSurv") and a multivariate Cox proportional hazard model in view of discrimination (C-index: 0.839 vs. 0.755 vs. 0.762, respectively), calibration (better fit with a 45-degree line), and ability of risk stratification, especially identifying patients with high risk of mortality. The proposed RNNSurv demonstrated an improved prediction performance in consideration of temporal information from time-varying covariates that could assist clinical decision-making. Additionally, this study found that significant risk and protective factors of mortality were specific to risk levels, highlighting the demand for an individual-specific clinical strategy instead of a uniform one for all patients.

Keywords: deep learning; heart failure; mortality; risk prediction; time-varying covariates

Citation: Nakamura, K.; Zhou, X.; Sahara, N.; Toyoda, Y.; Enomoto, Y.; Hara, H.; Noro, M.; Sugi, K.; Huang, M.; Moroi, M.; et al. Risk of Mortality Prediction Involving Time-Varying Covariates for Patients with Heart Failure Using Deep Learning. *Diagnostics* **2022**, *12*, 2947. https://doi.org/10.3390/diagnostics12122947

Academic Editors: Hiam Alquran and Wan Azani Mustafa

Received: 5 November 2022
Accepted: 18 November 2022
Published: 25 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

Heart failure (HF) has been significantly associated with mortality, especially among those older than 65 years [1,2]. Japan has a "super-aged" society, and the number of patients with HF was estimated to reach 1.3 million by 2030 [3]. Although a better survival prognosis in Japan than that in Europe had been observed, the length of hospital stays in Japan was approximately three times that in western countries [4]. Simultaneously, HF was a leading cause of hospitalization in Japan, bringing a heavy economic burden to society [5]. Considering these facts, a prognostic prediction model is expected to inform patients and their families about the course of the disease and guide physicians to more optimal treatment and management strategies such as allocation of medical resources in consideration of survival prognosis.

The multivariate Cox proportional hazard (CPH) model [6] is the most frequently used statistical-based method for risk prediction, but its performance is somewhat limited by the linearity of the functional form assumed in the partial hazard (exponential) part [7,8]. By contrast, deep learning-based prediction models can fit and learn more complicated (e.g., non-linearity by activation function) relationships between covariates and outcomes. A fully connected (FC) neural network-based survival prediction model, referred to "DeepSurv",

has been proposed by Katzman et al. [9]. It achieved better prediction performances than the CPH model for cancer survival [10] and cardiovascular risk [8], and displayed a workable performance on early triage of critically ill patients with COVID-19 [11].

Despite the widespread use of the CPH model and performance improvement achieved by DeepSurv, the two models only consider invariant covariates but have no ability to learn from time-varying covariates. Although a time-varying CPH model has been proposed to take time-varying covariates into consideration [12], this model deals with the time-dependent issue by simply considering the hazard at time t depending on the value of the time-varying covariate at that time but the regression effect or the weight of the covariate being constant or invariant. In essence, it is similar to a traditional CPH model. Therefore, the model cannot capture the temporal relationship implied in time-varying covariates. Furthermore, the time-varying CPH model requires the dataset to be pre-processed into a so-called "long" format, where each duration is represented in a start and stop view, which is monotonous. In this study, a recurrent neural network (RNN)-based model, referred to as "RNNSurv", was developed and temporally validated. Temporal validation (or narrow validation) was defined by Moons et al. as follows: "External validation may use participant data collected by the same investigators, typically using the same predictor and outcome definitions and measurements, but sampled from a later period" [13]. The proposed model could learn temporal features from time-varying covariates and fit nonlinear relationship to predict risk of all-cause mortality for patients with HF. RNNSurv demonstrated better performance than CPH and DeepSurv in terms of discrimination, calibration, and ability of risk stratification, especially identifying the patients with high risk of mortality. Patients' characteristics varied by risk level in view of the stratified risk groups (high risk and other (or low risk)). This study also suggested that significant risk and protective factors of mortality were required to be discussed specific to risk levels.

2. Materials and Methods

2.1. Study Design and Data Collection

The protocol for this retrospective study was prepared in accordance with the Declaration of Helsinki, and the study was approved by the Institutional Review Board and Ethics Committee of Ohashi Medical Center, School of Medicine, Toho University (No. H19031).

The data collection was the same as that explained in a previous study [14]. Medical records of patients with HF hospitalized at Toho University Ohashi Medical Center between April 2016 and March 2019 were reviewed by cardiovascular specialists, constituting a development dataset. Then, an independent validation dataset was built from patients admitted during the period of one year later. HF was diagnosed based on the following information in the medical records: clinical syndromes consisting of dyspnea, malaise, edema, or decreased exercise capacity due to the loss of compensation of cardiac pump function, secondary to structural or functional abnormalities of the heart [15].

Patient characteristics, laboratory data, and echocardiographic data were manually extracted from electronic medical records by a hired data collection staff. In addition, diagnosis procedure combination (DPC) data were also added to build a high-quality database in this study. DPC is made up of unified patient clinical information in Japan, patient information including disease name, surgical procedure, various stage classifications, medical expenses, and patients' medical practice is converted into electronic data, and medical insurance services such as claims are provided in Japan based on this DPC [16]. After verifying consistency between DPC data and medical records data, the database for this study was created. The database construction flow is shown in Supplementary Figure S1 and details of patients information are summarized in Supplementary Table S1.

Figure 1 shows the flowchart of the proposed prognostic prediction system. It mainly consists of predictors' selection, model development and validation, evaluation from discrimination, calibration, overall aspects [17], and risk stratification.

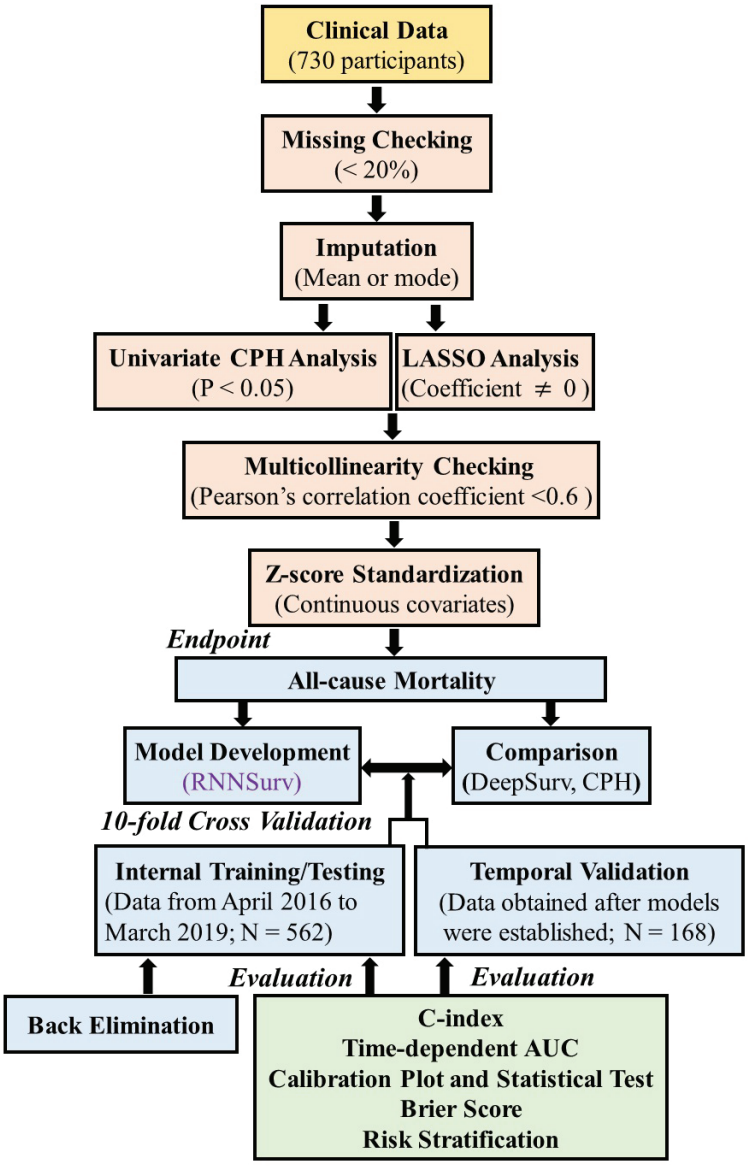


Figure 1. Overview of the proposed prognostic prediction system.

2.2. Outcome Definition

The endpoint was evaluated as all-cause mortality including both cardiovascular and non-cardiovascular mortality. Cardiovascular mortality was defined as death due to acute myocardial infarction, sudden cardiac death, HF, stroke, cardiovascular procedures, cardiovascular hemorrhage, and other cardiovascular deaths that could not be attributed to non-cardiac causes. Non-cardiovascular mortality was defined as any death that was not thought to be the result of a cardiovascular cause, such as infection and/or malignancy.

2.3. Preprocessing and Statistical Analysis

Candidate predictors were identified by preprocessing and statistical analysis, as previously described [18]. First, covariates that were missing in more than 20% of patients were excluded. As multiple imputation needs to be trained and is hard to converge with a relatively small sample size but a high dimension of variables, in addition, after checking the missing distribution, the missingness distribution seems balanced between death and survival groups as illustrated in Supplementary Table S2, and the continuous and categorical covariates with missing data were imputed using the simple imputation method [19]. Then, covariates significantly associated with mortality were identified by the univariate CPH model and least absolute shrinkage and selection operator regression [20] together. Multicollinearity was assessed and excluded by Pearson's correlation coefficient between pairs of continuous variables (setting 0.60 as a threshold) as was performed by Mesquita et al. [21]. Thereafter, the remaining covariates were identified as candidate predictors, where continuous predictors were Z-score standardized [22] when inputting to models. Available covariates were presented as mean ± standard deviation or frequency (percentage) in view of the covariates' type. For group comparison, a two-tailed, unpaired Student's *t*-test was used to assess the difference of continuous covariates with normal distribution, and a Mann–Whitney test was employed for skewed continuous covariates, where normality was tested using the Shapiro–Wilk test. Categorical data were compared using Pearson's Chi-square test or Fisher's exact test, as appropriate. A p value < 0.05 was considered statistically significant. Data preprocessing was performed using Python (Python Software Foundation, Beaverton, OR, USA; version 3.7.7), and statistical analysis was conducted using R (R Foundation for Statistical Computing, Vienna, Austria; version x64 3.6.0).

2.4. Model Development and Validation

The proposed RNNSurv's input includes two categories of candidate predictors: invariant (collected once) and time-varying (collected at both admission and discharge) with different input strategies. RNNSurv consists of two branches as shown in Supplementary Figure S2. The first is an RNN layer with inputs of time-step dynamic predictors (at admission and discharge). The second branch is an FC layer with inputs of invariant predictors. Each FC layer connects with a ReLu activation function and batch normalization, and is trained with a dropout strategy to reduce the risk of overfitting. Then, outputs of the two branches are concentrated as a vector and input to the output layer. The detailed parameters of the RNNSurv are illustrated in Supplementary Table S3 after trial and error. RNNSurv was trained with an Adam optimizer, a learning rate of 0.05, and a training batch size of 64. Early termination was activated if the loss function did not decrease after 20 consecutive epochs; otherwise, the training process continued to 1000 epochs. The model was developed and internally validated by a 10-fold cross validation strategy on a development dataset, which consisted of data from patients admitted between 7 April 2016 and 17 March 2019. Simultaneously, back-elimination was employed to further improve performance and to reduce required predictors. In detail, the candidate predictor that had the greatest negative impact on testing performance (C-index) was deleted, and the process was repeated until no further predictor deletion would improve testing performance. After the model was developed, it was validated on an independent dataset, of which data were obtained in the same way as the development dataset but collected from patients admitted between 20 March 2019 and 16 March 2020. The detailed development and validation processes are shown in Figure 2.

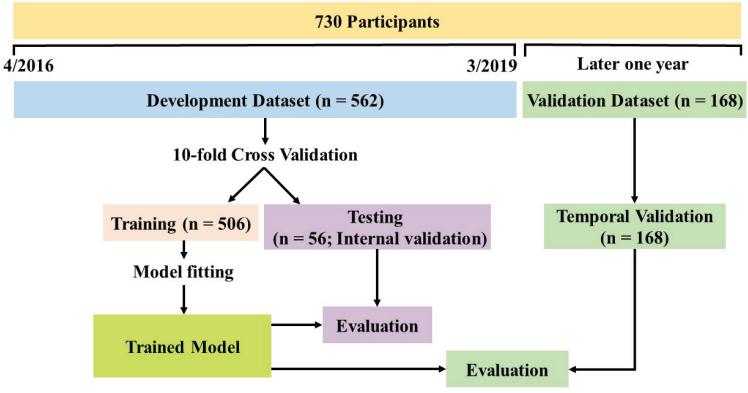


Figure 2. Overview of model development and validation.

In addition, a DeepSurv model [9] and a multivariate CPH model [6] were also developed and temporally validated for performance comparison. To avoid overfitting, the DeepSurv model employed a shallow structure with two FC layers. The first FC layer was followed by a ReLU activation function and batch normalization and trained using the dropout strategy as shown in Supplementary Figure S3. Its training configurations were the same as those of RNNSurv. The CPH model employed a penalty with 0.05 to the size of the coefficients during regression. This improved the stability of estimates and avoided convergence failure. The initial inputs, i.e., the candidate predictors obtained above, of the three models were the same. Because the DeepSurv and CPH models cannot capture temporal features by themselves, a covariate separately measured at admission and discharge would be regarded as two independent predictors. For example, heart rate (HR) was measured at both admission (HR_A) and discharge (HR_D), resulting in two predictors: HR_A and HR_D for DeepSurv and CPH. However, as RNNSurv could capture temporal information among time steps, it would regard the HR as one predictor but with two time steps (admission and discharge). After back-elimination, the optimal predictor sets may vary with models due to differences in algorithms.

2.5. Performance Evaluation

Discrimination performance of models was evaluated using Harrell's C-index [23] and cumulative/dynamic time-dependent area under the receiver operating characteristic curve (AUC) [24]. Higher C-index and time-dependent AUC values indicate better discrimination. Calibration was evaluated at 1 and 2 years during follow-up using a calibration plot [25] and a Hosmer–Lemeshow test simultaneously. If the calibration plot lies on a 45-degree line, the calibration performance is perfect. Furthermore, the Brier score [26] was used for overall performance evaluation, where a score closer to 0 indicates better performance. Apart from the abovementioned measures, the models' prediction performance was further evaluated in view of their risk stratification ability. The 20% of patients with lower predicted survival probabilities were considered at high risk of mortality [8]. Then, survival curves of high-risk and the other 80% patients (non-high-risk or low-risk) were checked using a Kaplan–Meier estimator and a log-rank test.

3. Results

3.1. Characteristics of Patients and Candidate Predictors

In total, 562 patients were included in the development dataset. The average age was 78 years and 45.7% of them were female. During follow-up of 30.9 ± 13.7 months, 81 (14.4%) patients died. Patients who died were older, with more prior hospital admissions, had a heavier burden on independence in daily life for the elderly with cognitive impairment (IDL) and activities of daily living (ADL), and were more likely to have chronic kidney

disease (estimated glomerular filtration rate (eGFR) < 60 mL/min/1.73 m^2), compared with patients who survived, as illustrated in Supplementary Table S1. The validation dataset included 168 patients of whom 28 (16.7%) patients died. Most characteristics were comparable between patients in the development and validation datasets, but patients in the validation dataset had lower body mass index (BMI), CHA_2DS_2-VAS_c score, and right ventricular systolic pressure (RVSP). The mortality between the development and validation datasets had no significant difference (14.4% vs. 16.7%, respectively, $p = 0.551$). The detailed comparison is illustrated in Supplementary Table S4.

After candidate predictor selection, age, length of stay, prior admission times, ischemic heart disease (IHD), New York Heart Association (NYHA) at discharge, frailty class, $CHADS_2$, left ventricular ejection fraction (LVEF), mitral regurgitation (MR), tricuspid regurgitation (TR), RVSP, angiotensin-converting enzyme inhibitor/angiotensin receptor blocker (ACEi/ARB), NT-proB-type natriuretic peptide (NT-proBNP), hemoglobin (HGB), and time-varying covariates including eGFR, systolic blood pressure (SBP), diastolic blood pressure (DBP), HR, and low ADL were identified as initial candidate predictors for all-cause mortality, as illustrated in Supplementary Table S5.

3.2. Discrimination Evaluation

Candidate predictors were used to build initial models (RNNSurv, DeepSurv, and CPH). After back-elimination, the final predictors and C-index were obtained, as illustrated in Table 1. Models were internally validated by 10-fold cross validation and then temporally validated. Once the final used predictors were determined by back-elimination, models were trained on all of the data of the development dataset and then validated on the validation dataset. In 10-fold cross validation, the proposed RNNSurv achieved a 0.807 ± 0.057 (mean $\pm$ standard derivation) testing C-index on the development dataset and a C-index of 0.809 ± 0.022 on the validation dataset, whereas those of the DeepSurv model were 0.783 ± 0.066 and 0.762 ± 0.022, respectively, and those of the CPH model were 0.754 ± 0.069 and 0.764 ± 0.002, respectively. An ANOVA test with the Tukey method confirmed that the C-index of RNNSurv was significantly different from that of CPH and DeepSurv (RNNSurv vs. DeepSurv: $p < 0.001$; RNNSurv vs. CPH: $p < 0.001$). When RNNSurv, DeepSurv, and CPH were trained on all of the data of the development dataset, the validation C-indexes were 0.839, 0.755, and 0.764, respectively. In terms of predictors, the three models perform both generalizability and specificity. For example, all three models take age, length of stay, NT-proBNP, etc., as the predictors; while some predictors are model-specific, such as TR and IDL.

Table 1. C-index and predictors of models.

Model	Model Development (10-Fold Cross Validation)	Validation	All of the Data	Predictors
RNNSurv	Train: 0.820 ± 0.022 Test: 0.807 ± 0.057	0.809 ± 0.022 (RNNSuev vs. DeepSurv: p value < 0.001; (RNNSuev vs. CPH: p value < 0.001;)	Development: 0.890 Validation: 0.839	Invariant predictors: Age, length of stay, IHD, NYHA at discharge, frailty, LVEF, TR, ACEi/ARB, NT-proBNP, HGB; Time-varying predictors: eGFR, SBP, DBP, HR, low ADL
DeepSurv	Train: 0.818 ± 0.033 Test: 0.783 ± 0.066	0.762 ± 0.022	Development: 0.872 Validation: 0.755	Age, length of stay, IHD, LVEF, RVSP, HGB, ACEi/ARB, IDL, NT-proBNP, (NYHA, eGFR, SBP, DBP, HR, low ADL) at discharge
CPH	Train: 0.768 ± 0.010 Test: 0.754 ± 0.069	0.764 ± 0.002	Development: 0.767 Validation: 0.762	Length of stay, IDL, ACEi/ARB, NT-proBNP, (NYHA, eGFR, SBP, low ADL) at discharge

In addition to the C-index, the time-dependent AUCs of every 3 months are visualized in Figure 3. The upper boundary was set at 4 years for the development dataset and 2 years for the validation dataset to avoid high censoring rates at later follow-up times.

The proposed model outperformed DeepSurv and CPH on both the development (Figure 3a) and validation datasets (Figure 3b).

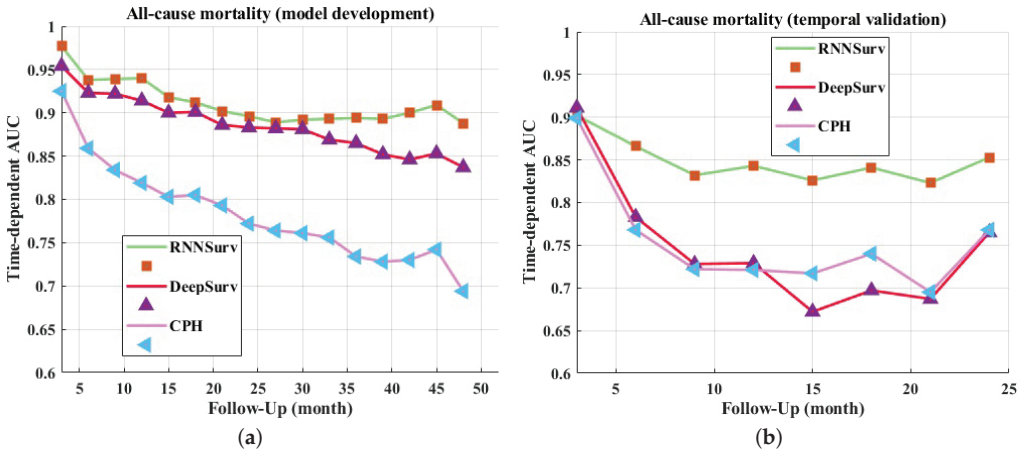


Figure 3. Time-dependent AUCs of models on development dataset (**a**) and validation dataset (**b**).

3.3. Calibration Evaluation

The calibration plots of the three models are shown in Figure 4. All three models fit well to a 45-degree line on the development dataset, indicating almost perfect calibration ability. On the validation dataset, the calibration plot of the proposed model seemed better than those of the other models. However, the three models showed a worse calibration ability on the validation dataset than the development dataset and the reason for this phenomenon has been explained by Park et al. [17]—if the sample size is small, few individuals will be included in subgroups. In this case, calibration analysis in view of the calibration plot and the statistical test are not robust or powerful enough.

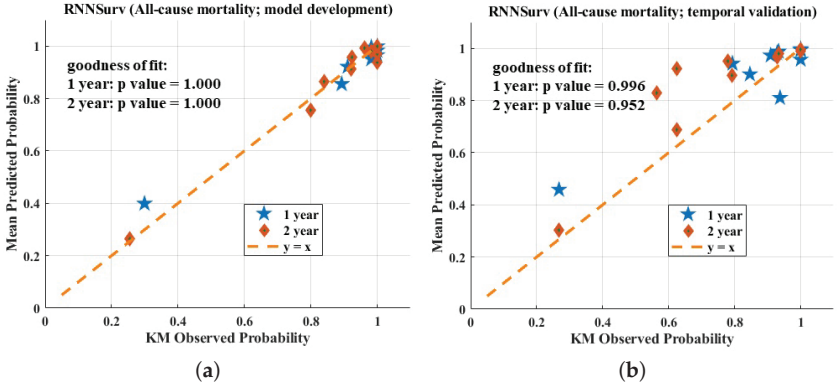


Figure 4. *Cont.*

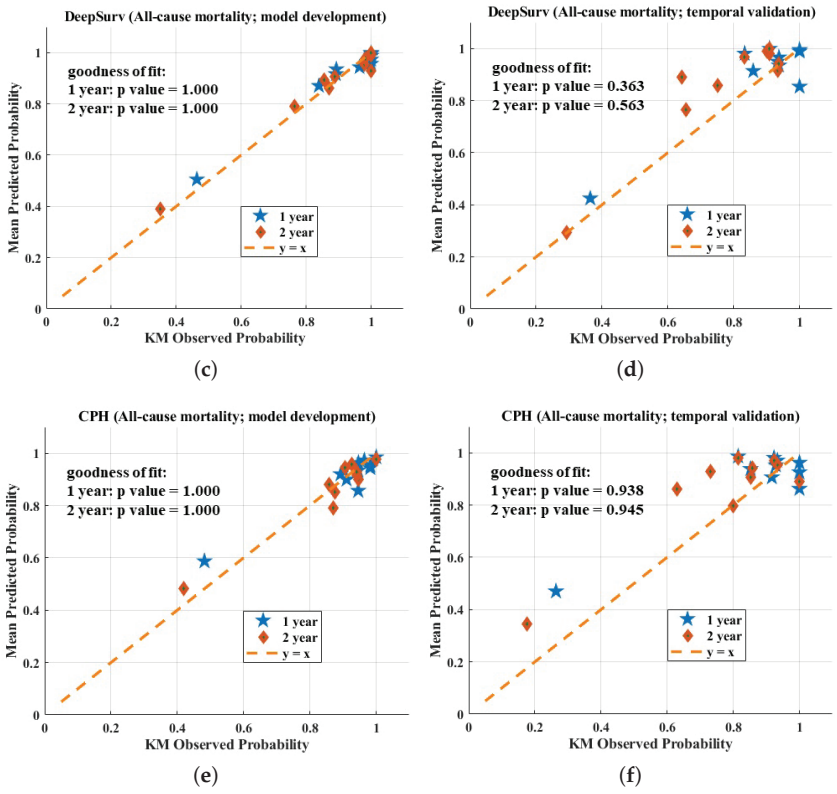


Figure 4. Calibration evaluation of RNNSurve (**a**,**b**), DeepSurve (**c**,**d**), and CPH model (**e**,**f**).

3.4. Overall Evaluation

The Brier score of the proposed model was smaller than those of DeepSurv and CPH on both the development and the validation datasets, as shown in Figure 5, indicating a better overall performance. In detail, the mean ± standard deviation of Brier scores of RNNSurv, DeepSurv, and CPH were 0.054 ± 0.017, 0.066 ± 0.020, and 0.080 ± 0.025 on the development dataset, and 0.099 ± 0.039, 0.107 ± 0.039, and 0.112 ± 0.040 on the validation dataset, respectively.

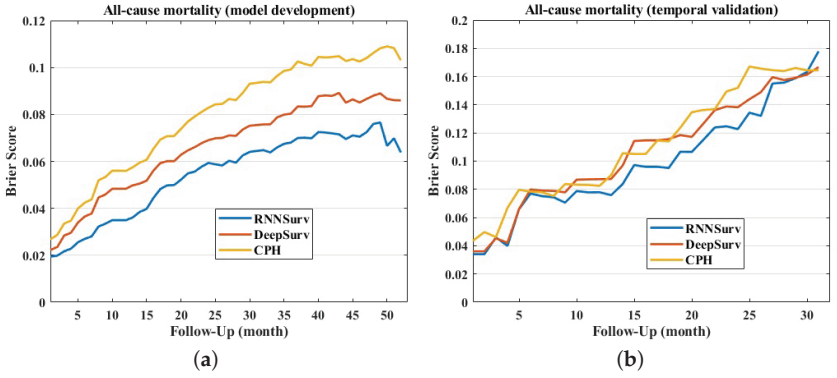


Figure 5. Brier scores of models on development dataset (**a**) and validation dataset (**b**).

3.5. Risk Stratification

The prediction performance was further evaluated in view of the ability of risk stratification. Kaplan–Meier survival curves of high-risk and other (non-high-risk or low-risk) patients stratified by the three models are visualized in Figure 6. Compared with DeepSurv (Figure 6c,d) and CPH (Figure 6e,f), the survival curves of high-risk patients stratified by the proposed RNNSurv (Figure 6a,b) had more obvious differences from other patients on both the development and validation datasets, indicating a better ability of risk stratification. Mortality in the high-risk patient group and other or low-risk patient group stratified by RNNSurv had a significant difference (51.8% vs. 5.1%, $p < 0.001$ on the development dataset; 48.5% vs. 8.9%, $p < 0.001$ on the validation dataset). Patients with high risk of all-cause mortality were older, were more often female, had a heavier burden on their medical history in terms of IHD and vascular disease (VD), had a longer length of stay and severe HF symptoms assessed by NYHA and heavier burden on daily activities, and were more likely to have chronic kidney diseases compared with low-risk patients. Details are illustrated in Supplementary Table S6.

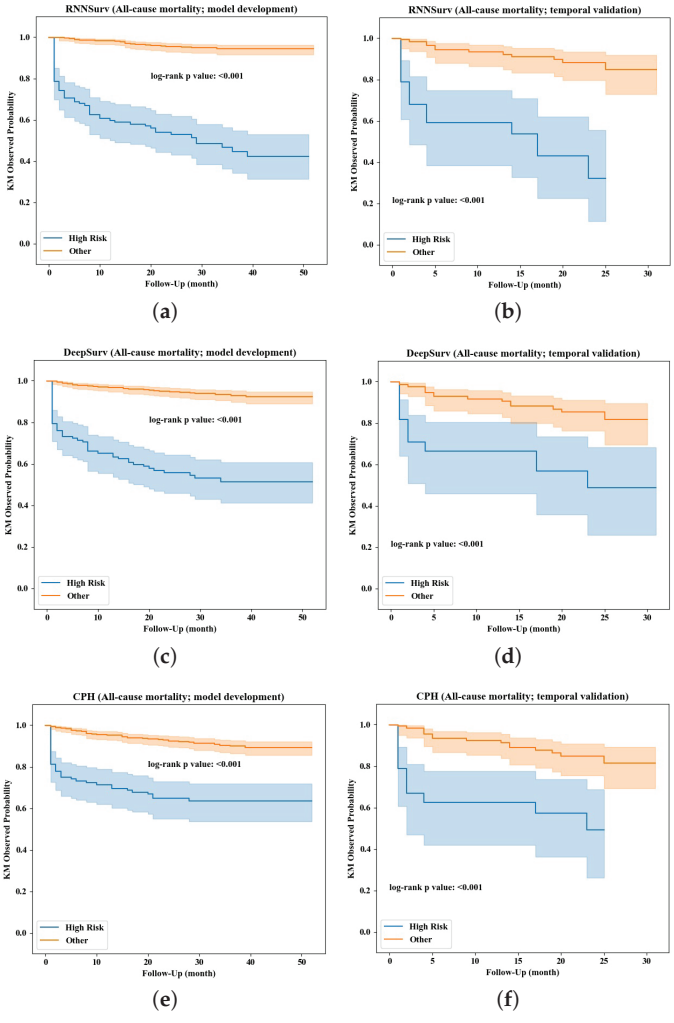


Figure 6. Risk stratification by the RNNSurve (**a**,**b**), DeepSurve (**c**,**d**), and CPH (**e**,**f**) models.

Additionally, significant risk and protective factors varied across all patients and high-risk patients only. Based on the results of univariate CPH analysis, in the patient group with a high risk of all-cause mortality, NYHA class (hazard ratio: 1.73 (95%CI 1.21–2.46), $p = 0.002$), low ADL (hazard ratio: 2.38 (95%CI 1.29–4.37), $p = 0.005$), creatinine (hazard ratio: 1.12 (95%CI 1.03–1.22), $p = 0.011$), and HR (hazard ratio: 1.02 (95%CI 1.00–1.03), $p = 0.049$) were significant risk factors, while protective factors were female (hazard ratio: 0.53 (95%CI 0.31–0.88), $p = 0.015$), direct oral anticoagulants/Warfarin usage (DOACWFuse) (hazard ratio: 0.47 (95%CI 0.27–0.83), $p = 0.009$), SBP (hazard ratio: 0.98 (95%CI 0.96–0.99), $p < 0.001$), and DBP (hazard ratio: 0.97 (95%CI 0.95–0.998), $p = 0.031$) (see Supplementary Table S7). However, in addition to the above significant factors, age (hazard ratio: 1.04 (95%CI 1.02–1.06), $p < 0.001$), CHA_2DS_2-VAS_c (hazard ratio: 1.15 (95%CI 1.02–1.30), $p = 0.024$), eGFR (hazard ratio: 0.98 (95%CI 0.96–0.99) at admission, $p < 0.001$; 0.96 (95%CI 0.94–0.97) at discharge, $p < 0.001$), etc., were also significantly associated with mortality in all patient groups (Supplementary Table S5).

4. Discussion

In this study, an RNN-based prognostic prediction model for patients with HF was developed and temporally validated. This model can consider time-varying covariates, which imply the course of disease (improving or worsening), and learn temporal information from these covariates for decision-making, while DeepSurv and CPH have no such ability. The proposed model achieved better prediction performance than DeepSurv and CPH in view of discrimination, calibration, and ability of risk stratification.

The multivariate CPH model is the most-used survival model to fit the relationship between patients' covariates and outcomes [27]. However, its prediction ability may be limited by linear assumptions. Recently, Bazoukis et al. concluded that machine learning methods were important and effective for diagnosis, management, and prediction of outcomes in HF patients [28]. In addition, instead of expensive imaging measurement, Liu et al. reported a cheap method to estimate LVEF achieved by machine learning algorithms [29]. Coincidentally, a modern CPH feed-forward deep neural network model (DeepSurv) was proposed as an alternative method for survival or risk prediction [8,9]. The neural network-based model could learn and capture more complex and abstract features and feature combinations from input covariates by nonlinear activation functions in layers. Both CPH and DeepSurv assume that covariates or predictors are invariant. However, some covariates are time-varying due to treatments such as polypharmacy in real-world situations, which are informative and reflect the course of disease. These temporal features should be considered in prediction models. Conversely, the proposed RNN-based prediction model could "remember" previous information and use it for current decision-making; therefore, it naturally accepts and considers time-varying predictors by serially learning.

The proposed model identifies and uses predictors that are easily obtained by routine examination and are measured at admission and discharge. Most of these predictors are previously reported prognostic markers. Specifically, age, NYHA, LVEF, ACEi/ARB, HGB, and SBP were previously identified as predictors in the Seattle Heart Failure Model to predict 1-, 2-, and 3-year survival rates for patients with HF [30]. NT-proBNP was used as one of the predictors for clinical outcomes in patients with HFrEF [31]. Although eGFR was widely reported as strongly associated with outcome in HF, few studies used it as a predictor [32]. In our study, eGFR was identified as a predictor.

The proposed model has a considerable ability of risk stratification, especially in identifying patients who have high risk of mortality. Patients with high risk of all-cause mortality are much older. As reported by Gustafsson et al., elderly patients hospitalized with HF had a very critical prognosis [33]. NYHA classification is generally used to estimate HF symptoms, and it was reported to be a significant predictor of all-cause mortality [34]. In our study, NYHA was identified as a predictor of all-cause mortality as well, and a greater proportion of patients with high risk of all-cause mortality had high NYHA (e.g., NYHA III, IV). Shamagian et al. defined patients with eGFR < 30 mL/min/1.73 m^2 as having severe

renal failure, which was reported to be a strong predictor of mortality, and these patients had significantly poor survival rates [35]. In our study, eGFR was a significant predictor for all-cause mortality. The proportion of patients with eGFR < 30 mL/min/1.73 m^2 in the high-risk group was about two or three times that of the non-high-risk group. It was reported that higher NT-proBNP levels were associated with higher incidences of all-cause mortality [36], which is consistent with our study. Dunlay et al. reported that mortality increased with increasing ADL difficulty [37] and low admission ADL was reported to be associated with cardiovascular mortality [38]. In this study, low ADL was a predictor for all-cause mortality and patients with high risk of all-cause mortality more often had low ADL at both admission and discharge. A systematic review and meta-analysis conducted by Lakhan et al. concluded that high C-reactive protein (CRP) was significantly associated with an increased risk of all-cause mortality and sometimes related to a greater risk of long-term adverse cardiovascular outcomes for patients with HF with preserved ejection fraction [39]. In this study with mixed cases, patients with high risk of all-cause mortality had significantly higher CRP than patients in the low-risk group.

Notably, significant risk and protective factors affecting survival or mortality are risk level-specific. Therefore, risk stratification or identifying patients at high risk are crucial to guide precision medicine and clinical strategies such as individually management and treatment. Based on the possible survival duration or probability of mortality after discharge, it is helpful in determining if or when the prevention treatment for high-risk patients should be introduced and determining if an early discharge could be conducted for low-risk patients to save costs and medical resources. Moreover, covariate value collected at discharge is more likely to perform a significant risk or protective effect on prognosis than the one collected at admission, indicating that covariates collected at discharge are more strongly associated with prognostic outcomes than ones collected at admission, and should be given more attention.

Limitations of the presented study are noted as follows. First, the proposed model was developed on a single-center dataset. Although a temporal validation, which is considered intermediate between internal and external validation, was conducted based on the TRIPOD report [13], an external validation is needed, such as conducting the validation on multiple centers and populations for further performance confirmation. Second, some predictors used in previously developed prediction models were not recorded, such as lymphocyte counts, uric acid, and sodium levels, which were used in the Seattle Heart Failure Model; therefore, performance comparison with these models is limited. Third, some covariates have missing data in varying proportions (>20%), limiting their inclusion in this study and prediction ability. Finally, the confirmation of event and survival information were determined by DPC data and medical records only, and thus it is possible that the event after discharge has not been fully confirmed or captured.

5. Conclusions

The proposed RNN-based risk prediction model for patients with HF demonstrated better performance than conventional statistical-based (CPH) and classical deep learning-based (DeepSurv) models. The model considers and accepts time-varying covariates, and in this way, it captures temporal information existing in a real-world situation. Risk and protective factors that are significantly associated with mortality are specific to risk levels, highlighting the demand for individual-specific clinical strategy instead of a uniform one-for-all strategy.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/diagnostics12122947/s1, Table S1. Characteristics of Patients with Heart Failure for Model Development; Table S2. Comparison of Covariate Missing between Death and Survival Patient groups in Development Dataset; Table S3. Parameters of RNNSurv; Table S4. Comparison Between Patients in Development and Validation Datasets; Table S5. Candidate Predictors for All-cause Mortality; Table S6. Comparison Between High- and Low-Risk Patients Stratified by RNNSurv for All-Cause Mortality; Table S7. Univariate analysis for High-risk group (All-

cause Mortality); Figure S1. Database construction flowchart; Figure S2. Structure of the proposed RNNSurv; Figure S3. Structure of DeepSurv.

Author Contributions: Conceptualization, K.N., X.Z. (Xue Zhou) and X.Z. (Xin Zhu); methodology, X.Z. (Xue Zhou), M.H. and X.Z. (Xin Zhu); software, X.Z. (Xue Zhou), M.H. and X.Z. (Xin Zhu); validation, K.N., X.Z. (Xue Zhou) and X.Z. (Xin Zhu); formal analysis, K.N., X.Z. (Xue Zhou) and X.Z. (Xin Zhu); investigation, K.N., X.Z. (Xue Zhou) and X.Z. (Xin Zhu); resources, K.N., N.S., Y.T., Y.E., H.H., M.N. (Mahito Noro), K.S., M.M. and M.N. (Masato Nakamura); data curation, K.N., X.Z. (Xue Zhou), N.S., Y.T., Y.E., H.H., M.N. (Mahito Noro), K.S., M.H., M.M., M.N. (Masato Nakamura) and X.Z. (Xin Zhu); writing—original draft preparation, K.N., X.Z. (Xue Zhou) and X.Z. (Xin Zhu); writing—review and editing, K.N., X.Z. (Xue Zhou), N.S., Y.T., Y.E., H.H., M.N. (Mahito Noro), K.S., M.H., M.M., M.N. (Masato Nakamura) and X.Z. (Xin Zhu); visualization, X.Z. (Xue Zhou) and X.Z. (Xin Zhu); supervision, K.N. and X.Z. (Xin Zhu); project administration, K.N. and X.Z. (Xin Zhu); funding acquisition, K.N. and X.Z. (Xin Zhu). All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Japan Society for the Promotion of Science Kakenhi Basic Research Fund C 21K10287 and Competitive Research Fund of The University of Aizu, 2022-P-12.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Institutional Review Board and Ethics Committee of Ohashi Medical Center, School of Medicine, Toho University (No. H19031).

Informed Consent Statement: Patient consent was waived for this study in accordance with the national legislation and the institutional requirements.

Data Availability Statement: The data used in this article are not readily available because of the restrictions of the Institutional Review Board. Requests to access the datasets should be directed to the corresponding authors.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ACEi/ARB	Angiotensin-converting enzyme inhibitor/angiotensin receptor blocker
ADL	Activities of daily living
AUC	Area under receiver operating characteristic curve
BMI	Body mass index
CPH	Cox proportional hazard
CRP	C-reactive protein
DBP	Diastolic blood pressure
DPC	Diagnosis procedure combination
eGFR	Estimated glomerular filtration rate
FC	Fully connected
HF	Heart failure
HGB	Hemoglobin
HR	Heart rate
HR_A	HR measured at admission
HR_A	HR measured at discharge
IHD	Ischemic heart disease
IDL	Independence in daily life for the elderly with cognitive impairment
LVEF	Left ventricular ejection fraction
MR	Mitral regurgitation
NT-proBNP	NT-proB-type natriuretic peptide
NYHA	New York Heart Association
RNN	Recurrent neural network
RVSP	Right ventricular systolic pressure
SBP	Systolic blood pressure
TR	Tricuspid regurgitation
VD	Vfascular disease

References

1. Roger, V.L. Epidemiology of heart failure: A contemporary perspective. *Circ. Res.* **2021**, *128*, 1421–1434. [CrossRef]
2. McMurray, J.J.; Stewart, S. Epidemiology, aetiology, and prognosis of heart failure. *Heart* **2000**, *83*, 596–602. [CrossRef]
3. Okura, Y.; Ramadan, M.M.; Ohno, Y.; Mitsuma, W.; Tanaka, K.; Ito, M.; Suzuki, K.; Tanabe, N.; Kodama, M.; Aizawa, Y. Impending epidemic: Future projection of heart failure in Japan to the year 2055. *Circ. J.* **2008**, *72*, 489–491. [CrossRef]
4. Hamaguchi, S.; Kinugawa, S.; Tsuchihashi-Makaya, M.; Goto, D.; Yamada, S.; Yokoshiki, H.; Tsutsui, H. Characteristics, management, and outcomes for patients during hospitalization due to worsening heart failure—A report from the Japanese Cardiac Registry of Heart Failure in Cardiology (JCARE-CARD). *J. Cardiol.* **2013**, *62*, 95–101. [CrossRef]
5. Kanaoka, K.; Okayama, S.; Nakai, M.; Sumita, Y.; Nishimura, K.; Kawakami, R.; Okura, H.; Miyamoto, Y.; Yasuda, S.; Tsutsui, H.; et al. Hospitalization costs for patients with acute congestive heart failure in Japan. *Circ. J.* **2019**, *83*, 1025–1031. [CrossRef]
6. Cox, D.R. Regression models and life-tables. *J. R. Stat. Soc. Ser. B (Methodol.)* **1972**, *34*, 187–220. [CrossRef]
7. Kim, D.W.; Lee, S.; Kwon, S.; Nam, W.; Cha, I.; Kim, H.J. Deep learning-based survival prediction of oral cancer patients. *Sci. Rep.* **2019**, *9*, 6994. [CrossRef]
8. Hathaway, Q.A.; Yanamala, N.; Budoff, M.J.; Sengupta, P.P.; Zeb, I. Deep neural survival networks for cardiovascular risk prediction: The Multi-Ethnic Study of Atherosclerosis (MESA). *Comput. Biol. Med.* **2021**, *139*, 104983. [CrossRef]
9. Katzman, J.L.; Shaham, U.; Cloninger, A.; Bates, J.; Jiang, T.; Kluger, Y. DeepSurv: Personalized treatment recommender system using a Cox proportional hazards deep neural network. *BMC Med. Res. Methodol.* **2018**, *18*, 24. [CrossRef]
10. She, Y.; Jin, Z.; Wu, J.; Deng, J.; Zhang, L.; Su, H.; Jiang, G.; Liu, H.; Xie, D.; Cao, N.; et al. Development and validation of a deep learning model for non–small cell lung cancer survival. *JAMA Netw. Open.* **2020**, *3*, e205842. [CrossRef]
11. Liang, W.; Yao, J.; Chen, A.; Lv, Q.; Zanin, M.; Liu, J.; Wong, S.; Li, Y.; Lu, J.; Liang, H.; et al. Early triage of critically ill COVID-19 patients using deep learning. *Nat. Commun.* **2020**, *11*, 3543. [CrossRef]
12. Fisher, L.D.; Lin, D.Y. Time-dependent covariates in the Cox proportional-hazards regression model. *Annu. Rev. Public Health* **1999**, *20*, 145–157. [CrossRef]
13. Moons, K.G.M.; Altman, D.G.; Reitsma, J.B.; Ioannidis, J.P.A.; Macaskill, P.; Steyerberg, E.W.; Vickers, A.J.; Ransohoff, D.F.; Collins, G.S. Transparent reporting of a multivariable prediction model for individual prognosis or diagnosis (TRIPOD): Explanation and elaboration. *Ann. Intern. Med.* **2015**, *162*, W1–W73. [CrossRef]
14. Zhou, X.; Nakamura, K.; Sahara, N.; Asami, M.; Toyoda, Y.; Enomoto, Y.; Hara, H.; Noro, M.; Sugi, K.; Moroi, M.; et al. Exploring and identifying prognostic phenotypes of patients with heart failure guided by explainable machine learning. *Life* **2022**, *12*, 776. [CrossRef]
15. McDonagh, T.A.; Metra, M.; Adamo, M.; Gardner, R.S.; Baumbach, A.; Böhm, M.; Burri, H.; Čelutkienė, J.; Chioncel, O.; Cleland, J.G.F.; et al. 2021 ESC Guidelines for the diagnosis and treatment of acute and chronic heart failure: Developed by the Task Force for the diagnosis and treatment of acute and chronic heart failure of the European Society of Cardiology (ESC) With the special contribution of the Heart Failure Association (HFA) of the ESC. *Eur. Heart J.* **2021**, *42*, 3599–3726.
16. Kodama, K.; Sakamoto, T.; Kubota, T.; Takimura, H.; Hongo, H.; Chikashima, H.; Shibasaki, Y.; Yada, T.; Node, K.; Nakayama, T.; et al. Construction of a heart failure database collating administrative claims data and electronic medical record data to evaluate risk factors for in-hospital death and prolonged hospitalization. *Circ. Rep.* **2019**, *1*, 582–592. [CrossRef]
17. Park, S.Y.; Park, J.E.; Kim, H.; Park, S.H. Review of statistical methods for evaluating the performance of survival or other time-to-event prediction models (from conventional to deep learning approaches). *Korean J. Radiol.* **2021**, *22*, 1697–1707. [CrossRef]
18. Zhou, X.; Nakamura, K.; Sahara, N.; Takagi, T.; Toyoda, Y.; Enomoto, Y.; Hara, H.; Noro, M.; Sugi, K.; Moroi, M.; et al. Deep learning-based recurrence prediction of atrial fibrillation after catheter ablation. *Circ. J.* **2022**, *86*, 299–308. [CrossRef]
19. Ng, K.; Steinhubl, S.R.; deFilippi, C.; Dey, S.; Stewart, W.F. Early detection of heart failure using electronic health records: Practical implications for time before diagnosis, data diversity, data quantity, and data density. *Circ. Cardiovasc. Qual. Outcomes* **2016**, *9*, 649–658. [CrossRef]
20. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. B* **1996**, *58*, 267–288. [CrossRef]
21. Mesquita, J.; Ferreira, A.M.; Cavaco, D.; Moscoso Costa, F.; Carmo, P.; Marques, H.; Morgado, F.; Mendes, M.; Adragão, P. Development and validation of a risk score for predicting atrial fibrillation recurrence after a first catheter ablation procedure–ATLAS score. *EP Eur.* **2018**, *20*, f428–f435. [CrossRef]
22. Shalabi, L.A.; Shaaban, Z.; Kasasbeh, B. Data mining: A preprocessing engine. *J. Comput. Sci.* **2006**, *2*, 735–739. [CrossRef]
23. Harrell, F.E.; Califf, R.M.; Pryor, D.B.; Lee, K.L.; Rosati, R.A. Evaluating the yield of medical tests. *JAMA* **1982**, *247*, 2543–2546. [CrossRef]
24. Lambert, J.; Chevret, S. Summary measure of discrimination in survival models based on cumulative/dynamic time-dependent ROC curves. *Stat. Methods Med. Res.* **2016**, *25*, 2088–2102. [CrossRef]
25. Park, S.H.; Han, K. Methodologic guide for evaluating clinical performance and effect of artificial intelligence technology for medical diagnosis and prediction. *Radiology* **2018**, *286*, 800–809. [CrossRef]
26. Brier, G.W. Verification of forecasts expressed in terms of probability. *Mon. Weather Rev.* **1950**, *78*, 1–3. [CrossRef]
27. Wang, P.; Li, Y.; Reddy, C.K. Machine learning for survival analysis: A survey. *ACM Comput. Surv.* **2019**, *51*, 1–36. [CrossRef]
28. Bazoukis, G.; Stavrakis, S.; Zhou, J.; Bollepalli, S.C.; Tse, G.; Zhang, Q.; Singh, J.P.; Armoundas, A.A. Machine learning versus conventional clinical methods in guiding management of heart failure patients—A systematic review. *Heart Fail. Rev.* **2021**, *26*, 23–34. [CrossRef]

29. Liu, S.; Yang, Z.; Pan, K.; Zhu, X.; Chen, W. Estimation of left ventricular ejection fraction using cardiovascular hemodynamic parameters and pulse morphological characteristics with machine learning lgorithms. *Nutrients* **2022**, *14*, 4051. [CrossRef]
30. Levy, W.C.; Mozaffarian, D.; Linker, D.T.; Sutradhar, S.C.; Anker, S.D.; Cropp, A.B.; Anand, I.; Maggioni, A.; Burton, P.; Sullivan, M.D.; et al. The Seattle Heart Failure Model: Prediction of survival in heart failure. *Circulation* **2016**, *113*, 1424–1433. [CrossRef]
31. Mentz, R.J.; Mulder, H.; Mosterd, A.; Sweitzer, N.K.; Senni, M.; Butler, J.; Ezekowitz, J.A.; Lam, C.S.P.; Pieske, B.; Ponikowski, P.; et al. Clinical outcome predictions for the VerICiguaT Global Study in subjects with heart failure with reduced ejection fraction (VICTORIA) trial: VICTORIA outcomes model. *J. Card. Fail.* **2021**, *27*, 949–956. [CrossRef]
32. Simpson, J.; Jhund, P.S.; Lund, L.H.; Padmanabhan, S.; Claggett, B.L.; Shen, L. Prognostic models derived in PARADIGM-HF and validated in ATMOSPHERE and the Swedish Heart Failure Registry to predict mortality and morbidity in chronic heart failure. *JAMA Cardiol.* **2020**, *5*, 432–441. [CrossRef] [PubMed]
33. Gustafsson, F.; Torp-Pedersen, C.; Seibaek, M.; Burchardt, H.; Køber, L.; DIAMOND study group. Effect of age on short and long-term mortality in patients admitted to hospital with congestive heart failure. *Eur. Heart J.* **2004**, *25*, 1711–1717. [CrossRef] [PubMed]
34. Kajimoto, K.; Sato, N.; Investigators of the Acute Decompensated Heart Failure Syndromes (ATTEND) Registry. Sex differences in New York Heart Association functional classification and survival in acute heart failure patients with preserved or reduced ejection fraction. *Can. J. Cardiol.* **2020**, *36*, 30–36. [CrossRef] [PubMed]
35. Shamagian, L.G.; Román, A.V.; Pérez, M.P.; Otero, I.G.; Lamela, A.V.; González-Juanatey, J.R. Renal failure is an independent predictor of mortality in hospitalized heart failure patients and is associated with a worse cardiovascular risk profile. *Rev. Esp. Cardiol.* **2006**, *59*, 99–108. [CrossRef]
36. Rudolf, H.; Mügge, A.; Trampisch, H.J.; Scharnagl, H.; März, W.; Kara, K. NT-proBNP for risk prediction of cardiovascular events and all-cause mortality: The getABI-study. *Int. J. Cardiol. Heart Vasc.* **2020**, *29*, 100553. [CrossRef] [PubMed]
37. Dunlay, S.M.; Manemann, S.M.; Chamberlain, A.M.; Cheville, A.L.; Jiang, R.; Weston, S.A.; Roger, V.L. Activities of daily living and outcomes in heart failure. *Circ. Heart Fail.* **2015**, *8*, 261–267. [CrossRef]
38. Gohbara, M.; Nishimura, K.; Nakai, M.; Sumita, Y.; Endo, T.; Matsuzawa, Y.; Konishi, M.; Kosuge, M.; Ebina, T.; Tamura, K.; et al. Low activities of daily living associated with increased cardiovascular disease mortality in Japan: Analysis of health records from a Nationwide Claim-Based Database, JROAD-DPC. *Circ. Rep.* **2019**, *1*, 20–28. [CrossRef] [PubMed]
39. Lakhani, I.; Wong, M.V.; Huang, J.K.F.; Gong, M.; Waleed, K.B.; Xia, Y.; Lee, S.; Roever, L.; Liu, T.; Tse, G.; et al. Diagnostic and prognostic value of serum C-reactive protein in heart failure with preserved ejection fraction: A systematic review and meta-analysis. *Heart Fail. Rev.* **2021**, *26*, 1141–1150. [CrossRef] [PubMed]

diagnostics

MDPI

Article

EEG-Based Mental Tasks Recognition via a Deep Learning-Driven Anomaly Detector

Abdelkader Dairi [1,†], Nabil Zerrouki [2,†], Fouzi Harrou [3,*,†] and Ying Sun [3,†]

[1] Computer Science Department, University of Science and Technology of Oran-Mohamed Boudiaf (USTO-MB), El Mnaouar, BP 1505, Bir El Djir 31000, Algeria
[2] Design and Implementation of Intelligent Machines (DIIM) Team, Center for Development of Advanced Technologies, Baba Hassen 16081, Algeria
[3] Computer, Electrical and Mathematical Sciences and Engineering (CEMSE) Division, King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Saudi Arabia
* Correspondence: fouzi.harrou@kaust.edu.sa
† These authors contributed equally to this work.

Abstract: This paper introduces an unsupervised deep learning-driven scheme for mental tasks' recognition using EEG signals. To this end, the Multichannel Wiener filter was first applied to EEG signals as an artifact removal algorithm to achieve robust recognition. Then, a quadratic time-frequency distribution (QTFD) was applied to extract effective time-frequency signal representation of the EEG signals and catch the EEG signals' spectral variations over time to improve the recognition of mental tasks. The QTFD time-frequency features are employed as input for the proposed deep belief network (DBN)-driven Isolation Forest (iF) scheme to classify the EEG signals. Indeed, a single DBN-based iF detector is constructed based on each class's training data, with the class's samples as inliers and all other samples as anomalies (i.e., one-vs.-rest). The DBN is considered to learn pertinent information without assumptions on the data distribution, and the iF scheme is used for data discrimination. This approach is assessed using experimental data comprising five mental tasks from a publicly available database from the Graz University of Technology. Compared to the DBN-based Elliptical Envelope, Local Outlier Factor, and state-of-the-art EEG-based classification methods, the proposed DBN-based iF detector offers superior discrimination performance of mental tasks.

Keywords: deep learning; motor imagery; Isolation Forest; anomaly detection; EEG signals classification

Citation: Dairi, A.; Zerrouki, N.; Harrou, F.; Sun, Y. EEG-Based Mental Tasks Recognition via a Deep Learning-Driven Anomaly Detector. *Diagnostics* **2022**, *12*, 2984. https://doi.org/10.3390/diagnostics12122984

Academic Editors: Hiam Alquran and Wan Azani Mustafa

Received: 22 October 2022
Accepted: 24 November 2022
Published: 29 November 2022

1. Introduction

The brain–computer interface (BCI) is a recent manner of communication where it allows translating oscillatory Electro Encephalogram (EEG) patterns into action [1]. BCI technology has proven very successful in the scientific community, since it allows the command of several devices, such as a computer, from the assisted control of the human brain [2]. It is based on the electrical brain activities of disabled patients who have lost their mobility autonomy [3]. In other words, several motivations can be highlighted for designing an MI-based system. Suppose one considers only the case of motor-impaired people. In that case, MI-based task recognition can solve several daily problems, such as a text-entry system, making a phone call, and wheel-chair control, especially if the recognition system has certain reliability and robustness in real time. In a more advanced stage, MI-based systems for BCI applications can be much more important if communication between two or more patients can be established via the MI-based system. It is worth noting that many researchers and engineers have developed several methods and systems for EEG signals classification during the last few years [4,5]. However, there are still several unsolved problems in motor imagery (MI)-based tasks recognition [6]. Essentially, transforming thoughts into actions via BCI remains challenging because recorded EEG signals are highly

affected by background noise and different artifacts, including eye blinking, cardiac activity, and the state of stress of the patient.

Various approaches have been proposed for MI tasks classification in the last few years. For instance, the authors in [3] applied tunable-Q wavelet transform (TQWT) on EEG signals for extracting time-frequency features, and the Least-squares Support Vector Machine (LSSVM) algorithm was applied for separating between right-hand and right-foot MI tasks. In another work, the same research team subsequently [6] proposed the use of Analytic Intrinsic Mode Functions (AIMFs) based on Empirical Mode Decomposition (EMD) and Hilbert transform for the EEG signal feature extraction module. To this end, several attributes have been generated, such as peak value and spectral moment of power spectral density, and the raw moment of the first derivative of instantaneous frequency. These features are then considered inputs to the LSSVM classifier with radial basis function as kernel function, where an overall accuracy of 97.56% was obtained. The study in [7] proposed optimal allocation features to discriminate the operative information from EEG data with a minimum possible rate of variability. The classification stage was performed using Naive Bayes and LS-SVM algorithms, obtaining an accurate detection percentage of 96.36% and 96.62%, respectively. The authors in [8] presented an EEG de-noising phase using a nonlinear filter based on the Multi-Scale Principal Component Analysis (MSPCA) technique. Diverse attributes were generated by the application of Empirical Mode Decomposition (EMD), discrete wavelet transform (DWT), and wavelet packet decomposition (WPD). In the classification stage, a k-nearest neighbor algorithm was applied to separate between two classes, namely right hand and foot, where a correct recognition rate of 92.8% was obtained according to their experimental results. Authors in [9] designed sliding window techniques to improve the binary classification of motor imagery, where features are extracted via Common spatial pattern (CSP), and the classification is conducted using linear discriminant analysis (LDA). They have used the BCI Competition IV-2a data dataset, which is publicly available, and demonstrated satisfactory classification performance of this approach by reaching an accuracy of around 80%. In [10], a combined approach merging an improved group least absolute shrinkage and selection operator (LASSO) is proposed for EEG signals spatial smoothing, features selection, and classification. It has been demonstrated that this approach can enhance the performance of BCI systems. In [11], a combination of cross-correlation and discrete wavelet transform (DWT) has been employed in feature generation and selection procedures to classify MI-based EEG signals. After that, the classification task was conducted by the application of five different methods, namely: multilayer perceptron neural network (MLP), probabilistic neural network (PNN), logistic regression (LR), kernelized logistic regression (KLR), and LS-SVM. In [12], an approach based on machine learning and feature selection techniques is considered for motor imagery EEG signal classification on the internet of medical things environment. To this end, actual feature sets are obtained from BCI Competition-II Dataset-III motor-imagery EEG signal using the Adaptive Auto-regressive approach. Then, an innovative fuzzified extension of the discernibility matrix is employed for feature selection. Results demonstrated that when used with SVM and Ensemble variants of classifiers, this feature selection procedure outperformed other commonly used approaches.

In recent years, there has been a growing interest in designing efficient techniques for EEG-based classification of motor imagery tasks exploiting machine learning and deep learning models. However, most of these techniques are developed in a supervised learning procedure where labeling information is needed. Essentially, this work focuses on developing a flexible and unsupervised data-driven approach to effectively identify mental tasks based on EEG signals. Here, we addressed the multiclass mental tasks classification as an anomaly detection problem employing an unsupervised deep learning model. Overall, the contributions of this study are recapitulated as follows.

- This work presents a deep learning-based anomaly detection strategy to enhance mental tasks recognition by EEG data. This strategy comprises several stages, i.e., artifacts removal, extraction of time-frequency features of EEG signals, anomaly detection, and

classes discrimination. Specifically, the EEG signals are first filtered using the Multichannel Wiener filter (MWF) to remove artifacts and achieve robust recognition. We adopted a quadratic time-frequency distribution (QTFD) for extracting high-resolution time-frequency signal representation of the EEG signals. The employment of a QTFD technique is expected to improve the recognition of mental tasks by capturing the EEG signals' spectral variations over time. The extracted time-frequency features are inputs to the proposed unsupervised deep learning-based approach for classifying the EEG signals. Indeed, this study treated multiclass classification as a multiple-binary discrimination problem. Specifically, this approach combines the desirable characteristics of both a deep belief network (DBN) and an isolation forest (IF) technique for separating mental tasks based on the time-frequency features of EEG signals. Crucially, this technique profits from the greedy learning characteristics of the DBN for extracting pertinent information from the QTFD features and the capacity of the IF detector to sense outliers. The IF algorithm's key characteristic is its ability to reveal anomalies without using distance or density metrics. This enables eliminating computational costs related to distance computation in all distance-driven and density-driven models. In addition, the iF detector can handle large-sized datasets with many irrelevant features [13]. Indeed, a single DBN-based IF detector is constructed based on training data in the targeted class, i.e., the samples in such class are considered inliers, and all remaining samples are considered anomalies (i.e., one-vs.-rest). We evaluated the efficacy of this technique through experimental data comprising five mental tasks: mental word association, mental subtraction, spatial navigation, right-hand motor imagery, and feet motor imagery, from a publicly available database from the Institute for Knowledge Discovery, Graz University of Technology, Austria. Thus, to separate the five mental tasks by EEG signals, by using one-vs.-rest method, we constructed five DBN-IF detectors.

- Furthermore, the discrimination capabilities of the DBN-IF scheme have been compared with those of DBN-based Local Outlier Factor (LOF) and Elliptical Envelope (EE) anomaly detection methods. As we know, DBN-based LOF and EE methods have not previously been used for EEG-based mental tasks identification. The essence of LOF is based on the idea of local anomalies [14], while the EE senses anomalies by fitting an ellipse around the data utilizing the Minimum Covariance Determinant [15]. We assessed the performance of the investigated technique using four commonly used statistical scores. Results revealed that the proposed DBN-IF approach dominates the other investigated approaches.
- In addition, the results of the DBN-IF approach are compared with the state-of-the-art techniques; the results demonstrated the proposed approach's outperformance in improving the separation of metal tasks based on EEG signals.

The remainder of this paper is organized as follows. Section 3 briefly describes the preliminary materials, including the MWF artifact removal technique, the QTFD features extractor, the DBN model, and the iF anomaly detector. Section 3 presents the proposed approach to distinguish between the five mental tasks. In Section 4, we present the used data and the obtained results. Finally, we offer conclusions in Section 5.

2. Related Works

Mental tasks recognition based on EEG signals is a challenging problem in EEG signal processing and analysis. Recently, deep learning models are employed to enhance medical applications in academia and industry due to their ability of extracting pertinent features of high dimensional data [16]. They demonstrated promising performance in various applications, including COVID-19 infection detection [17], Parkinson's disease detection [18]. Various studies have investigated deep learning techniques for EEG classification problems in recent years. For instance, in [19], Bashivan et al. proposed a new representation of EEG classification problematic, where raw EEG signals are transformed into a sequence of topology multi-spectral images or frames. Inspired by the representation

of images and videos classification using deep learning techniques, the authors applied a deep recurrent convolutional network (CNN) for classifying the transformed images. However, it remains challenging to preserve EEG signals' spatial, spectral, and temporal structure after transforming them into 2D images. The study in [20] applied Extreme Learning Machine (ELM) to discriminate five mental tasks based on EEG signals. Results demonstrated that ELM obtained similar classification performance in terms of accuracy as SVMs and Backpropagation Neural Network (BPNN) classifiers. However, it has less training time compared to SVMs and BPNN. Furthermore, it has been demonstrated that smoothing the classifiers' outputs enhances their discrimination accuracies. In [21], two approaches based on deep convolutional neural networks and deep residual learning are applied for the EEG classification of driver mental states. Data from a driving simulation platform is used to verify the effectiveness of these classifiers. Results based on intra- and inter-subject demonstrated that the two models achieved good classification performance and outperformed the LSTM- and SVM-driven classifiers. However, this study focused only on a binary classification to predict driver fatigue. In kuremoto2019mental, Kuremoto et al. proposed hybrid machine learning methods for EEG-based mental task recognition by combining SVM and neural networks (e.g., MLP, CNN, and stacked auto-encoder (SAE)), as well as the mixed SAE+MLP, CNN+MLP models. Results revealed the superior classification accuracy of the hybrid models than the traditional methods (i.e., MLP, SVM, and CNN). However, the inputs considered by these models consist of the raw EEG signals; other input alternatives can be considered for improvements, such as the preprocessed data by wavelet transform or Fourier transform. In [22], Opalka et al. adopted a Multi-Channel Convolutional Neural Networks architecture for EEG mental tasks classification. Results based on data from V from BCI Competition III demonstrated the superior classification performance of this approach with an accuracy of around 70%, surpassing alternative methods (i.e., AlexNet, VGG-16, and Cecotti's multi-channel NN).

In [23], time-frequency features and location information are first extracted from MI EEG signals, and the short-time Fourier transform (STFT) method was then applied to proceed into a 2D image representation. In the classification phase, CNN with only 1D convolutional and one max-pooling layer was combined with stacked autoencoders (SAE), obtaining a correct recognition rate of 90.0%. In [24], a classification framework using a long short-term memory (LSTM) with one dimension-aggregate approximation as a feature extractor was proposed to classify EEG motor imagery tasks. This approach employs a softmax layer for predicting the probability of every class. Classification results based on public BCI competition data demonstrated that the LSTM-based approach outperformed the state-of-the-arts approaches using no other deep networks by obtaining an averaged accuracy of 75.28%. In [25], a hybrid approach called frequential deep belief network (FDBN) is designed to deal with Motor imagery classification problems by combining a fast Fourier transform (FFT), and wavelet package decomposition (WPD) are combined with a deep belief neural network (DBN). At first, FFT and WPD are applied to obtain frequency domain representations of EEG signals, and their output features are used to train the DBN model. Here, a softmax layer is adopted to perform the classification task. In [26], a hierarchical flow convolutional neural network (HF-CNN) model is introduced to classify forearm movements using electroencephalogram (EEG) signals. This approach has been assessed using experimental and BNCI Horizon 2020 datasets and demonstrated a moderate classification performance. This study helps develop a brain-controlled robotic arm system to perform high-level tasks. Authors in [27] proposed a Graph-based CNN network combined with an attention model for motor imagery classification. Specifically, the positioning information of EEG nodes is first represented by a graph structure, and then the CNN with attention is applied to learn EEG features. They showed that the EEG graph with more nodes significantly enhances the overall performance. Authors in [28] present a motor imagery tasks' EEG signals classification using CNN in the brain–BCI system. Essentially, the CNN model is employed to classify the right hand and right foot MI-task using EEG signals. To this end, the CNN moded is trained with transformed EEG

signals into images via time-frequency approaches, namely short-time Fourier transform (STFT) and continuous wavelet transform (CWT). Results revealed that the classification performance achieved by using CWT was significantly better than that obtained via the STFT approach. In another study [29], a classification of hand movements framework based on EEG signals is introduced using a deep attention-based LSTM network. Importantly, the attention-based LSTM is trained using time and frequency domain features extracted from the EEG signals. However, this approach is based on the use of hand-crafted features.

3. Materials and Methods

The general framework of the proposed EEG-based mental tasks recognition is schematically illustrated in Figure 1. This framework consists of five main steps: data acquisition, artifacts removal through MWF, time-frequency representation of the EEG signals using QTFD, feature extraction via the DBN model, and mental tasks recognition based on the IF anomaly detection scheme.

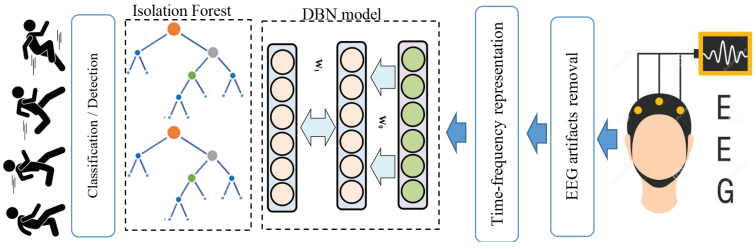


Figure 1. Schematic representation of the general deep learning framework for EEG-based metal tasks discrimination.

3.1. EEG Artifacts Removal Using Multi-Channel Wienner Filter

The collected EEG signals are usually tinted with glitches or spikes due to sudden changes in skin-electrode contact impedance. Specifically, this situation is often produced by the movement of the subject head, resulting in the shift of electrodes around. This movement artifact presents a very large magnitude peak that impacts a single channel or a few adjacent channels. The morphology of this kind of artifacts (with focused spatial and sparse temporal structure appearing only once in a few channels in the recording) can differ significantly compared to eye blinks artifacts, which are redundant for a subject.

This study gives EEG signals input to an MWF algorithm for EEG artifacts' removal. The MWF algorithm is known for its efficiency using both hybrid and actual EEG data, since it can eliminate a wide range of artifacts with more satisfactory performance than current existing techniques [30]. Notably, the main idea behind MWF is that a low-rank approximation replaces the artifact covariance matrix via the generalized eigenvalue decomposition [30]. The employment of MWF as an artifact removal technique was strongly motivated by the fact that the MWF is not limited to a specific kind of artifact, and it is robust and generic for various types of EEG artifacts (other artifacts than eye blink or muscle artifacts such as movement artifact) [30]. Figure 2 provides an illustration of some examples of EEG signals before and after artifacts' removal using the MWF algorithm.

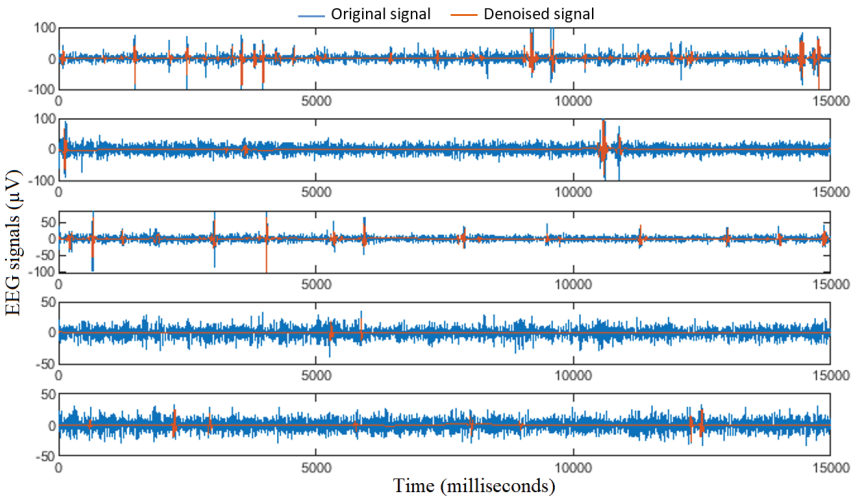


Figure 2. Illustration of EEG signals before and after artifacts' removal.

3.2. Time-Frequency Representation of EEG Data via a QTFD

After obtaining relevant EEG signals by removing artifacts, they are used as input to time-frequency features through quadratic time-frequency distributions (QTFDs). This study uses QTFD to extract time-frequency components due to its capacity to catch the EEG signals' spectral variations over time. In particular, the EEG signals could be represented using a QTFD by discriminant features that could improve the recognition ratio of distinct emotional classes. Moreover, the QTFD has been efficiently implemented to characterize EEG signals in several fields, including decoding motor imagery tasks [31]. The main idea behind using QTFD components is to consider the nonlinearity aspect in mapping EEG signals; moreover, QTFDs are invariant to the time-frequency shift, which can avoid some limitations encountered using other time-frequency analysis techniques, such as Scale-invariant feature transform or Wavelet Transform [31]. In order to calculate the QTFD components of EEG signals, a sliding window is used to split the EEG signal of each channel into a set of EEG segments.

For each EEG segment, the QTFD-based time-frequency representations are based on the Hilbert transform and Wigner–Ville distribution [32]. Given the time evolution EEG segment (real-valued signal) $s(t)$, the QTFD components are computed, as follows:

$$a(t) = s(t) + j\mathcal{HT}(s(t)), \tag{1}$$

where $a(t)$ and $\mathcal{HT}$ represent the Hilbert transform and the analytic signal, respectively. The Wigner–Ville distribution (WVD) of the signal $a(t)$ is then computed as follows:

$$WVD_a(t,f) = \int_{-\infty}^{\infty} a(t+\frac{\tau}{2})a^*(t-\frac{\tau}{2})e^{-j2\pi\tau f}\partial\tau, \tag{2}$$

where τ and $a^*(t)$ represent the parameter of the centered formulation and the complex conjugate of $a(t)$, respectively. Finally, to obtain QTFD, we simply convolve the computed $\mathrm{WVD}_a(t,f)$ with a time-frequency kernel $K(t,f)$.

$$\varrho_a(t,f) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} WVD_a(\phi,\tau)K(\phi,\tau)e^{-j2\pi f\tau - j2\pi t\phi}\partial\tau\partial\phi, \tag{3}$$

where $\varrho_a(t,f)$ represents the QTFD of $a(t)$. In the recognition cases, exponential time-frequency kernel is generally used to moderate the aspect of the cross-terms and keep a

satisfactory resolution in both the time- and frequency-domain. In the present case, the exponential kernel is expressed as:

$$K(t,f) = \exp\left(-\frac{t^2 f^2}{\alpha^2}\right). \quad (4)$$

Here, α represents the parameter controlling the suppression of the cross-terms. In other words, the exponential kernel tends to eliminate the interference, which is away from the origin. The QTFD based on the exponential kernel is called the Choi-Williams distribution (CWD). More details and explanations about the QTFD or CWD extraction process can be found in [31]. To be more explicit and provide more details concerning QTFD feature extraction, Figure 3a,b illustrates two examples of the Choi-Williams decomposition. For instance, in Figure 3a, the examined EEG signal is given in the bottom panel, the response in the frequency domain is shown in the right panel, and its corresponding spectrum is displayed in the left panel.

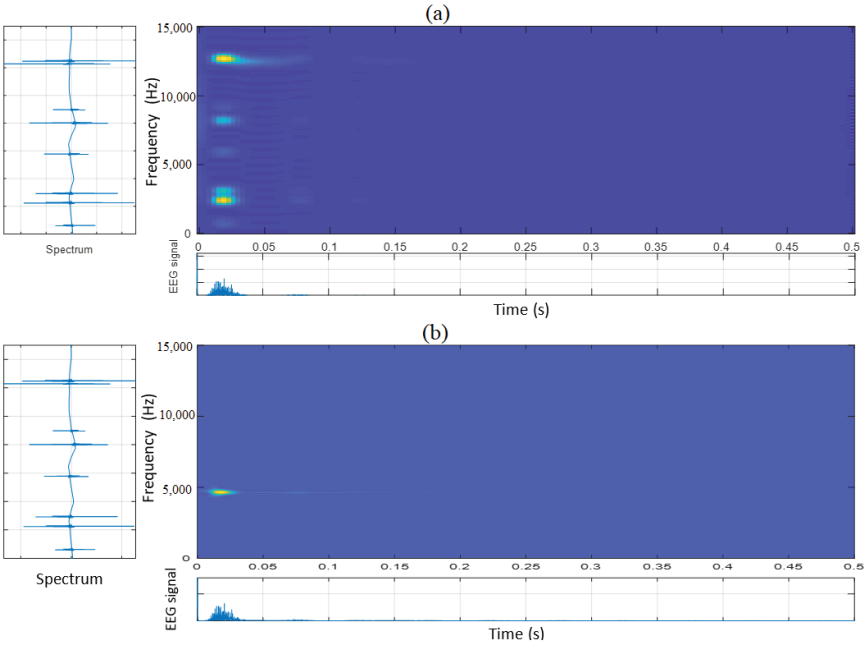


Figure 3. A practical example of Choi-Williams decomposition (QTFD) of EEG signal.

The extracted QTFD-based time-frequency features will be used as input by the proposed deep learning-based anomaly detection approach to discriminate between the various mental tasks.

3.3. Deep Belief Network (DBN)

DBNs are probabilistic generative models designed by stacked restricted Boltzmann machines (RBMs). RBMs are a powerful tool for extracting and representing data adopted in machine learning [33] (see Figure 4a). As schematized in Figure 4a, RBM is a variant of the conventional Boltzmann Machine (BM), which removes all connections in the same layer, and only the connections between visible and hidden layers are preserved [33].

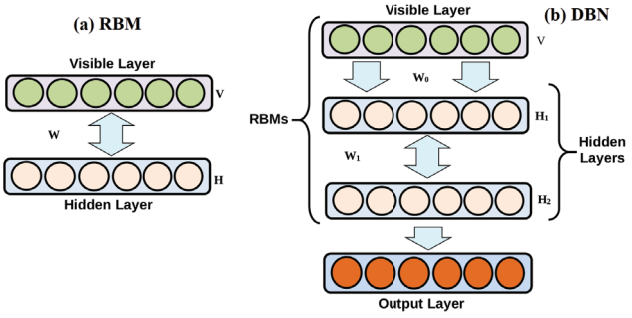


Figure 4. Schematic illustration of (**a**) RBM and (**b**) DBN models.

RBMs are energy-driven models and were employed as generative models for different kinds of data [34], including text, speech, and images. The energy of join structure is expressed [35]:

$$Energy(v,h) = -\sum_{i=1}^{m}\sum_{j=1}^{n} W_{ij}v_i h_j, -\sum_{i=1}^{m} b_i v_i - \sum_{j=1}^{n} c_j h_j, \tag{5}$$

where W_{ij} represents the element of W which connects the the ith visible variable v_i to the jth hidden variable h_j, b and c denote the parameters of the model.

Then, the underlying Boltzmann distribution can be computed as [36]:

$$P(v,h) = \frac{\exp(-Energy(v,h))}{\sum_v \sum_h \exp(-Energy(v,h))} = \frac{\prod_{ij} e^{W_{ij}v_i h_j} \prod_i e^{b_i v_i} \prod_j e^{a_j h_j}}{\sum_v \sum_h \exp(-Energy(v,h))}, \tag{6}$$

Since only v is observed, the hidden variables h are marginalized.

$$P(v) = \sum_h \frac{e^{-Energy(v,h)}}{\sum_v \sum_h \exp(-Energy(v,h))}, \tag{7}$$

where $P(v)$ refers to the probability allocated by the mode to a visible vector v. As there is no connection between the nodes at the same layer (Since the intra-connections are absent at both layers), the corresponding conditional probabilities are:

$$P(v|h) = \prod_i p(v_i|h), \text{and } P(h|v) = \prod_j p(h_j|v). \tag{8}$$

For binary data, equations in (8) can be reformulated as:

$$P(v_i = 1|h) = \sigma(\sum_j W_{ij}h_j + c_i), \tag{9}$$

$$P(h_j = 1|v) = \sigma(\sum_i W_{ij}v_i + b_j), \tag{10}$$

where $\sigma(.)$ denotes the logistic function and $\sigma(x) = (1 + \exp(-x))^{-1}$.

DBNs are built up by stacking RBMs (Figure 4b) and trained in an unsupervised manner for extracting pertinent features from the input data. They proved to be effective in uncovering layer-by-layer complex nonlinearity. In [35], a fast learning strategy for DBN was introduced, where the joint distribution between observed vector x and ℓ hidden layers h^k is obtained as [34],

$$P(x, h^1, \ldots, h^l) = (\prod_{k=0}^{\ell-2} P(h^k|h^{k+1}))P(h^{\ell-1}, h^\ell), \tag{11}$$

where $x = h^0$ and $P(h^k|h^{k+1})$ is a visible given hidden conditional distribution in an RBM associated with level k of the DBN, and $P(h^{\ell-1}, h^{\ell})$ is the joint distribution in the top-level RBM.

Basically, including more layers in the DBN enhances modeling power. The accuracy of the energy expression could be improved by incorporating more additional layers into the DBN model [34]. However, little is earned by employing more than three hidden layers in practice. For instance, in this study, we stacked two RBMs (Figure 4b) to construct our DBN model without any labeling information.

3.4. Isolation Forest Approach

The Isolation Forest is a promising anomaly detection algorithm primarily introduced by Lui in 2008 [13] and enhanced thereafter in 2011 [37]. It is constructed using unlabeled data, making it suitable for practice applications. The principal idea of the IF algorithm consists in identifying anomalies by the isolation of potential outliers from the data [37]. It is inspired based on the Random Forest, which consists of an ensemble of decision trees built in the training step [38]. Visually, Figure 5 depicts the basic construction of the IF, which consists in constructing an ensemble of trees for a given data. Importantly, the iF algorithm recursively divides the data by constructing an ensemble of trees until all samples are separated. Note that anomalies are recognized by a short average path length on the trees [13].

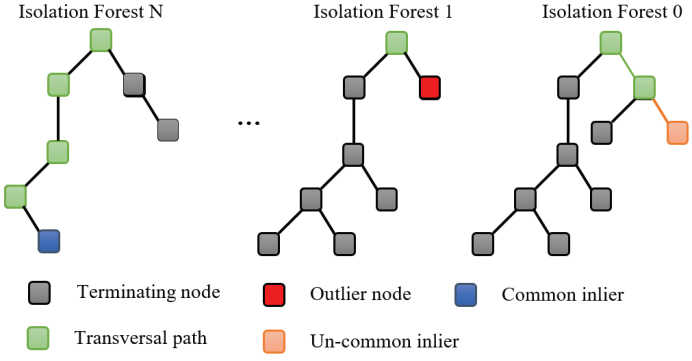


Figure 5. Illustration of anomaly detection using the Isolation Forest technique.

Implementing the iF-based anomaly detection approach demands only two parameters specified: the number of trees and the size of sub-samples used for the splitting operations to build the forest. In [13], it has been demonstrated that the detection performance of the iF approach can converge quickly based on a small number of trees, and it only needs a small sub-sampling size to reach high detection accuracy. In the iF approach, anomalies in a dataset can be detected by analyzing the path lengths for the anomaly data points, with the splitting process being short, which mean that anomalies require few splits in isolation Trees to be isolated [39]. Furthermore, the anomaly score is computed from the mean path length across all the isolation trees in the forest.

Two parameters are to be fixed in implementing the iF algorithm: the number of trees and the size of sub-samples employed for the splitting procedure to construct the forest [13]. Anomalies can be detected using the iF approach by investigating the path lengths for the anomaly data points. Anomalies are distinguished by a short splitting process in isolation Trees to be isolated [39]. The iF algorithm calculates the anomaly score to decide the presence of anomalies based on the mean path length across all the isolation trees in the forest.

Let us consider $l(d)$ the path length of a given data point d, and $\mathcal{D}$ a dataset constituted of N data samples. In the IF approach, $log(N)$ is the minimum depth of a used decision tree, and $N-1$ is the maximum depth. The anomaly score, $\mathcal{A}$, is computed as follows [13]:

$$\mathcal{A}(d,N) = 2^{-\frac{E\left[l(d)\right]}{\alpha(N)}}, \tag{12}$$

where $E\left[l(d)\right]$ is the expected path length of a given data point d from a collection of isolation trees, and $\alpha(N)$ is the average path length, given as [13]:

$$\alpha(N) = 2\lambda(N-1) - \frac{2(N-1)}{N}, \tag{13}$$

where $\lambda(i)$ denotes the harmonic number, which could be computed as:

$$\lambda(y) = ln(y) + \epsilon, \tag{14}$$

with ϵ denotes the Euler Constant, i.e., $\epsilon = 0.5772156649$.

In summary, we obtain the anomaly score of d, $\mathcal{A}(d,N)$ using iTree from the training data of N samples, and the range of $\mathcal{A}(d,N)$ is within $[0,\ 1]$. Note that the anomaly score is oppositely proportional to the path length. Anomaly detection is accomplished, as below.

$$\begin{cases} \text{an anomaly} & \text{if } \mathcal{A}(d,N) \text{ is close to 1} \\ \text{Normal instance} & \text{if } \mathcal{A}(d,N) \text{ is close to 0} \\ \text{Uncertain decision} & \text{if } \mathcal{A}(d,N) \text{ is close to 0.5} \end{cases} \tag{15}$$

4. Deep-Learning-Driven Mental Tasks Detector

This study addresses the problem of EEG-based mental tasks classification as multiple anomaly detection channels. Specifically, a hybrid deep generative model is used to model one given class in which its data are regarded as a normal observation, and the data from other classes are abnormal. Here, we first extract time-frequency features of the cleaned EEG data via the QTFD approach. Then, we applied the DBN-based iF detector to identify the mental tasks from the QTFD features of EEG signals, as illustrated in Figure 6. The primary goal for the DBN-iF approach is to identify the mental task from time-frequency features of the EEG signals.

As illustrated in Figure 6, for each class C_i, we implemented a DBN-based iF approach in an unsupervised manner to identify whether the mental task is from this class or not. Essentially, the ith DBN-iF detector is constructed using only the data of the ith class, C_i. Indeed, for a given class C_i, we trained the DBN in an unsupervised manner using only the data of the C_i class is used for the training. This step generates a compact features space for C_i, which will be used to train a dedicated isolation forest in order to isolate abnormal observations, which are the data points that belong to the other classes, and to keep the observed data points that are normal observation belonging to the current class C_i. Of course, the 1-vs-all procedure constructs k DBN-iF detectors to separate mental tasks by EEG signals. DBN is composed of a stacked RBM , where each RBM is trained separately in an unsupervised way; this approach is also called greedy layer-wise. This step permits to construct in the hierarchical process a reduced features space containing pertinent information that represents a given class C_i.

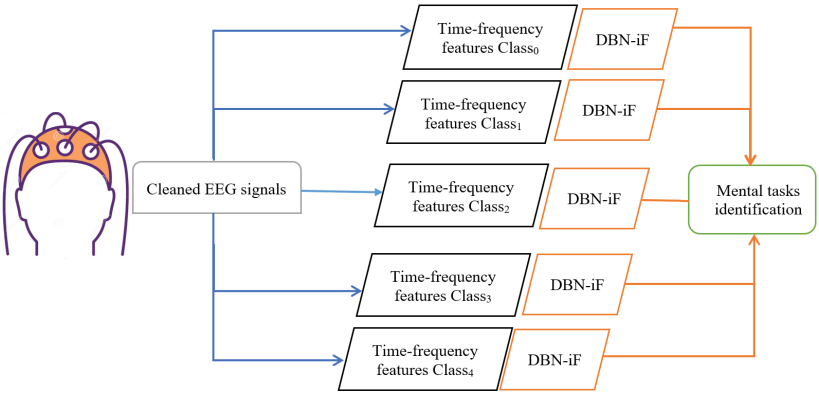


Figure 6. Schematic overview of the proposed DBN-iF approach for mental tasks' identification using EEG signals.

This work's central idea consists of constructing a DBN-based deep learning model for each EEG signals-based mental task. DBN aims to learn the probability distribution of the underlining training data. DBN is composed of a stacked RBM, where each RBM is trained separately in an unsupervised way; this procedure is also called greedy layer-wise learning. This step permits construction in the hierarchical process a reduced features space containing pertinent information that represents the data of a given class C_i. The training is performed by estimating the log-likelihood gradient based on the Gibbs Sampling method, Markov Chain Monte Carlo (MCMC) method. The output of unsupervised greedy layer-wise learning is indeed a new compact space constituted of relevant features that effectively represent the mapping of training data points into a latent space $\mathcal{L}$, which is used to feed the iF detector.

The iF approach is applied to the extracted features from the DBN model for EEG-based mental tasks recognition. In other words, the iF scheme is employed to discriminate a given observation as an anomaly if the EGG signal does not belong to target class C_i by assigning a high anomaly score (close to one). Otherwise, it assigns a low anomaly score (close to zero) if the EGG signal shares the same features as the target class C_i. In summary, the DBN-iF is designed for identifying each class separately without any data labeling.

Five statistical scores are computed in this study to compare the studied techniques: Recall, Precision, F1-Score, Accuracy, and Area under curve (AUC) [40]. For binary detection, the number of true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN) are used to calculate the statistical scores.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}. \tag{16}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{17}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{18}$$

$$\text{F1} - \text{Score} = 2\frac{\text{Precision.Recall}}{\text{Precision} + \text{Recall}} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}. \tag{19}$$

5. Results and Discussion

5.1. Data Description

This part is dedicated to evaluating the efficacy of the presented technique in recognizing different classes; experiments are conducted via actual data from the Institute for Knowledge Discovery. The EEG signals are recorded by nine different patients with disabilities (spinal cord injury and stroke) on distinct sessions. Nine patients completed a

specific experimental paradigm, including five mental tasks (MT): mental word association (condition WORD), spatial navigation (NAV), mental subtraction (SUB), feet motor imagery (FEET), and right-hand motor imagery (HAND. See [41] for more details.

The experiment protocol was carried out over several days, where for each day, a single subject session (involved eight runs resulting in 40 trials) is recorded. Each experimental run contained 25 cues with five different mental tasks. Cues (indicating different classes) were put in random ranking to allow a fair evaluation. EEG signals were then acquired from 30 electrodes positioned on the scalp according to the international 10–20 protocol. This protocol was developed to maintain standardized testing methods ensuring that a patient's study outcomes could be reproduced and effectively analyzed and compared to previously obtained results in the literature [42]. The locations of electrode contained channels AFz, F7, F3, Fz, F4, F8, FC3, FCz, FC4, T3, C3, Cz, C4, T4, CP3, CPz, CP4, P7, P5, P3, P1, Pz, P2, P4, P6, P8, PO3, PO4, O1, and O2. Channels corresponding to reference and ground have been placed at the left and right mastoid. For better representativeness, an overview of the experimental protocol is given in Figure 7. The acquisition was carried out using the g.tec GAMMAsys system, g.USBamp biosignal amplifiers and g.LADYbird active electrodes (Guger Technologies, Graz, Austria).

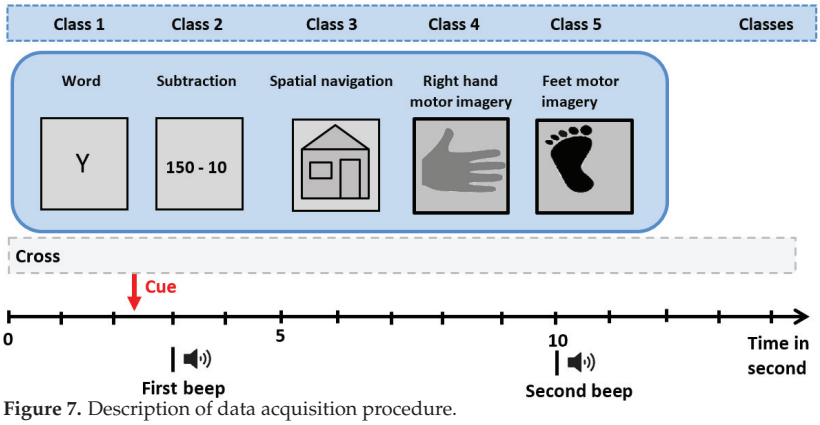


Figure 7. Description of data acquisition procedure.

After the acquisition phase, several processing operations were applied to the EEG signals: bandpass filter 0.5–100 Hz (notch filter at 50 Hz) and sampling stage at a rate of 256 Hz. The period of a single imagery experiment consists of ten seconds. In the beginning, a cross is shown on the screen, and participants are requested to relax and fixate the cross in order to avoid eye motions. A beep is given after three seconds to bring the participant's attention. The cue revealing the asked imagery task, one out of five graphical symbols, was shown between the time interval t = 3 s and t = 4.25 s. Then, a second beep was conducted at time t = 10 s, and the fixation-cross was removed, indicating the trial's end. Before the subsequent trial, a variable break (intertrial interval, ITI) stayed around 2.5 s and 3.5 s. Participants are requested to escape moving during the imagery period and avoid moving and blinking within the ITI. A blank screen is displayed for four seconds in the beginning and end of the experiment. For more details about this data, refer to [41].

5.2. Experiments and Settings

The proposed approach aims to build a discriminate method able to distinguish and classify EEG signal of a give motion. This section is dedicated to assessing the performance of the proposed detector in discriminating five distinct mental tasks based on EEG signals, namely mental word association, mental subtraction, spatial navigation, right-hand motor imagery, and feet motor imagery. After removing artifacts from EEG signals using the MWF procedure, QTFD is applied to generate a high-resolution time-frequency representation of the EEG signals and catch the EEG signals' spectral variations over time. After that,

the extracted QTFD features are used as input to the proposed approach for EEG signals classification. The study aims to build an unsupervised discriminate method able to distinguish and classify EEG signals of a given motion—specifically; there are five classes (five distinct mental tasks), namely mental word association, mental subtraction, spatial navigation, right-hand motor imagery, and feet motor imagery. The proposed approach addresses the problem of classification as a multi anomaly detection problem.

The dataset used consists of five classes, and we create a sub-set composed of only data points of a given class (target). Further, the training dataset is composed of 80%, a sub-set of the target class, and 20% will be used for the testing. Moreover, we create a testing data set composed of two parts 20% of the remaining data of the target class (as a normal observation: inliers), and we selected randomly 20% from the other classes (as abnormal or anomalies: outliers) to construct a testing dataset containing an amount of data of all classes. We repeat this procedure (testing) for all classes. This paper integrates a DBN model for feature extraction and an IF approach to recognize a given EEG signal class. In other words, we design a DBN-based IF detector for each class. This study is conducted using an ordinary PC with CPU i7 and 12Go RAM based on Ubuntu 20. The investigated methods are implemented using Python. Specifically, TensorFlow 2.3 and Keras 2.3 are used to implement the DBN, and Scikit-learn 1.1. to perform the detection via the Isolation Forest algorithm.

In the training phase, we fine-tune the parameters of the proposed approach using the training data via the grid search procedure. Specifically, the DBN model training employs a greedy layer-wise training procedure. The selected DBN model comprises three layers (30, 15, 5) hidden units, i.e., two stacked RBMs. Here, we adopted the following hyper-parameters: the number of Gibbs steps performed is 5, the learning rate used is 0.001, the number of epochs during the training is 1000 with a batch size of 250. Essentially, we train each RBM first in an unsupervised way, and then the IF detector will be applied to the output of the DBN model. The values of IF parameters are chosen in training so that the number of false alarms is reduced. The IF model constructs multiple isolation trees; in our study, we used $N = 150$ the number of trees and the size of the sub-samples 256. We compare the computed iF-driven anomaly score to a detection threshold of 0.5. If the calculated anomaly score overpass 0.5, it is a confirmed anomaly; otherwise, it is considered a normal observation. However, a special case may occur when the anomaly score is equal to 0.5 or close to 0.5, and this situation is considered an uncommon normal observation.

In this study, we considered two other commonly used anomaly detection schemes, namely Elliptical Envelope (EE) [43] , and Local Outlier Factor (LOF) [14]. Specifically, we compare the performance of the proposed DBN-based IF approach with that of DBN-based DCF and LOF methods. Similar to the DBN-IF, in DBN-EE and DBN-LOF, the EE and LOF detectors are applied to the extracted features from DBN to distinguish and classify the EEG signal of a given motion. We construct a model using training data based on unsupervised learning for each EEG signal class. In the LOF approach, the anomaly score is calculated for each data point by calculating the local divergence of the density of a given sample in comparison to its neighbors. Here, the number of neighbors in LOF is 35. On the other hand, the EE approach fits an ellipse around the data utilizing a minimum covariance determinant (MCD). In this experiment, the proportion of points to be enclosed in support of the raw MCD estimation is 0.9.

5.3. Discussion and Analysis

Now, we applied the trained DBN-iF model to the testing data to identify five distinct mental tasks based on EEG signals. As discussed above, we addressed this multiclass classification problem as multiple binary anomaly detection problems. Note that features characterizing the other classes are viewed as anomalies or outliers by the DBN-iF detector, and features from the targeted class are considered normal and should not be flagged as an outlier during the testing phase. Indeed, a single DBN-iF detector is constructed based on training data of each class, with the samples of that class as inlier and all other

samples as anomalies (i.e., one-vs.-rest). Detection results of the DBN-iF are summarized in Table 1. The column *class* in Table 1 represents the target class, meaning that the training data contains only the data of this class, while the column *Others* denotes the remaining classes where their data are used for the testing to evaluate the performance of the instances of the hybrid model used fro the target class. Table 1 reveals the high capacity of the DBN-iF in identifying data from the first class (WORD) from other classes (SUB, NAV, HAND, and FEET) by obtaining a high ACU values of $AUC_2 = 0.9779$, $AUC_3 = 0.9870$, $AUC_4 = 0.9902$, and $AUC_5 = 0.9840$. This means that the DBN-iF detector can efficiently recognize the first task from other tasks in other classes. Furthermore, results in Table 1 show that class 2 detection performance is very high, where the average AUC is 0.9918 for all classes. The averaged AUC values achieved by the DBN-iF detector for the third, fourth, and fifth classes are 0.9895, 0.9685, and 0.99, respectively. It can be observed from this first experiment that the DBN-iF approach provided satisfactory identification of mental tasks by EEG signals. The obtained high performance demonstrates the efficiency of the amalgamation of the deep learning model, DBN, with the Isolation Forest detector in recognizing mental tasks (e.g., motor imagery, calculation, reading) based on EEG signals. This could be attributed to different factors, including (i) the extended capacity of the QTFD in extracting spectral variations of the EEG signals, (ii) the flexibility of robustness of the DBN to approximate the data distribution of the underlining EEG signal class through creating a latent space that represents much better the original data (EEG signal) separately, and (iii) the sensitivity of the iF scheme in detecting anomalies through branching paths. In addition, this approach focuses on each class separately, making it easy to be separated from other classes.

Table 1. DBN-driven iF identification results.

CLASS	OTHERS	Accuracy	Precision	F1-Score	AUC
1	2	0.9779	0.9666	0.9781	0.9779
1	3	0.9870	0.9841	0.9870	0.9870
1	4	0.9902	0.9905	0.9902	0.9902
1	5	0.9840	0.9782	0.9840	0.9840
2	1	0.9918	0.9936	0.9918	0.9918
2	3	0.9912	0.9925	0.9912	0.9912
2	4	0.9924	0.9948	0.9924	0.9924
2	5	0.9920	0.9941	0.9920	0.9920
3	1	0.9917	0.9934	0.9917	0.9917
3	2	0.9867	0.9836	0.9867	0.9867
3	4	0.9909	0.9918	0.9908	0.9909
3	5	0.9892	0.9883	0.9892	0.9892
4	1	0.9806	0.9716	0.9808	0.9806
4	2	0.9593	0.9327	0.9605	0.9593
4	3	0.9762	0.9636	0.9766	0.9762
4	5	0.9580	0.9304	0.9593	0.9580
5	1	0.9910	0.9921	0.9910	0.9910
5	2	0.9881	0.9862	0.9881	0.9881
5	3	0.9899	0.9898	0.9899	0.9899
5	4	0.9917	0.9934	0.9917	0.9917

This study compared three different outlier detection models for mental task identification by EEG signals: DBN-based iF, LOF, and EE schemes. Results of DBN-based LOF and EE schemes are listed in Tables 2 and 3. From Table 2, we observe that the DBN-LOF scheme recognized classes 3 and 5 with high accuracy with an average AUC score of 0.9689 and 0.9768, respectively. However, it achieved relatively moderate identification of metal tasks related to classes: 1, 2, and 4 with the average AUC of 0.9208, 0.9382, and 0.9120,

respectively. Results in Table 3 indicate that EE performance was high for classes : 3 and 5 with an AUC average great than 0.95, while it is lower than 0.9 with 0.8908, 0.8889, and 0.8429 for classes 1, 2, and 4, respectively. Of course, from Tables 2 and 3 we can assume that DBN-LOF scheme outperformed the DBN-EE scheme.

Table 2. DBN-driven LOF identification results.

CLASS	OTHERS	Accuracy	Precision	F1-Score	AUC
1	2	0.9083	0.8498	0.9154	0.9083
1	3	0.9172	0.8628	0.9229	0.9172
1	4	0.9779	0.9647	0.9782	0.9779
1	5	0.8799	0.8103	0.8920	0.8799
2	1	0.8983	0.8345	0.9072	0.8983
2	3	0.9321	0.8847	0.9360	0.9321
2	4	0.9621	0.9346	0.9632	0.9621
2	5	0.9604	0.9317	0.9617	0.9604
3	1	0.9607	0.9313	0.9620	0.9607
3	2	0.9588	0.9279	0.9602	0.9588
3	4	0.9951	0.9955	0.9951	0.9951
3	5	0.9611	0.9320	0.9623	0.9611
4	1	0.9050	0.8463	0.9124	0.9050
4	2	0.8979	0.8363	0.9065	0.8979
4	3	0.9383	0.8975	0.9413	0.9383
4	5	0.9070	0.8492	0.9141	0.9070
5	1	0.9478	0.9107	0.9501	0.9478
5	2	0.9866	0.9805	0.9867	0.9866
5	3	0.9850	0.9774	0.9851	0.9850
5	4	0.9876	0.9823	0.9876	0.9876

Table 3. DBN-driven EE identification results.

CLASS	OTHERS	Accuracy	Precision	F1-Score	AUC
1	2	0.9187	0.8664	0.9241	0.9187
1	3	0.8995	0.8383	0.9078	0.8995
1	4	0.9015	0.8411	0.9095	0.9015
1	5	0.8436	0.7659	0.8636	0.8436
2	1	0.7846	0.7017	0.8213	0.7846
2	3	0.9122	0.8567	0.9185	0.9122
2	4	0.9421	0.9035	0.9447	0.9421
2	5	0.9168	0.8637	0.9225	0.9168
3	1	0.9606	0.9350	0.9617	0.9606
3	2	0.9716	0.9550	0.9722	0.9716
3	4	0.9905	0.9912	0.9905	0.9905
3	5	0.9718	0.9553	0.9723	0.9718
4	1	0.9172	0.8643	0.9228	0.9172
4	2	0.7579	0.6762	0.8035	0.7579
4	3	0.9134	0.8585	0.9195	0.9134
4	5	0.7833	0.7004	0.8204	0.7832

Table 3. *Cont.*

CLASS	OTHERS	Accuracy	Precision	F1-Score	AUC
5	1	0.9126	0.8574	0.9189	0.9126
5	2	0.9769	0.9648	0.9772	0.9769
5	3	0.9790	0.9687	0.9792	0.9790
5	4	0.9705	0.9528	0.9710	0.9705

Table 4 shows the aggregated AUC obtained by three investigated approaches: DBN-iF, DBN-LOF, and DBN-EE. Figure 8 displays the barplot of averaged AUC to visually aid the comparison of achieved results by the three considered schemes. Figure 9 illustrates the aggregated results of those shown in Table 4.

Table 4. Overall detection results of all anomaly detection methods.

CLASS	iF	LOF	EE
1	0.9848	0.9208	0.8908
2	0.9919	0.9382	0.8889
3	0.9896	0.9689	0.9736
4	0.9685	0.9120	0.8429
5	0.9902	0.9768	0.9597

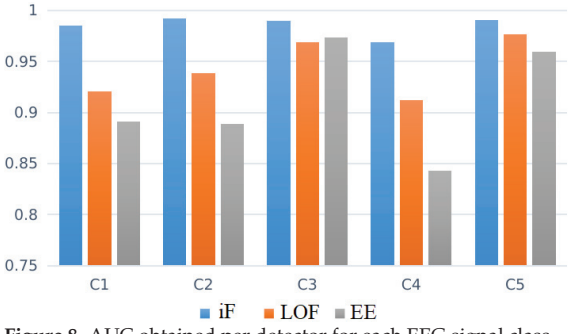


Figure 8. AUC obtained per detector for each EEG signal class.

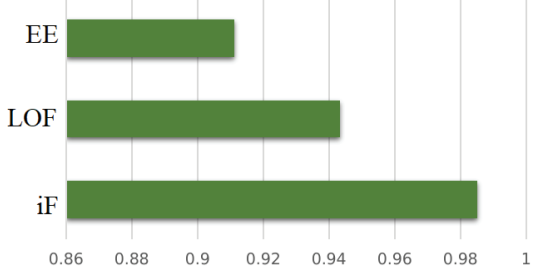


Figure 9. Averaged AUC obtained per anomaly detector for all EEG signal classes.

It would appear, based on results in Table 4 Figure 8 that DBN-iF would be the best model for mental task recognition based on EEG signals. It dominates the other models (DBN-LOF and DBN-EE) by better identifying mental tasks from the acquired EEG signals.

5.4. Comparison with the State-of-the-Art

Lastly, to measure the real contribution of the present work, comparisons with some existing and recent systems conducted on EEG classification datasets are reported in Table 5. Several powerful classifiers are invoked, namely: least squares support vector machine [3,6], convolutional neural networks and stacked autoencoders [23], KNN [8], logistic regression [11], and LR KLR, MLP, PNN, and LS-SVM [7]. From Table 5, the results demonstrate the outperformance of the proposed approach over classifying EEG signals by the state-of-the-art approaches, even in critical scenarios. The proposed approach dominates the other methods mainly because of its capability to recognize each class individually and approximate the data distribution by the generation of latent space, which is suitable in the representation of original data.

In the case of the logistic regression formalism [11] (applied alone or kernalised LR), one can observe some misclassification cases, which impacted the recognition rate (lower than 95%). In [23], a CNN-SAE combination has been used. One can mention the presence of numerous misclassification cases, which affects the accuracy presenting the lowest classification accuracy (90%). This could be explained by CNN formalism being more adapted for image or matrix classification (data in 2D form) than signal classification (EEG samples). Importantly, transforming EEG signals from 1D to 2D to obtain matrix representation can cause the loss of some pertinent information. In [3,6,7,11], an LS-SVM combination formalism has been used as a recognition algorithm, which achieved a relatively high accuracy of around 96%. Even if SVMs formalism is based on geometric aspect, where the samples' separation is established on the sparse solution via structural risk minimization, the DBN classification has outperformed the LS-SVM combinations. In summary, all of these observations confirm that the DBN formalism is more adapted to EEG signals' classification than several existing methods.

Table 5. Comparison of the proposed method with existing methods.

Paper	The Used Features	Approach	Accuracy (%)
[6]	EMD	LS-SVM	97.56
[3]	TQWT	LS-SVM	96.89
[23]	STFT and electrode location information	CNN-SAE	90
[8]	MSPCA, DWT and WPD	KNN	92.8
[11]	cross-correlation and DWT coefficients	LR	92.3
		KLR	94.3
		MLP	94.9
		PNN	92.9
		LS-SVM	96.1
[7]	Optimal allocation features	LS-SVM	96.62
		Naive Bayes	96.36
[44]	EEG-inception (time-series signals) with data augmentation	CNN	88.58
[45]	Semantic, intrinsic, and user-specific features (with data augmentation)	multi-scale CNN	93.74
This study	QTFD	DBN-iF	98.5

6. Conclusions

This study introduced an unsupervised deep learning-based strategy for discriminating the mental tasks of EEG signals. Essentially, multiclass classification is handled as an anomaly detection problem without using labeled data (i.e., fully unsupervised). After removing artifacts from EEG signals, a time-frequency representation of the EEG signals is obtained using the QTFD and used as input of the designed detector to improve mental tasks' recognition. The iF anomaly detection scheme is applied to the features extracted by the DBN model for separating mental tasks from the EEG signals. This approach is assessed on publicly available benchmark EEG datasets comprising five classes. We compared the DBN-iF approach with two other unsupervised detection approaches, DBN-based EE and LOF. Results demonstrated that the DBN-iF delivers superior discrimination performance of mental tasks by EEG Signals and dominates the investigated methods, DBN-CDE and DBN-LOF. It has demonstrated that merging the DBN deep learning model with anomaly detection methods presents a promising strategy to mitigate challenges in EEG-based mental tasks' recognition.

Despite the encouraging results for MI-based tasks recognition obtained using the EEG-based mental tasks recognition via a deep learning-driven anomaly detector approach, this work raises some directions of improvement that merit consideration from researchers for future work. In particular, the extended recognition of EEG signals is to perform tasks and establish possible communication between several patients via BCI applications.

In terms of methodology, this study demonstrated that the proposed DBN-driven Isolation Forest approach achieved satisfactory discrimination results. However, the DBN model does not consider feature selection, and the considerable amount of irrelevant data in high-dimensional data can constrain its performance. Therefore, we plan to develop an improved DBN model that focuses only on relevant features by integrating attention mechanisms within the DBN model; the attention mechanism allows the model to focus on essential features [46].

On the other hand, the Isolation Forest algorithm is computationally efficient and has proven effective in anomaly detection. However, its final anomaly score depends on the contamination parameter provided during the training stage. This implies that we need to have an idea of what percentage of the training data is anomalous beforehand to obtain a better prediction. In addition, it has the disadvantage of sensing local anomaly points, which influences the algorithm's precision [47]. As an alternative, we plan in future work to develop deep model-driven statistical monitoring schemes by merging the capacity of generative deep learning models, such as generative adversarial networks (GANs) [48] and variational autoencoders (VAE) [49], to find low-dimensional summaries that will be monitored by statistical monitoring charts, such as the generalized likelihood ratio (GLR) test [50].

Author Contributions: A.D., Conceptualization, formal analysis, investigation, methodology, software, supervision, writing—original draft, and writing—review and editing. N.Z., Conceptualization, formal analysis, investigation, methodology, software, supervision, writing—original draft, and writing—review and editing. F.H., Conceptualization, formal analysis, investigation, methodology, software, supervision, writing—original draft, and writing—review and editing. Y.S., Conceptualization, formal analysis, investigation, methodology, software, supervision, writing—original draft, and writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by funding from the King Abdullah University of Science and Technology (KAUST), Office of Sponsored Research (OSR), under Award No: OSR-2019-CRG7-3800.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: This work was supported by funding from the King Abdullah University of Science and Technology (KAUST), Office of Sponsored Research (OSR), under Award No.: OSR-2019-CRG7-3800.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, Y.; Guo, L.; Liu, Y.; Liu, J.; Meng, F. A Temporal-Spectral-Based Squeeze-and-Excitation Feature Fusion Network for Motor Imagery EEG Decoding. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2021**, *29*, 1534–1545. [CrossRef]
2. Yang, L.; Song, Y.; Ma, K.; Xie, L. Motor imagery EEG decoding method based on a discriminative feature learning strategy. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2021**, *29*, 368–379. [CrossRef] [PubMed]
3. Taran, S.; Bajaj, V. Motor imagery tasks-based EEG signals classification using tunable-Q wavelet transform. *Neural Comput. Appl.* **2019**, *31*, 6925–6932. [CrossRef]
4. Dai, M.; Wang, S.; Zheng, D.; Na, R.; Zhang, S. Domain Transfer Multiple Kernel Boosting for Classification of EEG Motor Imagery Signals. *IEEE Access* **2019**, *7*, 49951–49960. [CrossRef]
5. Duan, L.; Li, J.; Ji, H.; Pang, Z.; Zheng, X.; Lu, R.; Li, M.; Zhuang, J. Zero-Shot Learning for EEG Classification in Motor Imagery-Based BCI System. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2020**, *28*, 2411–2419. [CrossRef]

6. Taran, S.; Bajaj, V.; Sharma, D.; Siuly, S.; Sengur, A. Features based on analytic IMF for classifying motor imagery EEG signals in BCI applications. *Measurement* **2018**, *116*, 68–76. [CrossRef]
7. Siuly, S.; Li, Y. Discriminating the brain activities for brain–computer interface applications through the optimal allocation-based approach. *Neural Comput. Appl.* **2015**, *26*, 799–811. [CrossRef]
8. Kevric, J.; Subasi, A. Comparison of signal decomposition methods in classification of EEG signals for motor-imagery BCI system. *Biomed. Signal Process. Control* **2017**, *31*, 398–406. [CrossRef]
9. Gaur, P.; Gupta, H.; Chowdhury, A.; McCreadie, K.; Pachori, R.B.; Wang, H. A Sliding Window Common Spatial Pattern for Enhancing Motor Imagery Classification in EEG-BCI. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–9. [CrossRef]
10. Zhang, S.; Zhu, Z.; Zhang, B.; Feng, B.; Yu, T.; Li, Z. Fused Group Lasso: A New EEG Classification Model With Spatial Smooth Constraint for Motor Imagery-Based Brain–Computer Interface. *IEEE Sens. J.* **2021**, *21*, 1764–1778. [CrossRef]
11. Verma, N.K.; Rao, L.V.S.; Sharma, S.K. Motor imagery EEG signal classification on DWT and crosscorrelated signal features. In Proceedings of the 2014 IEEE 9th International Conference on Industrial and Information Systems (ICIIS), Gwalior, India, 15–17 December 2014; pp. 1–6.
12. Chatterjee, R.; Maitra, T.; Islam, S.H.; Hassan, M.M.; Alamri, A.; Fortino, G. A novel machine learning based feature selection for motor imagery EEG signal classification in Internet of medical things environment. *Future Gener. Comput. Syst.* **2019**, *98*, 419–434. [CrossRef]
13. Liu, F.T.; Ting, K.M.; Zhou, Z.H. Isolation forest. In Proceedings of the 2008 IEEE Eighth IEEE International Conference on Data Mining, Pisa, Italy, 15–19 December 2008; pp. 413–422.
14. Breunig, M.M.; Kriegel, H.P.; Ng, R.T.; Sander, J. LOF: Identifying density-based local outliers. *ACM Sigmod Rec.* **2000**, *29*, 93–104. [CrossRef]
15. Hubert, M.; Debruyne, M.; Rousseeuw, P.J. Minimum covariance determinant and extensions. *Wiley Interdiscip. Rev. Comput. Stat.* **2018**, *10*, e1421. [CrossRef]
16. Harrou, F.; Sun, Y.; Hering, A.S.; Madakyaru, M. *Statistical Process Monitoring Using Advanced Data-driven and Deep Learning Approaches: Theory and Practical Applications*; Elsevier: Amsterdam, The Netherlands, 2020.
17. Dairi, A.; Harrou, F.; Sun, Y. Deep Generative Learning-based 1-SVM Detectors for Unsupervised COVID-19 Infection Detection Using Blood Tests. *IEEE Trans. Instrum. Meas.* **2021**, *71*, 2500211. [CrossRef]
18. Wang, W.; Lee, J.; Harrou, F.; Sun, Y. Early detection of Parkinson's disease using deep learning and machine learning. *IEEE Access* **2020**, *8*, 147635–147646. [CrossRef]
19. Bashivan, P.; Rish, I.; Yeasin, M.; Codella, N. Learning representations from EEG with deep recurrent-convolutional neural networks. *arXiv* **2015**, arXiv:1511.06448.
20. Liang, N.Y.; Saratchandran, P.; Huang, G.B.; Sundararajan, N. Classification of mental tasks from EEG signals using extreme learning machine. *Int. J. Neural Syst.* **2006**, *16*, 29–38. [CrossRef]
21. Zeng, H.; Yang, C.; Dai, G.; Qin, F.; Zhang, J.; Kong, W. EEG classification of driver mental states by deep learning. *Cogn. Neurodyn.* **2018**, *12*, 597–606. [CrossRef]
22. Opałka, S.; Stasiak, B.; Szajerman, D.; Wojciechowski, A. Multi-channel convolutional neural networks architecture feeding for effective EEG mental tasks classification. *Sensors* **2018**, *18*, 3451. [CrossRef]
23. Tabar, Y.R.; Halici, U. A novel deep learning approach for classification of EEG motor imagery signals. *J. Neural Eng.* **2016**, *14*, 016003. [CrossRef]
24. Wang, P.; Jiang, A.; Liu, X.; Shang, J.; Zhang, L. LSTM-Based EEG Classification in Motor Imagery Tasks. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2018**, *26*, 2086–2095. [CrossRef]
25. Lu, N.; Li, T.; Ren, X.; Miao, H. A Deep Learning Scheme for Motor Imagery Classification based on Restricted Boltzmann Machines. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2017**, *25*, 566–576. [CrossRef]
26. Jeong, J.H.; Lee, B.H.; Lee, D.H.; Yun, Y.D.; Lee, S.W. EEG classification of forearm movement imagery using a hierarchical flow convolutional neural network. *IEEE Access* **2020**, *8*, 66941–66950. [CrossRef]
27. Zhang, D.; Chen, K.; Jian, D.; Yao, L. Motor Imagery Classification via Temporal Attention Cues of Graph Embedded EEG Signals. *IEEE J. Biomed. Health Inform.* **2020**, *24*, 2570–2579. [CrossRef]
28. Chaudhary, S.; Taran, S.; Bajaj, V.; Sengur, A. Convolutional Neural Network Based Approach Towards Motor Imagery Tasks EEG Signals Classification. *IEEE Sens. J.* **2019**, *19*, 4494–4500. [CrossRef]
29. Zhang, G.; Davoodnia, V.; Sepas-Moghaddam, A.; Zhang, Y.; Etemad, A. Classification of Hand Movements from EEG Using a Deep Attention-Based LSTM Network. *IEEE Sens. J.* **2020**, *20*, 3113–3122. [CrossRef]
30. Jiang, X.; Bian, G.B.; Tian, Z. Removal of artifacts from EEG signals: A review. *Sensors* **2019**, *19*, 987. [CrossRef]
31. Alazrai, R.; Alwanni, H.; Baslan, Y.; Alnuman, N.; Daoud, M.I. Eeg-based brain-computer interface for decoding motor imagery tasks within the same hand using choi-williams time-frequency distribution. *Sensors* **2017**, *17*, 1937. [CrossRef]
32. Hahn, S.L. *Hilbert Transforms in Signal Processing*; Artech House Signal Processing: Norwood, MA, USA, 1996.
33. Smolensky, P. Information Processing in Dynamical Systems: Foundations of Harmony Theory. In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*; MIT Press: Cambridge, MA, USA, 1987; pp.194–281.
34. Bengio, Y. Learning deep architectures for AI. *Found. Trends® Mach. Learn.* **2009**, *2*, 1–127. [CrossRef]
35. Hinton, G.E.; Osindero, S.; Teh, Y.W. A fast learning algorithm for deep belief nets. *Neural Comput.* **2006**, *18*, 1527–1554. [CrossRef]

36. Li, T.; Wang, B.; Jiang, Y.; Zhang, Y.; Yan, Y. Restricted Boltzmann machine-based approaches for link prediction in dynamic networks. *IEEE Access* **2018**, *6*, 29940–29951. [CrossRef]
37. Liu, F.T.; Ting, K.M.; Zhou, Z.H. Isolation-based anomaly detection. *ACM Trans. Knowl. Discov. Data (TKDD)* **2012**, *6*, 1–39. [CrossRef]
38. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
39. Mensi, A.; Bicego, M. Enhanced Anomaly Scores for Isolation Forests. *Pattern Recognit.* **2021**, *120*, 108115. [CrossRef]
40. Dairi, A.; Harrou, F.; Sun, Y. Efficient Driver Drunk Detection by Sensors: A Manifold Learning-Based Anomaly Detector. *IEEE Access* **2022**, *10*, 119001–119012. [CrossRef]
41. Scherer, R.; Faller, J.; Friedrich, E.V.; Opisso, E.; Costa, U.; Kübler, A.; Müller-Putz, G.R. Individually adapted imagery improves brain-computer interface performance in end-users with disability. *PLoS ONE* **2015**, *10*, e0123727. [CrossRef]
42. Homan, R.W.; Herman, J.; Purdy, P. Cerebral location of international 10–20 system electrode placement. *Electroencephalogr. Clin. Neurophysiol.* **1987**, *66*, 376–382. [CrossRef]
43. Rousseeuw, P.J.; Driessen, K.V. A fast algorithm for the minimum covariance determinant estimator. *Technometrics* **1999**, *41*, 212–223. [CrossRef]
44. Zhang, C.; Kim, Y.K.; Eskandarian, A. EEG-inception: An accurate and robust end-to-end neural network for EEG-based motor imagery classification. *J. Neural Eng.* **2021**, *18*, 046014. [CrossRef]
45. Roy, A.M. A CNN Model with Feature Integration for MI EEG Subject Classification in BMI. Available online: https://www.biorxiv.org/content/10.1101/2022.01.05.475058v3.full (accessed on 28 February 2022).
46. Dairi, A.; Harrou, F.; Khadraoui, S.; Sun, Y. Integrated multiple directed attention-based deep learning for improved air pollution forecasting. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–15. [CrossRef]
47. Gao, R.; Zhang, T.; Sun, S.; Liu, Z. Research and improvement of isolation forest in detection of local anomaly points. *J. Phys. Conf. Ser.* **2019**, *1237*, 052023. [CrossRef]
48. Kadri, F.; Dairi, A.; Harrou, F.; Sun, Y. Towards accurate prediction of patient length of stay at emergency department: A GAN-driven deep learning framework. *J. Ambient. Intell. Humaniz. Comput.* **2022**, 1–15. [CrossRef] [PubMed]
49. Harrou, F.; Dairi, A.; Zeroual, A.; Sun, Y. Forecasting of Bicycle and Pedestrian Traffic Using Flexible and Efficient Hybrid Deep Learning Approach. *Appl. Sci.* **2022**, *12*, 4482. [CrossRef]
50. Harrou, F.; Fillatre, L.; Bobbia, M.; Nikiforov, I. Statistical detection of abnormal ozone measurements based on constrained generalized likelihood ratio test. In Proceedings of the 52nd IEEE Conference on Decision and Control, Firenze, Italy, 10–13 December 2013; pp. 4997–5002.

diagnostics

MDPI

Article

Using Recurrent Neural Networks for Predicting Type-2 Diabetes from Genomic and Tabular Data

Parvathaneni Naga Srinivasu [1,*,†], Jana Shafi [2,†], T Balamurali Krishna [3], Canavoy Narahari Sujatha [4], S Phani Praveen [1] and Muhammad Fazal Ijaz [5,*]

[1] Department of Computer Science and Engineering, Prasad V. Potluri Siddhartha Institute of Technology, Vijayawada 520007, Andhra Pradesh, India
[2] Department of Computer Science, College of Arts and Science, Prince Sattam bin Abdul Aziz University, Wadi Ad-Dawasir 11991, Saudi Arabia
[3] Department of Computer Science and Engineering, Dhanekula Institute of Engineering and Technology, Vijayawada 521139, Andhra Pradesh, India
[4] Department of Electronics and Communication Engineering, Sreenidhi Institute of Science and Technology, Hyderabad 501301, Telangana, India
[5] Department of Intelligent Mechatronics Engineering, Sejong University, Seoul 05006, Republic of Korea
* Correspondence: parvathanenins@gmail.com (P.N.S.); fazal@sejong.ac.kr (M.F.I.)
† These authors contributed equally to this work and are the first co-authors.

Abstract: The development of genomic technology for smart diagnosis and therapies for various diseases has lately been the most demanding area for computer-aided diagnostic and treatment research. Exponential breakthroughs in artificial intelligence and machine intelligence technologies could pave the way for identifying challenges afflicting the healthcare industry. Genomics is paving the way for predicting future illnesses, including cancer, Alzheimer's disease, and diabetes. Machine learning advancements have expedited the pace of biomedical informatics research and inspired new branches of computational biology. Furthermore, knowing gene relationships has resulted in developing more accurate models that can effectively detect patterns in vast volumes of data, making classification models important in various domains. Recurrent Neural Network models have a memory that allows them to quickly remember knowledge from previous cycles and process genetic data. The present work focuses on type 2 diabetes prediction using gene sequences derived from genomic DNA fragments through automated feature selection and feature extraction procedures for matching gene patterns with training data. The suggested model was tested using tabular data to predict type 2 diabetes based on several parameters. The performance of neural networks incorporating Recurrent Neural Network (RNN) components, Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU) was tested in this research. The model's efficiency is assessed using the evaluation metrics such as Sensitivity, Specificity, Accuracy, F1-Score, and Mathews Correlation Coefficient (MCC). The suggested technique predicted future illnesses with fair Accuracy. Furthermore, our research showed that the suggested model could be used in real-world scenarios and that input risk variables from an end-user Android application could be kept and evaluated on a secure remote server.

Keywords: deep learning; PIMA dataset; Type-2 diabetes; Recurrent Neural Networks; weight optimization

Citation: Srinivasu, P.N.; Shafi, J.; Krishna, T.B.; Sujatha, C.N.; Praveen, S.P.; Ijaz, M.F. Using Recurrent Neural Networks for Predicting Type-2 Diabetes from Genomic and Tabular Data. *Diagnostics* **2022**, *12*, 3067. https://doi.org/10.3390/diagnostics12123067

Academic Editors: Wan Azani Mustafa and Hiam Alquran

Received: 4 November 2022
Accepted: 4 December 2022
Published: 6 December 2022

1. Introduction

Diabetes is a metabolic disorder influenced by high blood sugar levels due to insufficient insulin release or synthesis. Diabetes was predicted to affect 285 million people globally in 2010. According to the disease's current development pace, this figure will increase to 552 million by 2030. One in every ten people is projected to have diabetes by 2040 [1]. Diabetes is becoming more prevalent because of individual habits, divergent lifestyles, and living standards. Thus, researching how to effectively and promptly identify and treat diabetes is worthwhile. Diabetes is diagnosed based on genomic patterns. It will

result in a more accurate and precise outcome and assist in adhering to better habits that are less likely to result in diabetes shortly. Very effective identification of an illness allows individuals with future illnesses to slow or postpone the disease's development and enjoy better overall health. Machine learning techniques fall into two categories: screening future illnesses and diagnosing an abnormality [2]. Based on current and prior medical conditions, forward prediction techniques can anticipate diabetes before it occurs [3,4].

Type 2 diabetes (T2D), formerly called non-insulin-dependent diabetes) is a category of metabolic disorders distinguished through hyperglycemia, resulting in abnormalities in insulin production or insulin function. Lifestyle behaviors, including food habits, exercise, and dietary choices, may significantly affect its development. T2D is the type of disease known for decreasing the life span and reducing the standard of living. The illness may be controlled with lifestyle modification and pharmaceutical management. Thus, it is essential to have early diagnosis and treatment of T2D to help patients avoid life-threatening consequences. Many research studies have been conducted on medical diagnoses to predict illness and forecast the future with considerable efficiency accurately. Generally, most diseases are triggered by a combination of two or more gene patterns. Recognition of the combinational gene sequence by rigorous analysis of the reference genes of the healthy person with the samples that are trained genes acquired from the diseased. Deoxyribonucleic Acid (DNA) is significant for cell growth and is generally a hereditary component in each cell of an organism. As stated by A. Arshad and Y. D. Khan [5], DNA is coded through chemical bases adenine (A), guanine (G), cytosine (C), and thymine (T) that form up a cell which is unique for almost all human beings. Analysis of DNA molecular compositions of human genes is immensely used to predict illnesses associated with ancestors. The genomics study would assist in changing the lifestyle of an individual, which results in a lower risk of the disease in the future. DNA analysis can assist in the prediction of a disease that is caused by a mutation of the DNA. Biomedical engineering has recognized an enormous gene data set that could help predict various conditions. The Neural Network approach can identify gene patterns that harm cells of a human body with a high possibility of illness-causing patterns through the proposed mechanism. By incorporating Neural Networks, approaches would have high Accuracy for illness prediction with reasonably acceptable computational latency.

The advancement that has taken place in Genome-Wide Association Studies (GWAS) holds tremendous information related to various gene patterns associated with divergent illnesses that are complex and challenging to perform reductive analysis from a single locus, as stated by Cho Ys [6] and Coron [7]. The evolution of GWAS has focused on integrating data related to multi-locus across the gene that would assist in predicting complex illnesses in advance. Polygenic Risk Scores (PRS) were proposed by Duncan L. et al. [8] for risk analysis using the sum of the weight of each risk-associated locus of genomic sequence obtained from the corresponding evidence. These weights are assessed from the regression coefficient associated with each locus. These combined genetics features and correlation matrices would significantly assist the entire field of genomics study [9]. These studies on analyzing the genomic data and the tabular datasets such as PIMA would largely assist in analyzing the future illness had paved the motivation for the current study, and the role of various neural network components in the performance of the deep learning models are evaluated [10,11].

The current study is primarily motivated by the research challenges in handling genomic data and the pattern recognition for precisely identifying future illnesses. The genomic data comprises more extensive DNA sequences, which requires tremendous computational efforts to assess the disease's probability. Earlier assessment of the future illness would assist the individual in safeguarding themselves from such disease through better living standards. Moreover, the current study has also focused on evaluating the performances of various recurrent neural network models such as RNN, GRU, and LSTM in disease prediction. Performances evaluation metrics such as the confusion matrix

for Sensitivity, Specificity, F1-Score, and Mathews Correlation Coefficient measures are considered in the current study.

The recurrent neural network components such as RNN, GRU, and LSTM are efficient in learning from past experiments and can simultaneously process a sequence of inputs and outputs, which is a kind of sequence-to-sequence network that is exceptionally efficient in handling genomic data. The RNN-based neural network may represent a set of records such that each pattern is thought to rely on preceding ones. The Hidden State, which remembers certain information about a sequence, is the core and most essential aspect of RNN. LSTM feeds genetic sequences into a network and makes assumptions based on the sequence data's discrete time steps. It can learn long-term dependencies, which is notably valuable for sequence prediction issues. GRU uses less memory and is comparatively faster than the RNN and LSTM models. But can effectively work with smaller sequences. GRU employs fewer training parameters, requires less memory, executes quicker, and learns faster than LSTM, although LSTM is much more accurate on more extended sequence datasets.

The main contributions of this work are as follows:

- The reference gene sequence is analyzed against the trained genomic data for possible gene pattern matching. As well, the further correlation between the reference gene and gene pattern associated with diabetes is assessed.
- The probabilistic estimations are performed by the softmax layer towards the future illness based on the gene correlation. Additionally, based on the probabilities, the risk factor outcome is yielded.
- The proposed RNN model is evaluated over the tabular patient data such as PIMA for risk analysis, where the auxiliary memory components such as GRU and LSTM are integrated for better prediction performance.
- The feature selection and weight optimizations are performed over the features of the PIMA dataset for better prediction outcomes.
- The outcome of the present study is being evaluated against conventional classification techniques such as Decision Tree, J48, K Nearest Neighborhood, Logistic Regression, Naive Bayes, Random Forest, and Support Vector Machine.

The entire paper is arranged as follows. The paper's first section introduces the proposed approach and the Genomic domain. Section 2 presents the literature review of existing studies focusing on various disease prediction techniques using genomic and tabular data. Section 3 presents the methodology of the proposed model where various aspects such as background work of the domain that highlights data collection, data preprocessing, feature extraction strategies, and RNN with various memory components are presented. Section 4 presents the result and discussion. Section 5 presents the conclusion and future scope of the proposed model.

2. Literature Review

2.1. ML Models for Smart Diagnosis of Type-2 Diabetes

Machine Learning is the most emerging technology for addressing inevitable problems in various domains. Machine Learning through supervised, semi-supervised approaches, or weakly supervised approaches, is used with data from various sources, including medical records and information obtained from wearable gadgets, to forecast an illness. In either of these approaches, sickness cannot be predicted much earlier, and the patient cannot get rid of the illness by changing his or her lifestyle in a short period. The polygenic scores-based approach is among the most predominantly used strategies for the earlier prediction of an illness. The Polygenic Score approach has been tremendously evaluated before it is used in clinical trials. It is also used in illness screening mechanisms, as in the study of So et al. [12]. The current research and the genomic analysis could change lifestyles and reduce illnesses such as heart attack, cardiovascular diseases, cancer, and Alzheimer's disease. The process of polygenic risk score involves two significant phases: discovery and validation. The Discovery Phase identifies risks through a statistical association test using either linear or

Logistic Regression. The later phase validates approximations performed in the earlier stage for extracting information related to Single Nucleotide Polymorphism (SNP).

Deep learning (DL) [13,14] is the field of Machine Learning that is extensively used in predicting type-2 disease by processing the blood glucose level and spectrogram images generated from the blood glucose levels. Moreover, the DL models could also be used with tabular datasets such as PIMA for the prediction of diabetes. Every layer in the DL model reflects a degree of acquired information. The layer closest to the input layer reflects low-level data elements, whereas the layer closest to the output layer shows a higher degree of discrimination with more concise notions. Deep learning generally needs more data for precise classification and also needs tremendous computational resources for processing [15]. The major limitation of the deep learning models is that the decision mechanism is not interpretable, which limits the trustworthiness of the models.

Clustering is one of the most predominantly performed operations with un-supervisory approaches using dimensionality reduction approaches such as Singular Value Decomposition (SVD) [16], Principle Component Analysis (PCA) stated by Konishi T. et al. [17], Apriori stated by S. Mallik et al. [18]. Dynamic thresholding-based FP-Growth was stated by Mallik S. et al. [19] for treating unusual illnesses and certain types of diseases with unknown variations with different symptoms. However, most of these approaches do not label the output data, as the provided input does not have any labels. The Accuracy of the un-supervisory method is a significant concern as classes are not marked. In some instances, the proposed algorithm might end up misinterpretation. A classification-based illness prediction is a supervisory approach that includes various mechanisms such as Linear and Polynomial Regression, Decision Tree, Random Forest, and many other systems, including the Support Vector Machine (SVM) used by Huang S. et al. [20], K-Nearest Neighbour used by Parry R. et al. [21], and Logistic Regression approach that exhibits better efficiency in terms of accuracy and precision other classification models. Supervisory approaches exhibit optimal performance for known cases. The Accuracy of the prediction outcome is directly proportional to the training set size, which needs many computational efforts. However, in some cases, the approach diverges due to excessive training.

Random Forest is a rapid implementation approach using the Ranger package in the R tool described by Wright and Ziegler [22], which is used to predict future illness from tabular data such as PIMA. Artificial Neural Networks based on illness prediction mechanisms, as discussed by Anifat O. et al. [23] and Mantzaris D. et al. [24] involve a more profound architecture that includes input and output layers alongside multiple hidden layers to process records iteratively, moving data among layers that would minimize the loss function and acquaint weights and biases of each layer. Various ensemble approaches, such as random forest and boosting, have been experimented with as alternatives to machine learning approaches for predicting future illness. Exponential research has been conducted using either of those approaches with real-time and simulated data. The ensemble approaches work faster for classification when compared to the conventional classification models. However, either of the models ends up with non-additive issues. The resultant effect in the forward direction of the layers would determine the predictive analysis, and the backward pass would assess the standard error among the prediction made and the ground facts.

2.2. Deep Learning for Type-2 Diabetes

The Deep Learning (DL) model implements the framework that infers target gene expression obtained from the expression of landmark genes. Utilizing 111,000 individual gene patterns over a Gene expression Omnibus2, Deep Neural Network-based Gene Analysis model (D-GEX) trained a feedforward neural network through three hidden layers. DL models outperform linear Regression in summarizing expression levels of over 21,000 human genes based on a collection of landmark genes with about 1000 sequences. Although the DL model is more accurate than conventional classification models, performance is not adequate in the healthcare domain, where the design of DL models needs to be improved. The deepVariant model outperforms all other recent neural network models [25]. Deep-

Variant generalizes its training samples by utilizing various human genome expressions as train and test datasets.

Additionally, when training with human gene expressions and evaluating with a mouse genomic expressions dataset, DeepVariant obtained Accuracy that outperformed training with mouse data. DeepFIGV is a deep learning algorithm that uses DNA sequences to predict locus-specific signals from epigenetic tests. DeepFIGV quantifies epigenetic variance by employing several investigations with similar cell patterns and experiments [26]. It combines the entire gene sequence to provide a customized genetic line for each person. The Gene Co-Expression model is a differential network analysis model extensively used in gene data analysis to identify gene sequence similarities and topologies [27]. This model considers two classes of the gene through which the classification model is implemented. However, the Gene Co-Expression model has to deal with comparatively larger features than the size of the data and the non-linearity of the network architecture, where dependencies would make it difficult to trust the model's predictions. Reinforcement Learning (RL) based intelligent systems such as Q-Learning, State Action Reward State Action (SARSA), Deep Deterministic Policy Gradient (DDPG), and Deep Q Network (DQN), as stated by Travnik Jaden B. et al. [28] are the most suitable for handling healthcare to better forecast a future illness with minimal training of the algorithm recovers. The underlying technology remains the same with minimal training. The algorithm is mechanized to learn from its previous experiences.

Various studies have been presented to predict future illness through existing patient data using machine learning algorithms. Predicting future illness has become a demanding topic in healthcare [29]. Several studies have used machine intelligence techniques to analyze the Pima Indian Diabetes Dataset. C. Yue [30] has investigated various hybrid approaches, including Neural Networks, integrated Quantum Particle Swarm Optimization (QPSO), and Weighted Least Square (WLS) Support Vector Machine (SVM) for diabetes prediction, with the WLS-SVM hybrid model showing a classification accuracy of 82.18%. However, the hybridization model needs considerable effort in the evaluation process. In addition, the SVM model is not suitable for working with larger data [31]. Moreover, the SVM model underperforms if the number of attributes for every data point exceeds the training samples. The combinational models for diabetes prediction using Cross-validation and Self-Organizing Maps (SOM) have achieved an accuracy of 78.4% [32,33]. SOM can rely on the associated weights of neurons for precise classification. Inappropriate assignment of initial weights may impact the model's performance. A C4.5 technique [34] has been used to analyze the PIMA dataset, attaining an Accuracy of 71.1%. The model works through the entropy value associated with the feature vector. The conventional classification models exhibit poor performance when working with distinct feature vectors [35].

A fuzzy entropy approach for feature selection for a similarity classifier has been evaluated against various medical datasets, such as Pima–Indian diabetes, exhibiting an accuracy of 75.29% [36]. A fuzzy model primarily depends on the membership evaluation that requires considerable effort. Non-linearity in evaluating the model will limit the model's performance [37]. Genetic Algorithm (GA) with Radial Basis Function Neural Network (RBF NN) has been used in the evaluation process of diabetes data, exhibiting an accuracy of 77.39% over the testing dataset [38]. Moreover, for artificial evolutionary algorithms such as GA, the most prohibitive and restricting element is frequently repeated fitness function assessment for complex gene patterns. Hybridization of models with GA would need more computational efforts than neural networks alone. Various cutting-edge technologies for the classification and prediction of type-2 diabetes are presented in Table 1.

Table 1. Various existing models for diabetes prediction.

Approach	Type of Data	Applicability	Limitations
polygenic scores-based approach [12]	Genomic Data	Used in the evaluation of clinical trials and illness screening mechanisms	The polygenic score approach needs larger samples and tremendous training for considerable Accuracy.
Singular Value Decomposition [13]	Genomic Data Tabular Data The image they are used	They are used in ranking the feature set and compression of the data through the least-square fitting. Gene sequences are ranked based on the probability of illness.	SVD is not an algorithm designed to perform; it is a matrix decomposition mechanism. They are various neural ranking models that perform much better than SVD.
Principle Component Analysis [14]	Genomic Data Tabular Data	PCA technique is extensively used in gene analysis to discover the regional and ethnic patterns of genetic variation.	The independent gene expressions are less interpretable, and information loss is possible if the number of components is carefully chosen.
Gene Co-Expression model [27]	Genomic Data	The Gene Co-Expression model analyzes the genomic data's insights through similarity assessment of expressions and topologies.	The Gene Co-Expression model may not deal with larger features than the data size and non-linearity in the network architecture.
Reinforcement approaches (SARSA, DDPG, DQN) [28]	Genomic Data Tabular Data Image Data	The reinforcement learning models are widely used in studies where the states in the problem are deterministic and in situations where control over the environment is needed. RL models are proven to exhibit better non-linearity in gene analysis.	Adding excessive amounts of reinforcement learning may result in an overflow of states, which might reduce the effectiveness of the findings. As well, RL models are data-hungry.
Decision Tree [39]	Tabular Data Image Data	Using Decision Trees, the efforts to preprocess data can be reduced as normalization and scaling are not required, and missing values will not influence the model's outcome.	DT models consume more time to train the model, and more effort is desired.
J48 [40]	Tabular Data Image Data	J48 is a decision tree that can handle outliers effectively and robustly in non-linear problems.	J48 model is less stable, and noisy data compromises the efficiency of the data.
K Nearest Neighbor [41]	Tabular Data Image Data	The K Nearest Neighbor model does not need prior training for classifying the class data. It requires lesser computational efforts and a faster resultant outcome.	The KNN model fails to work with a larger dataset and high-dimensional data. The feature scaling phase is crucial for an optimal classification level, which requires considerable effort.
Logistic Regression [42]	Tabular Data Image Data	Logistic Regression is the very predominantly used classification technique. The model efficiently classifies the data based on the likelihood and the association among the data items. The model can sustain the overfitting and underfitting issues.	The challenging part of the Logistic Regression is linear separatable and often leads to overfitting when observations are fewer concerning the feature set size.
Naive Bayes [43]	Tabular Data Image Data	Naive Bayes algorithms perform well for multi-class classification models with minimal training.	NB assumes all the feature vectors as mutually independent components in the classification process. NB may not perform better in evaluating the problems with the interdependent feature set.
Random Forest [44]	Tabular Data Image Data	Random Forest models perform bagging for classification. RF models efficiently reduce the over-fitting issue and can handle the missing effectively. Moreover, the feature scaling task need not be performed.	RF models need tremendous training, and frequent hyperparameter tuning is required for considerable Accuracy.

Table 1. *Cont.*

Approach	Type of Data	Applicability	Limitations
Support Vector Machine [45]	Tabular Data Image Data	Support Vector Machine is efficient in handling thigh-dimensional and efficient memory handling capability.	SVM is inappropriate for working with a larger dataset with a larger feature set. The outcome of the SVM model is largely dependent on the objective function. Too many support vectors will be generated when choosing a larger kernel, which might impact the model's training process.
Genetic Algorithm [46]	Genomic Data Tabular DataImage Data	A genetic algorithm is an evolutionary algorithm that uses probabilistic transaction rules, and non-linearity in the searching process would yield better model accuracy. As well, can effectively handle the larger search space.	The genetic algorithm has susceptible to local maxima and minima and similarly to global maxima and minima. That might result in poor prediction performances.

All the mentioned models rely on tabular datasets such as PIMA and ECG signals [47] in classifying the records with possible diabetic illnesses. The current study considers that genomic data yields a better patient-centric outcome than tabular data.

2.3. Genomics for Type 2 Diabetes

Many research studies have been carried out on genetic-based illness prediction. Incorporating machine learning approaches with genetic-based illness prediction could result in an accurate outcome. This has intensified the role of Artificial Intelligence (AI) in healthcare. It has been estimated that approximately $36 billion will be invested in AI by 2025 [48]. Deep genomics through machine learning approaches has outperformed accuracy in predicting and diagnosing illnesses such as cancer with minimal inclusion of radiologists. It is desired to have sufficient biological knowledge to understand how genetics can help us predict various conditions and analyze each chromosome to identify the disease-causing gene. Pre-existing research studies have focused on genomics and gene interaction patterns of various persistent illnesses such as Alzheimer's, multiple cancers, and Parkinson's.

Many aspects need to be considered in the predictive analysis of an illness, as a gene mutation might lead to two or three diseases. The main challenge when handling genomic data for illness prediction is that the prototypical microarray image consists of fewer records. In contrast, the number of fields concerning genres could result in a few lakhs that might misinterpret the data with a significant false-positive ratio. Enhanced Gene-Set analysis can be deployed to extract and analyze genes resulting in soaring throughput on molecular assessments. Gene-Set analysis, as stated by Mooney M. A. and Wilmot B. [49] and Mathur R. et al. [50], is also referred to as pathway analysis, is meticulous in aggregating gene-sets with identical properties or sequences per the reference's gene trained or presented in the disease's knowledge base. Genome-wide association studies (GWAS) have demonstrated that many disease-causing genes are related to human diseases. GWAS has also provided polygenic characteristics of diseases. Figure 1 presents a block of GWAS in disease prediction. There are many steps during a gene-set analysis. They are shown below as Steps 1 through Step 6:

Step 1: Preliminary genome-wide analysis and data preprocessing;
Step 2: Identifying gene-set definitions whose patterns have to be recognized;
Step 3: Processing genomic data such as filtering and identifying gene patterns;
Step 4: Identify gene set analysis models, such as identifying the statistical hypothesis;
Step 5: Assessing the statistical magnitude;
Step 6: Report summarization and visualization.

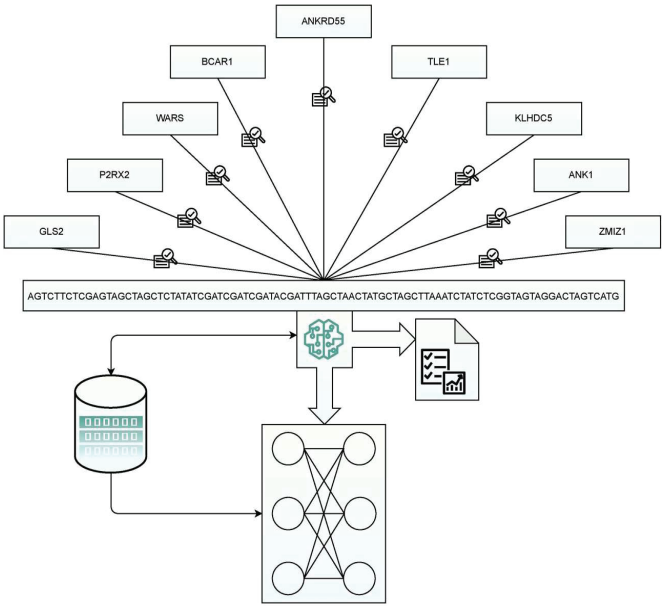


Figure 1. Gene analysis-based disease perdition framework.

Gene data include metadata about the information associated with type 2 diabetes, consisting of alleles, MegaBase, and Single nucleotide polymorphisms. An allele is a word that denotes a particular gene sequence copy associated with a specific context. A mutation might be considered one of two or more varieties of a particular gene. Most individuals have SNPs. However, some variants are more prevalent than others in particular populations. A single DNA-building unit, the nucleotide, is found at tens of thousands of different sites on the human genome. In genetics, a MegaBase is a unit of length representing a genomic region's length. MegaBase is used to determine the distance between two genes. Values of these gene features mentioned above are considered when evaluating the possibility of feature disease.

A highly dense genotyping collection is considered for coverage throughout the whole genome, such as covering common and uncommon variations in the genome. These gene sequences contain many single-nucleotide polymorphisms (SNP) that can significantly improve the capture of low-frequency variations, which is advantageous to users of other genome-wide collections. Gene sequences that hold a higher possibility of T2D in the future are listed in Table 2. Disease-corresponding gene sequences are cross-validated against individual data for forecasting the likelihood of the disease.

Table 2. Genomic information associated with Type 2 diabetes.

Gene Data	Type 2 Diabetes	Fasting Glucose	Alleles	SNP	Megabase
GLS2		✔	G/A	rs2657879	55.2
P2RX2		✔	A/G	rs10747083	131.6
WARS		✔	G/T	rs3783347	99.9
BCAR1	✔		T/G	rs7202877	73.8
ANKRD55	✔		G/A	rs459193	55.8
TLE1	✔		G/A	rs2796441	83.5
KLHDC5	✔		C/T	rs10842994	27.9
ANK1	✔		C/T	rs516946	41.6
ZMIZ1	✔		A/G	rs12571751	80.6

3. Methodology

This study is focused on predicting future illnesses such as type-2 diabetes from genomic and tabular data. Genomic data are analyzed for possible gene expression highly likely to be affected by type-2 diabetes. Tabular data from the PIMA dataset with various features are also explored through the proposed RNN model by identifying the feature vector's pivotal features. The proposed model relies on the Deep Neural Networks (DNN) framework for analyzing the genomic data, making the precise assessment of possible future illnesses with better Accuracy than the conventional pattern-matching techniques. DNN is a probabilistic measure that would summarize the possible illness outcome that would better assist in decision-making by the physicians. The working procedure and implementation details are discussed in the current section. The models are trained from the available gene base from scratch initially, and at the later stages, the model learns from the experimental outcomes.

3.1. Recurrent Neural Network Model for Type 2 Diabetes Forecasting Based on Genomic Data

Predictions of future illness can be performed through Convolutional Neural Networks (CNN), as stated by Leevy J.L. et al. [51] and Yadav S.S. and Jadhav S. M. [52] using Recurrent Neural Network (RNN) module-based architecture described by SivaSai J.G. et al. [53]. CNN model consists of many intermediate nodes connected. Each node is significant in delivering the output following the anticipated outcome. RNN is robust in handling variable-length input sequences with the help of internal auxiliary memory modules [54]. The detailed architecture along with the implementation procedure for the proposed approach, is presented in this section.

With the proposed approach, gene patterns are analyzed against pre-trained sequences of genes that cause the disease. For the effective implementation of illness prediction, the recurrent neural network component is incorporated with gene set analysis, which could minimize the false positive ratio. The Recurrent neural network model is a layered architecture approach where each layer works independently. The output of the previous phase is fed as the input for the next phase. Recurrent neural networks can transform individualistic components into contingent components by adjusting each layer's weight and bias by minimizing the number of parameters to be considered and reducing the complexity of memorizing the previous layer's output. The responsibility of each layer is presented in this section, along with the working procedure of the proposed model.

3.1.1. Data Collection and Processing

Gene-related data were acquired from the open-access comprehensive miRbase-18.0 R dataset with human gene sequences of 10,094 records labeled and annotated [55,56]. In the present experimental study, 303 samples were considered for the training and validation of the model at 70:30 proportion, respectively. Generally, gene sequences are 84 nucleotides in length, ranging from 43 nucleotides to 154 nucleotides.

The data acquired from online repositories must be processed following the model's outcome. The information is organized in tables to be further refined to predict gene sequence better, including aligning the region of interest in genomic patterns. Gene sequences could be expressed as a grid in which each location corresponds to a single-hot vector containing letters A, C, G, and T. Gene expression is indeed a matrix containing absolute values, each such element representing the pattern that is an integral part of the gene in a particular environment, such as a cell. Spatial information is often described as a three-dimensional array, with two dimensions representing the entity's actual location and a third dimension representing colors or genes. Typically, texts are defined as a one-hot matrix for each token entering a stable database. While most cells have the same genome, individual genes are expressed at highly variable amounts in variable tissues and cells in response to various treatments and settings. Such degrees of gene expression could be quantified by measuring levels of mRNA transcripts. In such context, comparison of gene

expression of patients with the illness to that in healthy cohorts (without the disease of interest) and different link genes with the diseases underlying biological systems.

3.1.2. Feature Selection

When analyzing gene data for illness prediction, features are significant in obtaining an accurate and precise outcome. Feature selection is one of the vital phases of the proposed approach. The feature selection process performed during the training step would have a noticeable contribution to the dimensionality reduction of gene data, including discarding irrelevant data and recognizing vital records in the dataset. The proposed approach's performance depends on the feature selection mechanism in the present work. It is significant in identifying the diseased gene from the extracted genomic information for the human body. Minimum Redundancy Maximum Relevance (mRMR), as stated by Zena M. Hira and Duncan F. Gillies [57] and M. B. Shirzad and M. R. Keyvanpour [58], was used for feature selection and extraction of microarray data in the current study.

The minimum Redundancy Maximum Relevance (mRMR) approach maximizes the relevancy of components concerning the genomic information and minimizes the number of corresponding classes. mRMR-based feature selection technique that favors features that have a strong correlation with class but a low correlation among themselves. In the feature extraction process, divergent statistical metrics are considered, including Mutual Information (MI), which assesses the entropy of a random variable concerning other variables in the corresponding class. The mRMR approach can also be used with both continuous and discrete variables. The amount of MI among features is used to calculate redundancy. If the value of MI is substantial, it indicates a significant degree of data redundancy among the two characteristics, i.e., redundancies. A lower redundancy measure value suggests more effective feature selection criteria. The purpose of redundancy is to locate the feature with the lowest MI value among all features. According to the premise that the lower the value of information redundancy across features, the more helpful it is to activity categorization, which may be stated by decreasing MI among features [59]. The following equation determines the gene that is not redundant for a set of features $\beta(x \in \{1, 2, \ldots f\})$

$$M_r = \frac{1}{|f|^2} \sum_{\alpha,\beta \in C} MI(\alpha, \beta) \quad (1)$$

In the above Equation (1), the discrete variable MI is the Mutual Information, variables α and β represent genes, and $|f|$ represents the number of features in class C. The maximum relevance concerning the target class is determined through the following equation:

$$R = \frac{1}{|f|} \sum_{\alpha \in C} MI(\gamma, \alpha) \quad (2)$$

In the above Equation (2), the variable γ is the class label for discrete variables. F-Statistics for assessing the mean of two classes are significantly divergent for determining the maximum relevance among corresponding genes and the class label. The minimal redundancy that approximates the correlation of the complementary gene pairs in the class is approximated as shown in Equations (3) and (4)

$$R = \frac{1}{|f|} \sum_{\alpha \in C} F_s(\alpha, \gamma) \quad (3)$$

$$M_r = \frac{1}{|f|^2} \sum_{\alpha,\beta \in C} x(\alpha, \beta) \quad (4)$$

The mutual information among the two gene sequences let them be p and q, and if some gene-sequence of p is there in the gene sequence q. The MI for the gene sequences is assessed using the generic formula for mutual information, as shown in Equation (5).

$$MI(p;q) = \sum_{p,q} f(p,q) log \frac{f(p,q)}{f(p), f(q)} \quad (5)$$

The Pymrmre package helps work with the mRMR method by employing an ensemble mechanism to further investigate the feature map and construct a more robust feature set. Figure 2 represents the feature selection mechanism for selecting the optimal features for gene-data analysis.

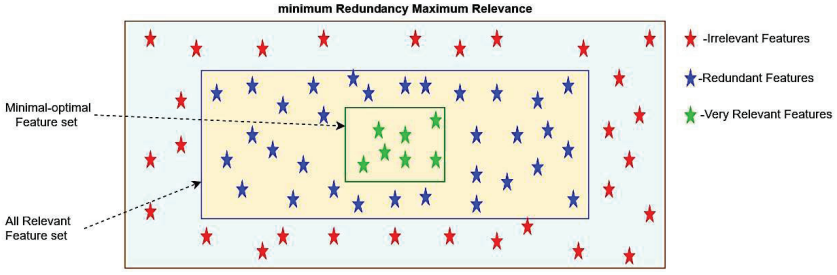


Figure 2. Diagram representing the mRMR Features selection technique.

3.1.3. Layered Architecture of RNN-Based Prediction Model

There are multiple layers in the proposed RNN model. Each plays a vital role in performing the predictive analysis of the illness, according to Carrara, F., Elias [60], and Che et al. [61]. The RNN model has kernels that work on inputs to create a feature map to detect referee patterns in the corresponding input sequence. The outermost layers would be the input and output layers. There are many other intermediate layers, including the Convolutional, max-pooling, Flattening, fully connected, and softmax layers. The outermost layer captures the gene sequences that must be validated against the training set. The inner Convolutional layers are used to handle complex patterns. Each of those Convolutional layers also decomposes gene sequences.

The pooling layer acts as the interface between two convolution layers. Its focus is on minimizing the number of parameters required for processing the data, thereby handling overfitting. The pooling layer is responsible for reducing the spatial size of the model so that the model is computationally feasible. The max-pooling layer would result in the statistical outcome of decomposing the input to the minor extent possible and performing components' filtering. Members that hold the maximum values are processed to the further stage. The rest of the components are left unprocessed. To flatten the layer associated with the conversion process of the data obtained from the previous layer, it is necessary to create a one-dimensional array of gene data that contain data to be fed to the next layer. The convolutional result is flattened to compress the outcome of convolutional layers into a single lengthy feature vector. The final classification model is termed a fully connected layer. It is linked to the output. The fully connected layer does have connections to all nodes in the layer. It is feasible to learn all nonlinear combinations of various complex patterns. The reasonability of this layer is to obtain the probability of the gene causing the abnormality. The fully connected layers comprise two significant layers. The first fully connected layer gets the input data from parameter analysis and labels the input GENE sequence for accurate prediction through weights. The fully connected output layer approximates probabilities of illness-causing genes from gene sequences [62].

In addition, the Softmax layer expands the concept into something similar to a multi-class environment. Specifically, in a multi-class classification issue such as disease predictions, Softmax gives decimal probability to each class. The sum of all such probabilities

associated with each category is equivalent to 1.0 in the long run when dealing with decimal probability. Figure 3 presents the layered approach of RNN used in the prediction model.

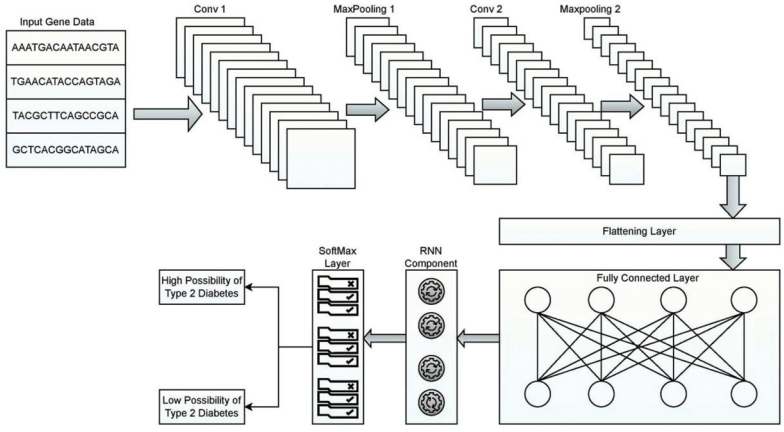


Figure 3. Layered architecture diagram of the RNN model.

Convolution is an operation that transforms a function into an output component; it is a technique that follows a certain sequence and involves intertwining two different information sources. Every single Convolutional Neural Network starts with a Convolutional Layer as its very first layer. The input is subjected to a convolutional operation in convolutional layers, and the output is then passed on to the next layer. A convolution reduces the values of all the pixels included within its receptive field to one. ReLU activation function is used with the Convolution layer. When using ReLU, all of the negative pixels are converted to 0 via an element-wise procedure. The result is a corrected feature map, which adds non-linearity to the network.

- After the Convolutional layer, the pooling layer is often applied. The pooling layer's purpose is to minimize the volume of the input matrix for subsequent layers. In the current study, the MaxPooling function is used in the current study.
- A flattening operation transforms data into a one-dimensional array to be used in a subsequent layer. This is conducted so that CNN's output may be sent to a fully connected network.
- A neural network is a collection of non-linear, mutually dependent functions. Neurons are the building blocks of every single function (or a perceptron). The neuron uses a weights matrix as a fully connected layer to apply a transformation matrix to the input vector. The result is then subjected to a non-linear transformation via a non-linear input signal s as shown in Equation (6).

$$f_c = f\left(\sum_{i=1}^{p} \omega_{ck} a_i + \omega_{c0}\right) \quad (6)$$

- One way to represent a set of numbers as probabilities are to use the Softmax mathematical function, which multiplies all the values in a set by the scale at which they appear in the vector. The likelihood of belonging to each class is calculated using the outcome of the softmax algorithm.

3.1.4. RNN Component Structure

A recurrent Neural Network, also known as a back-feeding neural network, is a more robust alternative to conventional feedforward neural networks as it does not need an internal auxiliary memory. As the outcome of a current input relies on the previous calculation, RNN is recurrent. After the outcome has been produced, copying and sending the output into the recurrent network is known as "back-feeding." The decision-making

process analyses what it has learned from the prior information and applies it to the present situation. Using the gene patterns present in the sequence, RNN may extract the correlated patterns that result in type-2 diabetes. The same could be employed in analyzing variable-length gene data for the probability of being affected by type 2 diabetes.

An RNN can evaluate any sequences, irrespective of length, iteratively through its transition function over the state vector O_i. At iteration i, state activation may be calculated as a function of the input sequence character Z_i and the prior state vector N_{i-1} transformed into the R_i in the current state cell. The tan h is the activation function associated with each cell. In RNN, the vanishing gradient issue is considered the most crucial challenge. More extensive sequences need an activation function such as tan h with a high second derivative that can maintain the gradient over iterations. Mathematical notations for each RNN module are presented in Equations (7)–(11).

$$N_i = \sigma_N(x_i) \tag{7}$$

$$\sigma_N(x_i) = \sigma_N(\alpha_N z_i + \beta_N R_{i-1} + b_N) \tag{8}$$

$$g_i = \sigma_g(x_t) \tag{9}$$

$$\sigma_g(x_t) = \sigma_g(W_g N_i + b_N) \tag{10}$$

$$O_i = \tan h\left(W_g O_{i-1} + W_{g-1} Z_i\right) \tag{11}$$

In Equations (7)–(11), the variable R_i denotes the input vector of size $(1 \times x)$, the variable Z_i. The input for the RNN cell denotes the input vector of length $(1 \times x)$. Variables α and β denote the parameter matrix associated with pivotal features. The bias is represented by b_N. Variables σ_N and σ_g denote the activation function in the RNN cell. Variable W_g and W_{g-1} denote weights associated with the cell in the current and previous state. The RNN cell structure is presented in Figure 4, where the output of the previous component is fed as the input for the upcoming component in the RNN cell.

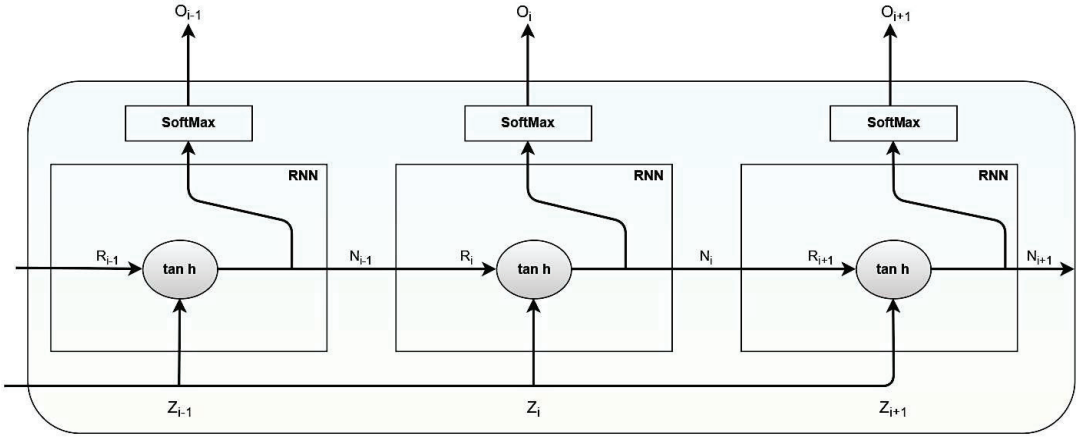


Figure 4. Image representing the RNN cell structure.

In the RNN model, the tan h denotes the Activation function, which implements a non-linearity that negates maximum activation values, creating a negative activation -1. The softmax layer deliberates probabilities that will assist in determining the possibility of future illness from the provided input genomic data.

3.1.5. GRU Component Structure

GRU's component in neural networks is used to address the degradation issue and create a feasible deeper layout for better Accuracy that can retain lengthy semantic patterns

without calibrating model parameters [63]. The GRU component consists of the update gate and the reset gate. The update gate regulates the inflow of data to the memory component. The reset gate regulates data flowing out of the memory component, GRU. The gating unit controls the data flow inside rather than having a separate memory component to perform the task. The unit consists of two activation functions: σ and tan h. The output of the current units is identified by cs_t becomes the input for next unit as cs_{t-1} over the time t. The variable α_t is assumed as the input training data and β_t is the corresponding output generated by the activation functions Γ_r and Γ_u that denotes the reset gate and the update gate, respectively. The value of Γ_u lies in between 0 and 1. When its values are close to 0, more data from the previous states are retained. The range of the variable Γ_r lies in between -1 and 1. When the value is close to -1, it implies that more previous data are ignored. The GRU can be shown mathematically through Equations (12)–(15).

$$\Gamma_u = \sigma(\omega_u[cs_{t-1}, \alpha_t] + bias_u) \tag{12}$$

$$\Gamma_r = \sigma(\omega_r[cs_{t-1}, \alpha_t] + bias_r) \tag{13}$$

$$\hat{cs}_t = \tan h(\omega_{cs}[\Gamma_r \times cs_{t-1},\ \alpha_t] + bias_{cs}) \tag{14}$$

$$cs_t = (1 - \Gamma_u) \times cs_{t-1} + \Gamma_u \times \hat{cs}_t \tag{15}$$

From Equations (12)–(15), variables ω_u, ω_r, and ω_{cs} designate weights associated with training the update gate, reset gate, and candidate activation, respectively. Similarly, variables $bias_u$, $bias_r$, and $bias_{cs}$ designate the bias associated with the update gate, reset gate, and candidate activation, respectively. Figure 5 presents the architecture of the GRU module.

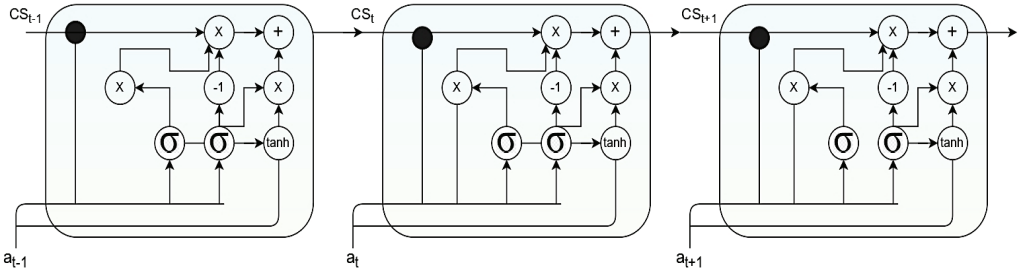


Figure 5. Image representing the GRU cell structure.

3.1.6. LSTM Component Structure

LSTM component is often used in recurrent neural network designs for pattern estimation issues in the sequential data over divergent time scales. Memory cells handle memory components in an abstract LSTM layer module, including an input and output gate, a forgetting gate, and a window connection [64,65]. Associated weights are comparable to those that change during a model's training process to regulate input and hidden states. Activation functions for the LSTM component are explained with Equations (16)–(20). States are identified through variable S_t with a hidden state vector identified by ϑ_t concerning the time t over the input i^t.

$$Input\ Gate\ (\rho_t) = \sigma\left(i^t\omega_{i\rho} + \gamma_{t-1}\omega_{\gamma\rho 0} + ps_{t-1}\omega_{ps\rho} + bias_\rho\right) \tag{16}$$

$$Output\ Gate\ (o_t) = \sigma\left(i^t\omega_{io} + \gamma_{t-1}\omega_{\gamma o} + ps_t\omega_{pso} + bias_o\right) \tag{17}$$

$$Forget\ Gate\ (\chi_t) = \sigma\left(i^t\omega_{i\chi} + \gamma_{t-1}\omega_{\gamma\chi} + ps_t\omega_{ps\chi} + bias_\chi\right) \tag{18}$$

$$Cell\ State\ Gate\ (ps_t) = \chi_t \cdot ps_{t-1} + \rho_t \cdot \tan\gamma\left(i^t\omega_{ips} + \gamma_{t-1}\omega_{\gamma ps} + bias_{ps}\right) \tag{19}$$

$$LSTM\ Output(\gamma_t) = o_t \cdot \tan\gamma\ (ps_{t-1}) \tag{20}$$

From Equations (16)–(20), variables $\omega_{i\rho}$, ω_{io}, $\omega_{i\chi}$, and ω_{ips} designate weights associated with the input, output, forget, and cell state gates, respectively. In addition, $\omega_{\gamma\rho}$, $\omega_{\gamma o}$, and $\omega_{\gamma\chi}$ designate weights associated with the hidden layer. Similarly, variables $bias_{\rho}$, $bias_{o}$, $bias_{\chi}$, and $bias_{ps}$ designate the bias component associated with the input gate, output gate, forget gate, and cell state gate, respectively. The architecture of the LSTM component is presented in Figure 6.

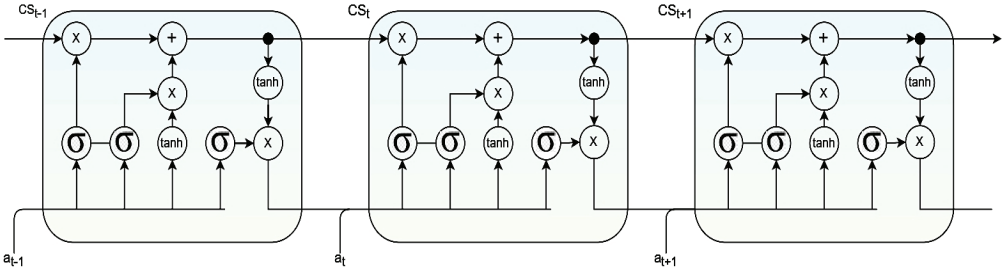


Figure 6. Image representing the LSTM cell structure.

3.1.7. Working Procedure of the Proposed Approach

The working procedure presents the sequence of tasks performed for future illness prediction, including tasks ranging from initial data acquisition to final assumptions of the future illness.

Step 1: Acquire gene data from the annotated miRbase data set;
Step 2: Data are preprocessed to remove the outlier data and fill out acquired data gaps;
Step 3: Data is converted into 1D data, followed by aligning of genomic patterns;
Step 4: Data is categorized into a training set (80% of the data) and a testing set (20% of the data);
Step 5: Patterns are labeled based on sequence patterns of various illnesses. Moreover, weights are assigned in the later phases according to the correlation between the input sequence and the trained gene pattern;
Step 6: When a new GENE sequence is fed as input for testing the algorithm, features are extracted through the mRMR approach that is pivotal in the prediction process;
Step 7: The cumulative weight is evaluated from assigned weights based on the correlation of gene sequences between the input and the trained set;
Step 8: Based on the approximated weight of the gene sequence, the probability of a future illness is assessed;
Step 9: Final assumptions are made based on probabilistic approximations.

3.2. RNN Model for Illness Prediction from Tabular Data (PIMA Dataset)

The possibility of being affected by a chronic disease such as type-2 diabetes is analyzed from the tabular data with various features such as Stabilized Glucose, age, High-Density Lipoproteins (HDL) Ratio, Total Cholesterol, First Systolic Blood Pressure, Second Diastolic Blood Pressure, body mass, the height of individual, gender, and many other things. Significant features are selected, weights are adjusted in favor of pivotal features, and prediction is performed based on the feature vector. The significance of these features in the evaluation process has been discussed in earlier studies on a similar feature vector for type 2 diabetes [66–68]. Ranks associated with each of these features are presented in Table 3, shown below.

Table 3. Feature set associated with Type 2 diabetes.

Feature	Data_Type	Min_Value	Max_Value	Information Gain	Mean Rank
Glucose (mg/dL)	Integer	0	199	0.2497	3
Pregnancies	Integer	0	17	~	~
Age	Integer	21	81	0.0761	3.17
Heart Rate	Integer				7.67
Waist	Integer	~	~	0.0356	9.5
Pulse Pressure	Integer			~	12.33
Insulin (mm U/mL)	Integer	0	846	~	13.33
Hypertension (Blood Pressure) (mm Hg)	Integer	0	122	0.0304 (bp1), 0 (bp2)	15
BMI (weight) (kg/m^2)	Real	0	67.1	~	~
Diabetes Pedigree Function	Real	0.08	2.42	~	~
Skin thickness (mm)	Real	0	99	~	~

3.2.1. Feature Weight Initialization

Features are essentially important for analyzing the possibility of diabetes disease. Weights associated with features and corresponding layers are updated over iterations [69]. When the feature is significant, it will cascade forward via hidden nodes, showing greater influence over output nodes. Thus, weighting such a significant feature is important. The feature that contributes more to the prediction process will be given considerable weightage for further processing. After training, the feature weight is obtained from the trained neural network, as shown in Equation (21).

$$I_w = \sum_{i=0}^{p-1} \sum_{j=0}^{q-1} \left| \omega_{i,j} \times \omega_{j,k} \right| \tag{21}$$

From Equation (21), the variable I_w is the initial weight assigned to the feature vector, $\omega_{i,j}$ denotes the network weight between the input node i through hidden node j. Similarly, the variable $\omega_{j,k}$ denotes weights of the hidden node j through output node k. The summation covers all potential forwarding routes between input node i and output nodes. For a few less significant features in the evaluation process, their weights are adjusted so that the sum of approximated weights of less significant features is equivalent to the total number of features. Associated weights for less significant features are given through Equation (22).

$$\omega_{lsf} = \frac{1}{n} \sum_{i=0}^{n-1} \omega_{lsf} \tag{22}$$

From Equation (22), the variable ω_{lsf} designates less significant weights, and the variable n designates the number of features in the considered problem. In the current context, the value of n is 8 from the Pima dataset.

3.2.2. Weight Optimization

Weights associated with features must be optimized regularly for better performance of the model. These weights are optimized concerning the loss function and model parameters associated with each parameter in the training dataset [70,71]. In the current study, the input-target pair (i, j) and the $\left\{ \left(i_p, j_p \right), 0 \le p \le n \right\}$ denote the training set. The validation set is associated with the model for fine-tuning the model's performance using the set $\left\{ i'_p, j'_p \right), 0 \le p \le m \}$, where the size of m is much smaller than the size of records in n. The RNN model is denoted by $\Re(p, \theta)$. The associated loss function will be $L\left(j', j\right)$ which is desired to be minimal, where $j' = \Re(i, \theta)$. The expected loss associated with the training set is determined through the variable T_l as shown in Equation (23).

$$T_l = \frac{1}{n} \sum_{p=0}^{n-1} L\left(j', j\right) \tag{23}$$

$$T_l = \frac{1}{n}\sum_{p=0}^{n-1} f_p(\theta) \text{ where } L(j', j) = f_p(\theta) \quad (24)$$

In Equation (24), the function $f_p(\theta)$ is the loss function concerning data i_p. Weights associated with parameters are optimized to minimize the weighted loss through Equation (25).

$$O(\theta)_w = \theta' \sum_{p=0}^{n-1} \omega_p f_p(\theta) \quad (25)$$

The value of the variable ω_p is not known at the initial iteration. The value $\{\omega_p\}_{p=0}^{n-1}$ is tuned by training hyperparameters. The validation dataset could result in fine-tuning the value of ω to reduce the weighted loss of the prediction model, as shown in Equation (26).

$$\omega' = \min_{\omega, \omega'} \frac{1}{m} \sum_{p=0}^{m-1} f'_p(\theta \times \omega) \quad (26)$$

To reduce negative training loss that could result in an unstable model, The value associated with the weight $\omega \geq 0$ for all parameters p.

3.3. Dataset Description

The Pima Indian Dataset is used in the current study to predict Type-2 diabetes. It is part of the UCI machine learning repository maintained by the National Institute of Diabetes, Digestive, and Kidney Diseases. The dataset consists of eight columns representing parameters of Pregnancy, Glucose, Blood Pressure, Skin Thickness, Insulin, Body mass index (BMI), Diabetes Pedigree, and age. The PID dataset consists of a single output class with a binary value indicating whether or not an individual has diabetes. The dataset consists of 768 cases (500 non-diabetics and 268 diabetics) [72,73]. The Pima dataset is considered in the current study as it is widely used for comparing the performances of techniques. The dataset is partitioned as training and testing in a ratio of 70:30, with an initial learning rate of 0.0002, and it is observed that the model has

3.4. Implementation Environments

The computer is equipped with an Intel(R) Core i7(11th Gen) 4.70 GHz processor and 16 GB of main memory running over a 64-bit Windows 10 environment. The proposed RNN model for gene analysis is implemented over Kaggle, an online platform for executing such frameworks [74]. Python version 3.6.6, also widely known as the anaconda, is used in the implementation. Tensor Flow version 2.4.1, along with various libraries such as NumPy, pandas, matplotlib, seaborn, and sklearn, are used in the implementation process of the proposed model.

4. Results and Discussion

The proposed model has been evaluated on genomic data and the tabular data by using the same feature engineering mechanism and the layered approach for predicting the type-2 diabetes. The proposed RNN-based type-2 diabetes is evaluated against genomic and tabular data from the PIMA Indian dataset independently and the evaluations are presented independently in the current section. The model was evaluated against two datasets concerning various evaluation metrics such as sensitivity, specificity, Accuracy, and F1 score. The classification efficiency of the proposed model was assessed using true positive (TuP, the number of times that the model accurately predicted the gene with a high possibility of diabetes correctly), true negative (TuN, identifying the gene with less possibility of diabetes precisely), false positive (FsP, misinterpreting the gene with the high possibility of diabetes as low possibility of diabetes), and false negative (FsN, misinterpreting the low diabetes gene as a high possibility of illness). The sensitivity metric determines the ratio of how many were accurately recognized as positive samples out of how many were truly positive samples in the complete dataset. The specificity measure determines the ratio of how many were recognized as negative samples out

of how many among the samples are truly negative from the complete dataset. The Accuracy measures the correctly predicted True positives and Negative samples against the overall sample in the complete dataset. The harmonic mean of sensitivity and specificity measures are determined as the F1 score. MCC is the best single-value classification score for summarizing the confusion matrix. The formulas for the aforementioned metrics are presented through Equations (27)–(32) [75].

$$senstivity(recall) = \frac{TuP}{(TuP + FsN)} \tag{27}$$

$$Specificity = \frac{TuN}{(TuN + FsP)} \tag{28}$$

$$Accuracy = \frac{TuP + TuN}{(TuP + FsP + TuN + FsN)} \tag{29}$$

$$Precision = \frac{Tup}{(TuP + FsP)} \tag{30}$$

$$F1\text{-}score = 2 \times \frac{(precision \times recall)}{(precision + recall)} \tag{31}$$

$$mcc = \frac{(TuP \times TuN) - (FsP \times FsN)}{\sqrt{(TuP + FsP)(TuP + FsN)(TuN + FsP)(TuN + FsN)}} \tag{32}$$

It is a far more appropriate statistical rate that yields a good score only if the prediction performed well among all the assumptions in the confusion matrix. The current section presents results about the experimental outcome of both genomic and tabular data with adequate analysis concerning existing models.

4.1. Experimental Outcome of Genomic Data

The performance of the proposed RNN model for predicting type 2 diabetes was analyzed using performance evaluation metrics such as sensitivity, specificity, F1 score, Mathews correlation Coefficient, and accuracy measures [76]. The above-discussed metrics are assessed through true positive, true negative, false positive, and false negative values approximating experimental outcomes. The dataset is split into a training set and a validation set at a ratio of 70:30. In the following graph, as shown in Figure 7, it is clear that data values are skewed toward data instances, indicating that no diabetes exists. The percentage of available data records of non-diabetic patients (or those who do not have diabetes) is almost double that of diabetic patients.

Correlation coefficients among data points as input gene data are analyzed using linear bivariate Pearson correlation coefficient (PCC). The correlation coefficient between two samples of gene expression is expressed as PCC. Correlation coefficient with a common confidence interval and covariance, the relationship among them is the ratio of the covariance of two variables and the product of their standard deviations. This gives a numeric representation of the covariance with an outcome between -1 and 1. Only a linear correlation between variables can be considered, even using the metric. Also, the metric does not represent several relationships or correlations. Figure 8 shows a two-dimensional heat map of data records.

Training and validation performances of the proposed model were evaluated using hyperparameters such as train and testing scores. The training score determined how perfectly the algorithm could generalize across its training samples. The testing score determined how well the model could accurately correlate the known gene sequence among individual records. An exceptionally high training score combined with a low-test result indicates overfitting. When the training score is quite low, and the test score is low, it indicates an underfitting. Performances of the proposed model concerning hyperparameters are presented in Figure 9.

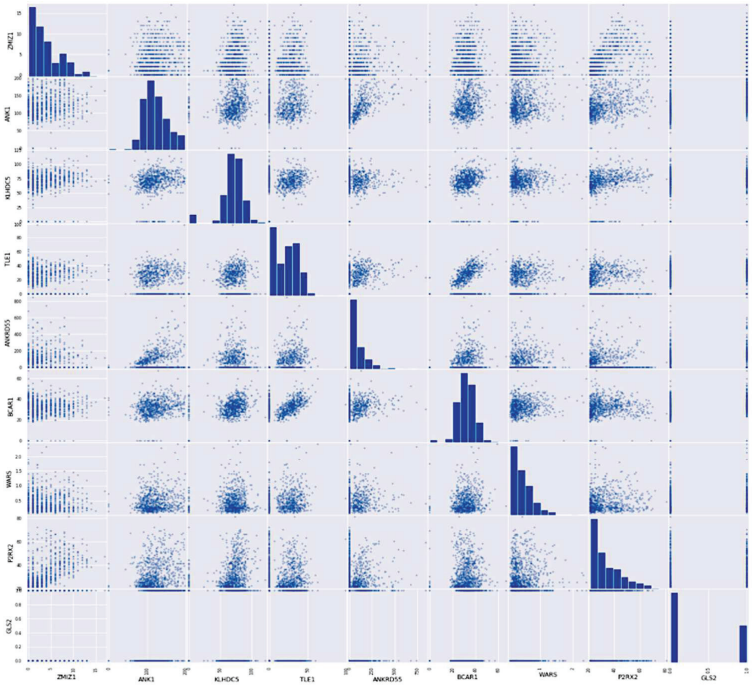


Figure 7. A scatter plot shows relationships among data points in input records.

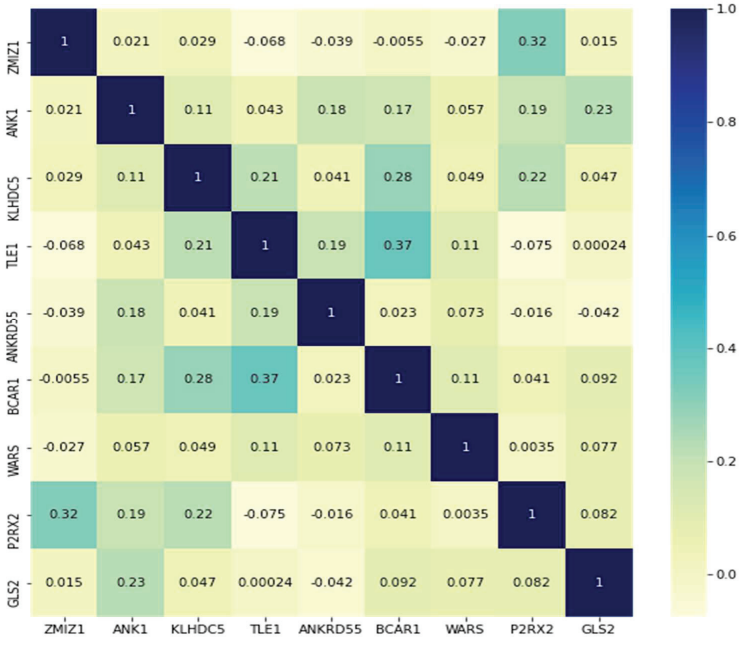


Figure 8. A heat map was generated from the gene dataset.

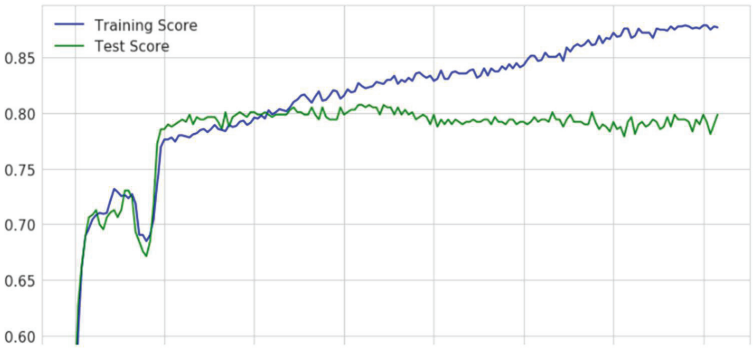


Figure 9. A graph showing training and testing scores of the proposed model.

From graphs on the training and test scores, it can be depicted that the performance is reasonably fair in making the classifications precisely as there is no considerable deviation among either of the scores. In the current context, the gene expressions are classified as sequences with a high possibility of affecting type 2 diabetes and sequences with a low possibility of type 2 diabetes. The Decision Boundary is shown in a Scatter Plot, with every data point visualized on the data scatter plot and characteristics represented by x- and y-axes. The Decision Boundary forms a boundary for dividing data points into regions and their classes. Categories of gene sequences with high and low possibilities of developing diabetes are shown in Figure 10.

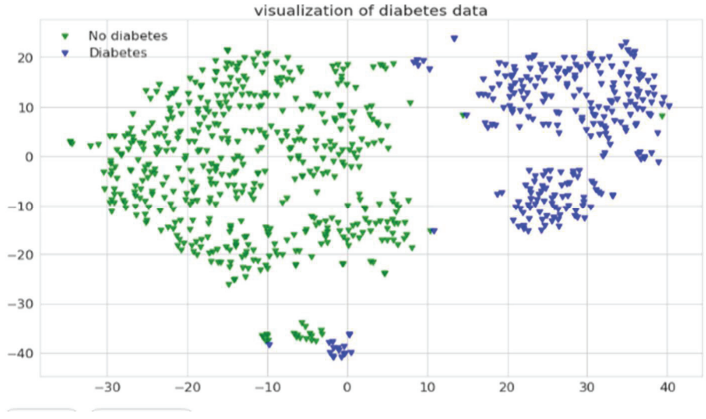


Figure 10. A graph showing the decision boundaries of two classes of records.

The confusion matrix would assist in analyzing the performance of the proposed model in analyzing future illness. The evaluated samples, i.e., TuP, TuN, FsP and FnP are shown in the confusion matrix in Figure 11, and the corresponding performance evaluation metrics are shown in Table 4.

Table 4. Performance evaluation metric and estimated values.

Metric	Estimated Value
Sensitivity	83.66
Specificity	49.38
Precision	75.73
Accuracy	71.79
Mathew's correlation Coefficient	35.09

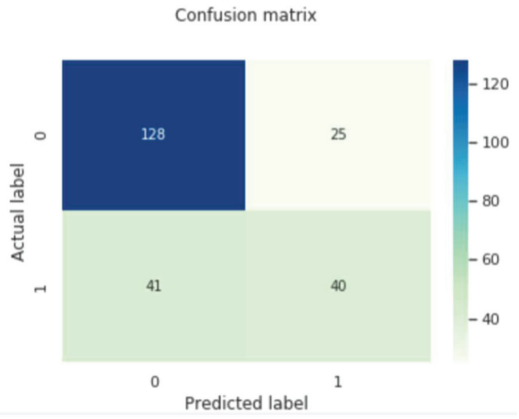


Figure 11. Image representing the confusion matrix for the proposed RNN model for future illness prediction.

As shown in Table 4, estimated values clearly demonstrated that the model made predictions reasonably with few records. However, the model's performance could be further improvised when more data records could be used. The Receiver Operating Characteristic (RoC) Curve of the proposed model is presented in Figure 12, and it is depicted that the model has outperformed with reasonable accuracy in precisely classifying the genomic data. The RoC curve estimates how well the proposed approach can differentiate the two-class records that include gene expression with a higher or lower possibility of being affected with type-2 diabetes. An accurate model can tell the difference between the two. An improper model will find it difficult to tell the difference between the two sets of records.

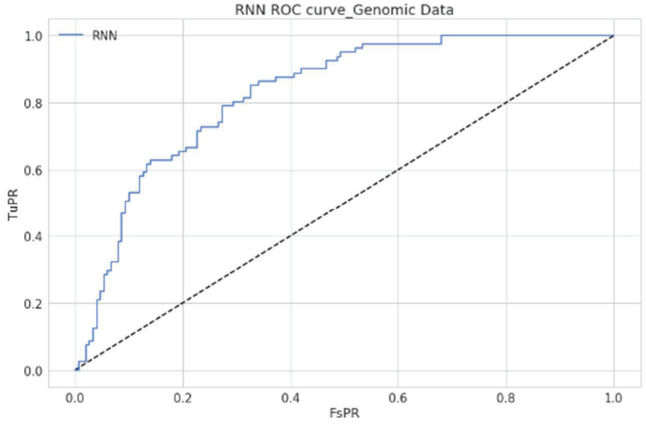


Figure 12. Graph presenting the ROC curve of the proposed model.

4.2. Experimental Outcome with Tabular Data (PIMA Dataset)

The Pima Indian dataset consists of eight features that help predict the possibility of affecting type-2 diabetes. The model's performance was evaluated using various evaluation metrics. The heat map represents the association of multiple parameters in determining a future illness. Figure 13 illustrates the heat map of features in the PIMA Indian dataset.

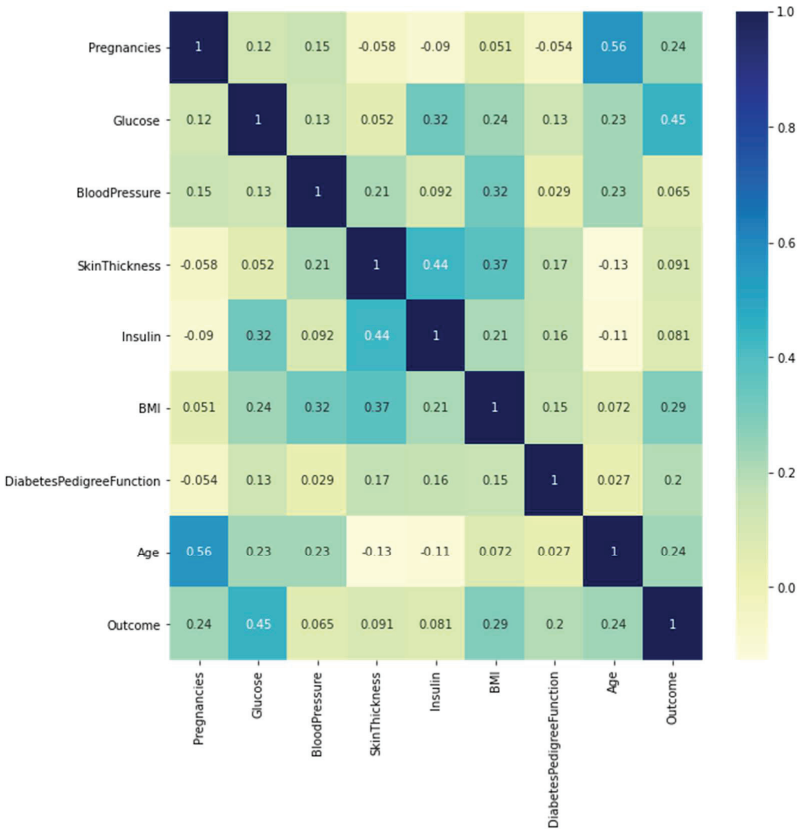


Figure 13. Heat map generated with the PIMA dataset.

The experimentation was performed with the PIMA data by optimizing initial weights assigned to parameters. The proposed model exhibited better accuracy in optimizing weights. Figure 14 presents the resultant confusion matrix obtained over data with and without weight optimization for all three recurrent neural network components. Among these, the LSTM-based architecture has outperformed in terms of classification accuracy. Correctly identifying the individual record as a diabetic patient was assumed as a True Positive (TuPR). The correctly predicting the non-diabetic patient was assumed as a True Negative (TuNR). When the model misinterpreted normal cases as diabetic cases, False Positive (FsPR) was considered. When diabetic cases were recognized as normal cases, False Negative (FsNR) was considered.

The percentage of true positives that are accurately recognized is what sensitivity analyzes. Specificity, often known as the real negative rate, is a measurement that determines the percentage of actual negative instances that are accurately classified as such. The ratio of the number of instances properly categorized to the total number of instances is called Accuracy. The F1 score is a statistic calculated by taking the harmonic mean of a classifier's accuracy and recall values and combining them into a single value. A low number of false positives and false negatives gives you an excellent F1 score. The Matthews correlation coefficient, or MCC, is a correlation coefficient that compares predicted values to actual values, which is mostly used in binary class classification problems. The weight optimization process could help evaluate the dataset more precisely as features with more significance would be considered in the evaluation process. Weights are optimized over the iteration. Resultantly, more significant features are involved in the evaluation process.

The weight optimization could yield considerable Accuracy over a conventional model. The experimental outcome presented in Table 5 shows the outcome of the proposed model concerning optimized weights.

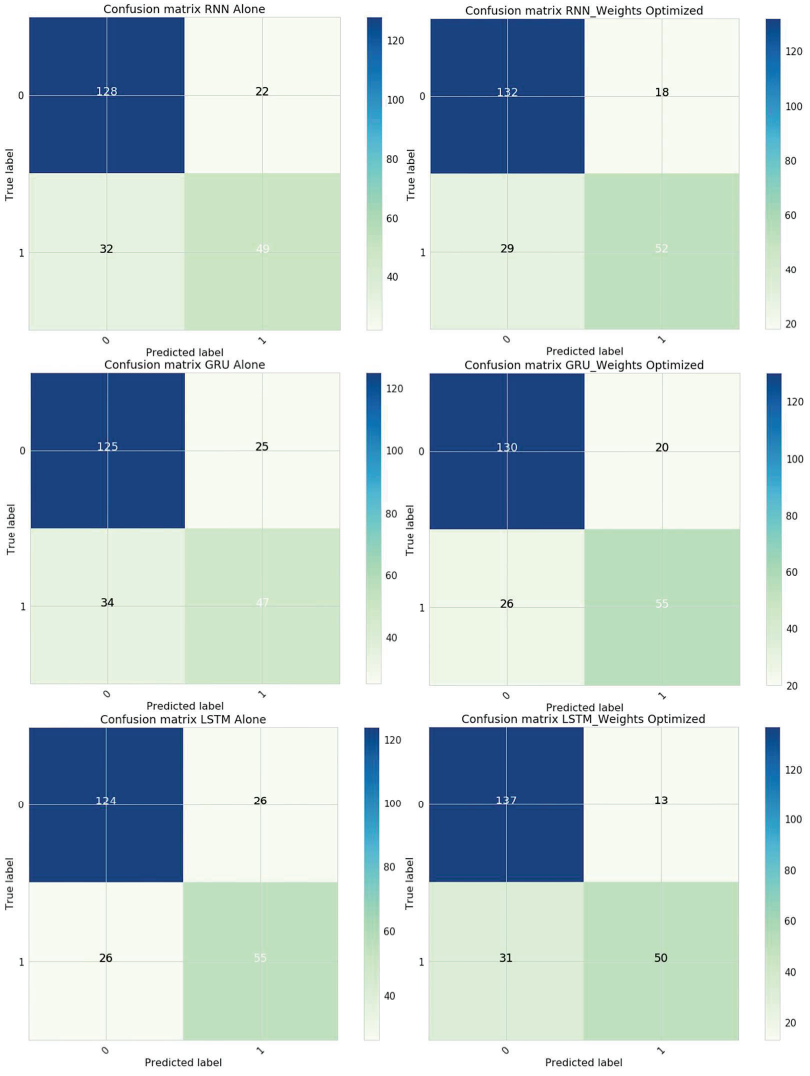


Figure 14. Confusion matrix of the proposed model.

Table 5. Performance of the proposed model with weight optimization.

	Sensitivity	Specificity	Accuracy	F1-Score	MCC
RNN Model	0.800	0.690	0.753	0.825	0.473
RNN + GRU	0.786	0.652	0.744	0.809	0.426
RNN + LSTM	0.826	0.679	0.774	0.823	0.505
RNN Model (WO)	0.819	0.742	0.796	0.848	0.541
RNN + GRU(WO)	0.833	0.733	0.800	0.849	0.558
RNN + LSTM(WO)	0.815	0.793	0.810	0.856	0.568

The classification efficiency assessment of the proposed model was compared with various existing studies concerning evaluation parameters such as sensitivity, specificity, Accuracy, and F1 score. Table 6 presents experimental values obtained by the proposed model over other existing models such as Naive Bayes, J48, Logistic Regression, K Nearest Neighbor, Random Forest, Decision Tree, REPTree, Sequential Minimal Optimization (SMO) and BayesNet. Experimental outcomes of the current model are evaluated against the outcomes of other existing models using similar datasets [77,78].

Table 6. Performance analysis of the proposed model with existing studies.

	Sensitivity	Specificity	Accuracy	F1-Score	MCC
Decision Tree	0.781	0.561	0.697	0.762	0.349
J48	0.688	0.695	0.691	0.754	0.383
K Nearest Neighbour	0.748	0.603	0.708	0.787	0.331
Logistic Regression	0.775	0.666	0.744	0.813	0.416
Naive Bayes	0.820	0.687	0.689	0.830	0.502
Random Forest	0.789	0.661	0.750	0.813	0.436
Support Vector Machine	0.775	0.666	0.744	0.813	0.416
REPTree	0.530		0.744	0.590	
SMO	0.280		0.724	0.410	
BayesNet	0.570		0.738	0.600	
RNN model	0.837	0.774	0.818	0.864	0.591

A resampling technique for evaluating the machine learning approaches is known as cross-validation, where a small data sample is considered for evaluation. The technique includes a single parameter, k which specifies how many groups are provided with sample data. The k-fold cross-validation describes the number of groups associated with the evaluation. When $k = 2$ means the model reference to 2-fold cross-validation. The formula for the cross-validation over kf folds concerning to the mean square error (MSE) is shown in Equation (33). In the current study, the accuracies of the RNN model with different auxiliary memory components are evaluated against divergent K-Values, as presented in Table 7.

$$cross\_validation_{k_f} = \frac{1}{k_f} \sum_{x=1}^{k_f} mse_x \quad (33)$$

Table 7. The Accuracy of the RNN model with auxiliary memory components against divergent K- Values.

Value of K	RNN Model	RNN + GRU	RNN + LSTM	RNN Model (WO)	RNN + GRU (WO)	RNN + LSTM (WO)
2	0.716	0.704	0.723	0.752	0.771	0.789
5	0.745	0.739	0.770	0.791	0.799	0.812
10	0.774	0.762	0.798	0.810	0.821	0.824

The ROC curve of the proposed model states the trade-off between the True Positive assumption and the False Positive assumption of the proposed model concerning predictions made with the Pima diabetic dataset. Figure 15 presents the ROC of the proposed model with optimized weights using tabular data. The proposed model has shown the classification's desired performance concerns.

The present model was trained with limited genomic data or PIMA diabetic dataset. Either of these datasets consisted of approximately 700 records. Of them, 30% of the overall data were meant for testing, resulting in training the model with inadequate records that could impact its performance. RNN-based models in various applications have exhibited noticeable accuracies. However, the neural network model's Accuracy can vary depending on the ratio of the training sample to the testing sample.

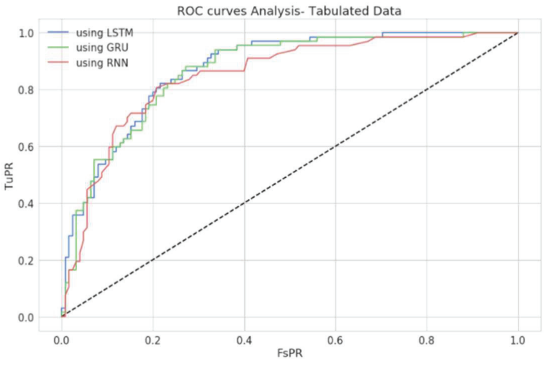


Figure 15. ROC curve of the proposed model using PIMA data.

4.3. Practical Implications

The proposed technique for forecasting type-2 diabetes can be implemented over a mobile framework with a front-end module. Patients and practitioners can perform the initial assessment of the illness. Users can provide details such as glucose levels, pregnancies, insulin, hypertension, BMI, Diabetes Pedigree, skin thickness, and heart rate. Based on the provided input and the trained data, the model can analyze the input with the trained data for predicting the illness. The model can be implemented in the iOS platform to the back-end Kaggle using the back-end service such as the MBaaS component. A secured socket layer (SSL) and two-factor authentication can ensure the security of the model [79,80].

Images of the user interface of the application model are presented in Figure 16. The leftmost image represents the registration page of the application. The middle image shows the user information page, followed by the resultant prediction screen of the model. The model makes the task of predicting a future illness more convenient. The model can be improved by incorporating the genomic module for accepting gene data and evaluating the illness based on gene information.

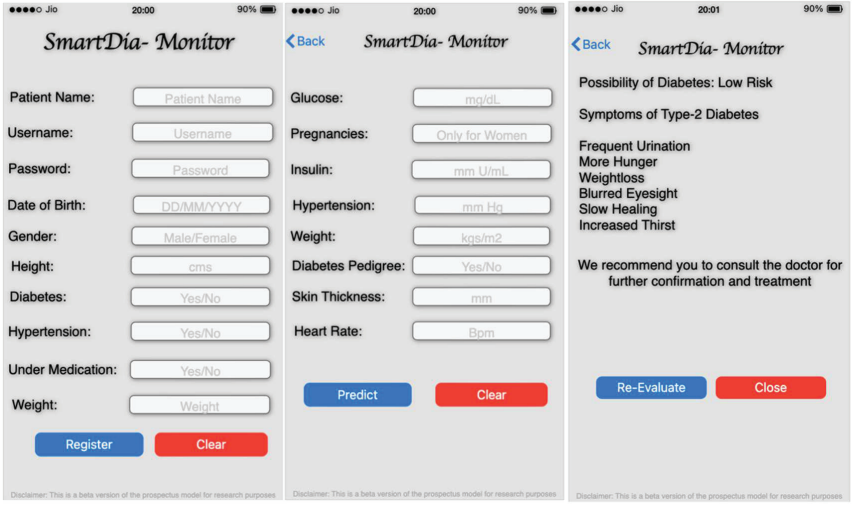


Figure 16. Images presenting the mobile interface of the future implication model.

The future implication model was inspired by a non-invasive way of assessing future illness more precisely. The assessment could help physicians and individuals better

evaluation of health conditions. The prediction could assist individuals in adopting better life standards and living habits to avoid or prolong the chance of being affected by the illness [81]. The computationally efficient approaches like the MobileNet V2 and MobileNet V3 architectures would assist better in deploying the models in lightweight computational devices. The RNN models need tremendous computation efforts despite providing highly accurate performances.

5. Conclusions

The genomic-based future illness prediction is a path-breaking approach for precisely assessing future illness. Genomic-based data can be conveniently analyzed through supervised-based approaches such as Neural Network models. The sequence of GENE can be downsampled and analyzed based on the weight concerning the diseased sequence. The approach decomposes a large GENE sequence into smaller GENE strings, raising the chances of accurate matching with diseased sequences and resulting in a precise prediction. Although there might be a considerable burden on the machine to decompose the large genomic patterns, decomposing them to a certain predetermined extent has better Accuracy than conventional approaches. When the proposed model was evaluated over the PIMA diabetic dataset, it exhibited a reasonable performance in predicting type-2 diabetes. The PIMA dataset consists of 768 records, of which only 537 are used for training. The Accuracy would be much better when more records are for training purposes. Statistical analysis for disease progression [82] has exhibited better performance than the existing models when adequate data are available. Users may access the suggested model's prediction result via the Android application. As a result, it is desired to give an effective method for determining the possibility of being affected by diabetes at an early stage. However, it is exceedingly challenging to work with larger sequence gene data, the quantum and federated learning techniques would effectively handle such a larger sequence data. On the other side, when dealing with tabular data, the ensemble classification models would yield almost identical performance with minimal computation to the suggested RNN models.

Although the proposed approach showed promising results, it was challenging when decomposing to a more significant extent. In such situations, incorporating Long Short-Term Memory (LSTM) can make the approach more robust with considerably lesser computational latency. For handling an unusual illness, self-Learning Based algorithms and the use of cognitive technology would be appropriate to minimize the steps needed for training the algorithm [83]. The proposed approach based on Genomics with Self-Learning algorithms might result in better results than supervisory approaches alone. In future work, comparison with other smart diagnosis techniques and assessment of other clinical datasets need to be performed. Once the model validation is performed with more datasets, other risk factors affecting diabetes can be revealed. The future dimensions of the research include the deep learning-driven pattern recognition models for analyzing the gene sequences for identifying the possible future illness and developing mobile applications that can generalize the information from the genomic data. However, there is great demand for explainable Artificial Intelligence models that are interpretable in decision-making.

Author Contributions: The authors contributions are as follows, Conceptualization of the study, P.N.S., M.F.I. and J.S.; Formal analysis of the study, T.B.K., J.S. and S.P.P.; Formal Investigation, S.P.P., C.N.S. and P.N.S.; Methodology, M.F.I., S.P.P. and T.B.K.; Project administration, C.N.S., P.N.S. and M.F.I.; Resources acquisition, P.N.S., C.N.S. and M.F.I.; Software, J.S., C.N.S., T.B.K. and S.P.P.; Writing—original draft, P.N.S., J.S. and M.F.I.; and Writing—review and editing, M.F.I.; All authors have checked and approved the final version of the work. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: Parvathaneni Naga Srinivasu would like to thank the management of Prasad V Potluri Siddhartha Institute of technology for their support in carrying out the current study. Jana Shafi would like to thank the Deanship of Scientific Research, Prince Sattam bin Abdul Aziz University, for supporting this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zou, Q.; Qu, K.; Luo, Y.; Yin, D.; Ju, Y.; Tang, H. Predicting Diabetes Mellitus with Machine Learning Techniques. *Front. Genet.* **2018**, *9*, 515. [CrossRef] [PubMed]
2. Hemu, A.A.; Mim, R.B.; Ali, M.; Nayer, M.; Ahmed, K.; Bui, F.M. Identification of Significant Risk Factors and Impact for ASD Prediction among Children Using Machine Learning Approach. In Proceedings of the 2022 Second International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Bhilai, India, 21–22 April 2022; pp. 1–6. [CrossRef]
3. Ravaut, M.; Sadeghi, H.; Leung, K.K.; Volkovs, M.; Rosella, L.C. Diabetes mellitus forecasting using population health data in Ontario, Canada. *arXiv* **2019**, arXiv:1904.04137.
4. Deberneh, H.; Kim, I. Prediction of Type 2 Diabetes Based on Machine Learning Algorithm. *Int. J. Environ. Res. Public Health* **2021**, *18*, 3317. [CrossRef] [PubMed]
5. Arshad, A.; Khan, Y.D. DNA Computing: A Survey. In Proceedings of the 2019 International Conference on Innovative Computing (ICIC), Lahore, Pakistan, 1–2 November 2019; pp. 1–5. [CrossRef]
6. Cho, Y.S.; Chen, C.-H.; Hu, C.; Long, J.; Ong, R.T.H.; Sim, X.; Takeuchi, F.; Wu, Y.; Go, M.J.; et al.; DIAGRAM Consortium. Meta-analysis of genome-wide association studies identifies eight new loci for type 2 diabetes in east Asians. *Nat. Genet.* **2011**, *44*, 67–72. [CrossRef] [PubMed]
7. The Coronary Artery Disease (C4D) Genetics Consortium. A genome-wide association study in Europeans and South Asians identifies five new loci for coronary artery disease. *Nat. Genet.* **2011**, *43*, 339–344. [CrossRef]
8. Duncan, L.; Shen, H.; Gelaye, B.; Meijsen, J.; Ressler, K.; Feldman, M.; Peterson, R.; Domingue, B. Analysis of polygenic risk score usage and performance in diverse human populations. *Nat. Commun.* **2019**, *10*, 3328. [CrossRef]
9. Jordan, D.M.; Do, R. Using Full Genomic Information to Predict Disease: Breaking Down the Barriers Be-tween Complex and Mendelian Diseases. *Annu. Rev. Genom. Hum. Genet.* **2018**, *19*, 289–301. [CrossRef]
10. Rahaman, A.; Ali, M.; Ahmed, K.; Bui, F.M.; Mahmud, S.M.H. Performance Analysis between YOLOv5s and YOLOv5m Model to Detect and Count Blood Cells: Deep Learning Approach. In Proceedings of the 2nd International Conference on Computing Advancements (ICCA'22). Association for Computing Machinery, Dhaka, Bangladesh, 10–12 March 2022; pp. 316–322. [CrossRef]
11. Ontor, Z.H.; Ali, M.; Ahmed, K.; Bui, F.M.; Al-Zahrani, F.A.; Mahmud, S.M.H.; Azam, S. Early-Stage Cervical Cancerous Cell Detection from Cervix Images Using YOLOv5. *Comput. Mater. Contin.* **2023**, *74*, 3727–3741. [CrossRef]
12. So, H.-C.; Sham, P.C. Exploring the predictive power of polygenic scores derived from genome-wide association studies: A study of 10 complex traits. *Bioinformatics* **2016**, *33*, 886–892. [CrossRef]
13. Sarra, R.R.; Dinar, A.M.; Mohammed, M.A.; Ghani, M.K.A.; Albahar, M.A. A Robust Framework for Data Generative and Heart Disease Prediction Based on Efficient Deep Learning Models. *Diagnostics* **2022**, *12*, 2899. [CrossRef]
14. Ali, M.; Ahmed, K.; Bui, F.M.; Paul, B.K.; Ibrahim, S.M.; Quinn, J.M.; Moni, M.A. Machine learning-based statistical analysis for early stage detection of cervical cancer. *Comput. Biol. Med.* **2021**, *139*, 104985. [CrossRef]
15. Ali, M.M.; Paul, B.K.; Ahmed, K.; Bui, F.M.; Quinn, J.M.W.; Moni, M.A. Heart disease prediction using supervised machine learning algorithms: Performance analysis and comparison. *Comput. Biol. Med.* **2021**, *136*, 104672. [CrossRef]
16. Bell, C.G.; Teschendorff, A.E.; Rakyan, V.K.; Maxwell, A.P.; Beck, S.; Savage, D.A. Genome-wide DNA methylation analysis for diabetic nephropathy in type 1 diabetes mellitus. *BMC Med. Genom.* **2010**, *3*, 33. [CrossRef]
17. Konishi, T.; Matsukuma, S.; Fuji, H.; Nakamura, D.; Satou, N.; Okano, K. Principal Component Analysis applied directly to Sequence Matrix. *Sci. Rep.* **2019**, *9*, 19297. [CrossRef]
18. Mallik, S.; Mukhopadhyay, A.; Maulik, U.; Bandyopadhyay, S. Integrated analysis of gene expression and genome-wide DNA methylation for tumor prediction: An association rule mining-based approach. In Proceedings of the 2013 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), Singapore, 16–19 April 2013; pp. 120–127. [CrossRef]
19. Mallik, S.; Bhadra, T.; Mukherji, A. DTFP-Growth: Dynamic Threshold-Based FP-Growth Rule Mining Algorithm Through Integrating Gene Expression, Methylation, and Protein–Protein Interaction Profiles. *IEEE Trans. NanoBiosci.* **2018**, *17*, 117–125. [CrossRef]
20. Huang, S.; Cai, N.; Pacheco, P.P.; Narandes, S.; Wang, Y.; Xu, W. Applications of Support Vector Machine (SVM) Learning in Cancer Genomics. *Cancer Genom.-Proteom.* **2018**, *15*, 41–51. [CrossRef]
21. Parry, R.M.; Jones, W.; Stokes, T.H.; Phan, J.H.; Moffitt, R.; Fang, H.; Shi, L.; Oberthuer, A.; Fischer, M.; Tong, W.; et al. k-Nearest neighbor models for microarray gene expression analysis and clinical outcome prediction. *Pharmacogenom. J.* **2010**, *10*, 292–309. [CrossRef]

22. Wright, M.N.; Ziegler, A. Ranger: A fast implementation of random forests for high dimensional data in C++ and R. *J. Stat. Softw.* **2017**, *77*, 1–17. [CrossRef]
23. Nagaraj, P.; Deepalakshmi, P.; Ijaz, M.F. Optimized adaptive tree seed Kalman filter for a diabetes recommen-dation system—Bilevel performance improvement strategy for healthcare applications. In *Intelligent Data-Centric Systems, Cognitive and Soft Computing Techniques for the Analysis of Healthcare Data*; Academic Press: Cambridge, MA, USA, 2022; pp. 191–202.
24. Mantzaris, D.H.; Anastassopoulos, G.C.; Lymberopoulos, D.K. Medical disease prediction using Artificial Neural Networks. In Proceedings of the 8th IEEE International Conference on BioInformatics and BioEngineering, Athens, Greece, 8–10 October 2008; pp. 1–6. [CrossRef]
25. Huang, P.-J.; Chang, J.-H.; Lin, H.-H.; Li, Y.-X.; Lee, C.-C.; Su, C.-T.; Li, Y.-L.; Chang, M.-T.; Weng, S.; Cheng, W.-H.; et al. DeepVariant-on-Spark: Small-Scale Genome Analysis Using a Cloud-Based Computing Framework. *Comput. Math. Methods Med.* **2020**, *2020*, 7231205. [CrossRef]
26. Koumakis, L. Deep learning models in genomics; are we there yet? *Comput. Struct. Biotechnol. J.* **2020**, *18*, 1466–1473. [CrossRef]
27. Van Dam, S.; Võsa, U.; Van Der Graaf, A.; Franke, L.; De Magalhães, J.P. Gene co-expression analysis for functional classification and gene–disease predictions. *Brief. Bioinform.* **2018**, *19*, 575–592. [CrossRef] [PubMed]
28. Travnik, J.B.; Mathewson, K.W.; Sutton, R.S.; Pilarski, P.M. Reactive Reinforcement Learning in Asynchronous Environments. *Front. Robot. AI* **2018**, *5*, 79. [CrossRef] [PubMed]
29. Battineni, G.; Sagaro, G.G.; Chinatalapudi, N.; Amenta, F. Applications of Machine Learning Predictive Models in the Chronic Disease Diagnosis. *J. Pers. Med.* **2020**, *10*, 21. [CrossRef] [PubMed]
30. Yue, C.; Xin, L.; Kewen, X.; Chang, S. An Intelligent Diagnosis to Type 2 Diabetes Based on QPSO Algorithm and WLS-SVM. In Proceedings of the 2008 International Symposium on Intelligent Information Technology Application Workshops, Shanghai, China, 21–22 December 2008; pp. 117–121. [CrossRef]
31. Srinivasu, P.N.; Rao, T.S.; Dicu, A.M.; Mnerie, C.A.; Olariu, I. A comparative review of optimisation techniques in segmentation of brain MR images. *J. Intell. Fuzzy Syst.* **2020**, *38*, 6031–6043. [CrossRef]
32. Nadesh, R.K.; Arivuselvan, K. Type 2: Diabetes mellitus prediction using Deep Neural Networks classifier. *Int. J. Cogn. Comput. Eng.* **2020**, *1*, 55–61. [CrossRef]
33. Abedini, M.; Bijari, A.; Banirostam, T. Classification of Pima Indian Diabetes Dataset using Ensemble of Decision Tree, Logistic Regression and Neural Network. *Int. J. Adv. Res. Comput. Commun. Eng.* **2020**, *9*, 1–4. [CrossRef]
34. Kundu, N.; Rani, G.; Dhaka, V.S.; Gupta, K.; Nayak, S.C.; Verma, S.; Ijaz, M.F.; Woźniak, M. IoT and Interpretable Machine Learning Based Framework for Disease Prediction in Pearl Millet. *Sensors* **2021**, *21*, 5386. [CrossRef]
35. Reddy, G.S.; Chittineni, S. Entropy based C4.5-SHO algorithm with information gain optimization in data mining. *PeerJ Comput. Sci.* **2021**, *7*, e424. [CrossRef]
36. Luukka, P. Feature selection using fuzzy entropy measures with similarity classifier. *Expert Syst. Appl.* **2011**, *38*, 4600–4607. [CrossRef]
37. Szmidt, E.; Kacprzyk, J. Some Problems with Entropy Measures for the Atanassov Intuitionistic Fuzzy Sets. In *Applications of Fuzzy Theory, Proceedings of the WILF 2007, Camogli, Italy, 7–10 July 2007*; Lecture Notes in Computer Science; Masulli, F., Mitra, S., Pasi, G., Eds.; Springer: Berlin/Heidelberg, Germany, 2007; Volume 4578. [CrossRef]
38. Choubey, D.K.; Paul, S. GA_RBF NN: A classification system for diabetes. *Int. J. Biomed. Eng. Technol.* **2017**, *23*, 71–93. [CrossRef]
39. Jackins, V.; Vimal, S.; Kaliappan, M.; Lee, M.Y. AI-based smart prediction of clinical disease using random forest classifier and Naive Bayes. *J. Supercomput.* **2020**, *77*, 5198–5219. [CrossRef]
40. Almustafa, K.M. Prediction of heart disease and classifiers' sensitivity analysis. *BMC Bioinform.* **2020**, *21*, 278. [CrossRef] [PubMed]
41. Tayeb, S.; Pirouz, M.; Sun, J.; Hall, K.; Chang, A.; Li, J.; Song, C.; Chauhan, A.; Ferra, M.; Sager, T.; et al. Toward predicting medical conditions using k-nearest neighbors. In Proceedings of the 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, USA, 11–14 December 2017; pp. 3897–3903. [CrossRef]
42. Xu, W.; Zhao, Y.; Nian, S.; Feng, L.; Bai, X.; Luo, X.; Luo, F. Differential analysis of disease risk assessment using binary logistic regression with different analysis strategies. *J. Int. Med. Res.* **2018**, *46*, 3656–3664. [CrossRef]
43. Wei, W.; Visweswaran, S.; Cooper, G.F. The application of naive Bayes model averaging to predict Alzheimer's disease from genome-wide data. *J. Am. Med. Inform. Assoc.* **2011**, *18*, 370–375. [CrossRef]
44. Benbelkacem, S.; Atmani, B. Random Forests for Diabetes Diagnosis. In Proceedings of the 2019 International Conference on Computer and Information Sciences (ICCIS), Sakaka, Saudi Arabia, 10–11 April 2019; pp. 1–4. [CrossRef]
45. Son, Y.-J.; Kim, H.-G.; Kim, E.-H.; Choi, S.; Lee, S.-K. Application of Support Vector Machine for Prediction of Medication Adherence in Heart Failure Patients. *Health Inform. Res.* **2010**, *16*, 253–259. [CrossRef]
46. Ghaheri, A.; Shoar, S.; Naderan, M.; Hoseini, S.S. The Applications of Genetic Algorithms in Medicine. *Oman Med. J.* **2015**, *30*, 406–416. [CrossRef]
47. Swapna, G.; Soman, K.P.; Vinayakumar, R. Diabetes Detection Using ECG Signals: An Overview. In *Deep Learning Techniques for Biomedical and Health Informatics*; Studies in Big Data; Dash, S., Acharya, B., Mittal, M., Abraham, A., Kelemen, A., Eds.; Springer: Cham, Switzerland, 2019; Volume 68. [CrossRef]

48. Available online: https://www.forrester.com/webinar/AI+Software+Market+Sizing+Understand+Forresters+Four+Segments+To+Invest+Wisely/-/E-WEB32605?utm_source=prnewswire&utm_medium=pr&utm_campaign=cio20 (accessed on 7 October 2021).
49. Mooney, M.A.; Wilmot, B. Gene set analysis: A step-by-step guide. *Am. J. Med. Genet. Part B Neuropsychiatr. Genet.* **2015**, *168*, 517–527. [CrossRef]
50. Mathur, R.; Rotroff, D.; Ma, J.; Shojaie, A.; Motsinger-Reif, A. Gene set analysis methods: A systematic comparison. *BioData Min.* **2018**, *11*, 8. [CrossRef]
51. Leevy, J.L.; Khoshgoftaar, T.M.; Villanustre, F. Survey on RNN and CRF models for de-identification of medical free text. *J. Big Data* **2020**, *7*, 73. [CrossRef]
52. Yadav, S.S.; Jadhav, S.M. Deep convolutional neural network based medical image classification for disease diagnosis. *J. Big Data* **2019**, *6*, 113. [CrossRef]
53. SivaSai, J.G.; Srinivasu, P.N.; Sindhuri, M.N.; Rohitha, K.; Deepika, S. An Automated Segmentation of Brain MR Image through Fuzzy Recurrent Neural Network. In *Bio-Inspired Neurocomputing*; Studies in Computational Intelligence; Bhoi, A., Mallick, P., Liu, C.M., Balas, V., Eds.; Springer: Singapore, 2020; Volume 903. [CrossRef]
54. Ahmed, S.; Srinivasu, P.N.; Alhumam, A.; Alarfaj, M. AAL and Internet of Medical Things for Monitoring Type-2 Diabetic Patients. *Diagnostics* **2022**, *12*, 2739. [CrossRef] [PubMed]
55. Kozomara, A.; Birgaoanu, M.; Griffiths-Jones, S. miRBase: From microRNA sequences to function. *Nucleic Acids Res.* **2019**, *47*, D155–D162. [CrossRef] [PubMed]
56. Guan, Z.-X.; Li, S.-H.; Zhang, Z.-M.; Zhang, D.; Yang, H.; Ding, H. A Brief Survey for MicroRNA Precursor Identification Using Machine Learning Methods. *Curr. Genom.* **2020**, *21*, 11–25. [CrossRef] [PubMed]
57. Hira, Z.M.; Gillies, D.F. A Review of feature selection and future extraction methods applied on microarray data. *Adv. Bioinform.* **2015**, *2015*, 198363. [CrossRef] [PubMed]
58. Shirzad, M.B.; Keyvanpour, M.R. A feature selection method based on minimum redundancy maximum relevance for learning to rank. In Proceedings of the 2015 AI & Robotics (IRANOPEN), Qazvin, Iran, 12–12 April 2015; pp. 1–5. [CrossRef]
59. Fang, H.; Tang, P.; Si, H. Feature Selections Using Minimal Redundancy Maximal Relevance Algorithm for Human Activity Recognition in Smart Home Environments. *J. Health Eng.* **2020**, *2020*, 8876782. [CrossRef]
60. Carrara, F.; Elias, P.; Sedmidubsky, J.; Zezula, P. LSTM-based real-time action detection and prediction in human motion streams. *Multimed. Tools Appl.* **2019**, *78*, 27309–27331. [CrossRef]
61. Che, Z.; Purushotham, S.; Cho, K.; Sontag, D.; Liu, Y. Recurrent Neural Networks for Multivariate Time Series with Missing Values. *Sci. Rep.* **2018**, *8*, 6085. [CrossRef]
62. Srinivasu, P.N.; JayaLakshmi, G.; Jhaveri, R.H.; Praveen, S.P. Ambient Assistive Living for Monitoring the Physical Activity of Diabetic Adults through Body Area Networks. *Mob. Inf. Syst.* **2022**, *2022*, 3169927. [CrossRef]
63. Wu, L.; Kong, C.; Hao, X.; Chen, W. A Short-Term Load Forecasting Method Based on GRU-CNN Hybrid Neural Network Model. *Math. Probl. Eng.* **2020**, *2020*, 1428104. [CrossRef]
64. Swapna, G.; Soman, K.; Vinayakumar, R. Automated detection of diabetes using CNN and CNN-LSTM network and heart rate signals. *Procedia Comput. Sci.* **2018**, *132*, 1253–1262.
65. Srinivasu, P.N.; SivaSai, J.G.; Ijaz, M.F.; Bhoi, A.K.; Kim, W.; Kang, J.J. Classification of skin disease using deep learning neural networks with MobileNet V2 and LSTM. *Sensors* **2021**, *21*, 2852. [CrossRef]
66. Ijaz, M.F.; Alfian, G.; Syafrudin, M.; Rhee, J. Hybrid Prediction Model for Type 2 Diabetes and Hypertension Using DBSCAN-Based Outlier Detection, Synthetic Minority Over Sampling Technique (SMOTE), and Random Forest. *Appl. Sci.* **2018**, *8*, 1325. [CrossRef]
67. Zhang, L.; Wang, Y.; Niu, M.; Wang, C.; Wang, Z. Machine learning for characterizing risk of type 2 diabetes mellitus in a rural Chinese population: The Henan Rural Cohort Study. *Sci. Rep.* **2020**, *10*, 4406. [CrossRef]
68. Naz, H.; Ahuja, S. Deep learning approach for diabetes prediction using PIMA Indian dataset. *J. Diabetes Metab. Disord.* **2020**, *19*, 391–403. [CrossRef]
69. Hertzog, M.I.; Correa, U.B.; Araujo, R.M. SpreadOut: A Kernel Weight Initializer for Convolutional Neural Networks. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–7. [CrossRef]
70. Yang, B.; Urtasun, R. Learning to Reweight Examples for Robust Deep Learning. In Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018.
71. Mitra, A.; Mohanty, D.; Ijaz, M.F.; Rana, A.u.H.S. Deep Learning Approach for Object Features Detection. In *Advances in Communication, Devices, and Networking*; Lecture Notes in Electrical Engineering; Dhar, S., Mukhopadhyay, S.C., Sur, S.N., Liu, C.M., Eds.; Springer: Singapore, 2022; Volume 776.
72. Pranto, B.; Mehnaz, S.M.; Mahid, E.B.; Sadman, I.M.; Rahman, A.; Momen, S. Evaluating Machine Learning Methods for Predicting Diabetes among Female Patients in Bangladesh. *Information* **2020**, *11*, 374. [CrossRef]
73. Lai, H.; Huang, H.; Keshavjee, K.; Guergachi, A.; Gao, X. Predictive models for diabetes mellitus using machine learning techniques. *BMC Endocr. Disord.* **2019**, *19*, 101. [CrossRef]
74. Web-Based Data-Science Environment. Available online: https://www.kaggle.com/ (accessed on 8 January 2022).

75. Ontor, Z.H.; Ali, M.; Hossain, S.S.; Nayer., M.; Ahmed, K.; Bui, F.M. YOLO_CC: Deep Learning based Approach for Early Stage Detection of Cervical Cancer from Cervix Images Using YOLOv5s Model. In Proceedings of the 2022 Second International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Bhilai, India, 21–22 April 2022; pp. 1–5. [CrossRef]
76. Srinivasu, P.N.; Rao, T.S.; Balas, V.E. A systematic approach for identification of tumor regions in the human brain through HARIS algorithm. In *Deep Learning Techniques for Biomedical and Health Informatics*; Academic Press: Cambridge, MA, USA, 2020; pp. 97–118. [CrossRef]
77. Tigga, N.P.; Garg, S. Prediction of Type 2 Diabetes using Machine Learning Classification Methods. *Procedia Comput. Sci.* **2020**, *167*, 706–716. [CrossRef]
78. Larabi-Marie-Sainte, S.; Aburahmah, L.; Almohaini, R.; Saba, T. Current Techniques for Diabetes Prediction: Review and Case Study. *Appl. Sci.* **2019**, *9*, 4604. [CrossRef]
79. Ijaz, M.F.; Attique, M.; Son, Y. Data-Driven Cervical Cancer Prediction Model with Outlier Detection and Over-Sampling Methods. *Sensors* **2020**, *20*, 2809. [CrossRef]
80. Vulli, A.; Srinivasu, P.N.; Sashank, M.S.K.; Shafi, J.; Choi, J.; Ijaz, M.F. Fine-Tuned DenseNet-169 for Breast Cancer Metastasis Prediction Using FastAI and 1-Cycle Policy. *Sensors* **2022**, *22*, 2988. [CrossRef]
81. Chae, S.; Kwon, S.; Lee, D. Predicting Infectious Disease Using Deep Learning and Big Data. *Int. J. Environ. Res. Public Health* **2018**, *15*, 1596. [CrossRef] [PubMed]
82. Pinto, M.F.; Oliveira, H.; Batista, S.; Cruz, L.; Pinto, M.; Correia, I.; Martins, P.; Teixeira, C. Prediction of disease progression and outcomes in multiple sclerosis with machine learning. *Sci. Rep.* **2020**, *10*, 21038. [CrossRef] [PubMed]
83. Srinivasu, P.N.; Balas, V.E. Self-Learning Network-based segmentation for real-time brain M.R. images through HARIS. *PeerJ Comput. Sci.* **2021**, *7*, e654. [CrossRef]

diagnostics

MDPI

Article

Hamlet-Pattern-Based Automated COVID-19 and Influenza Detection Model Using Protein Sequences

Mehmet Erten [1], Madhav R. Acharya [2], Aditya P. Kamath [3], Niranjana Sampathila [2], G. Muralidhar Bairy [2], Emrah Aydemir [4], Prabal Datta Barua [5,6], Mehmet Baygin [7], Ilknur Tuncer [8], Sengul Dogan [9,*] and Turker Tuncer [9]

[1] Laboratory of Medical Biochemistry, Malatya Training and Research Hospital, 44000 Malatya, Turkey
[2] Department of Biomedical Engineering, Manipal Academy of Higher Education, Manipal 04478, India
[3] Center for Biomedical Engineering, Brown University, Providence, RI 02912, USA
[4] Department of Management Information, College of Management, Sakarya University, 54050 Sakarya, Turkey
[5] School of Management & Enterprise, University of Southern Queensland, Toowoomba, QLD 4350, Australia
[6] Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, NSW 2007, Australia
[7] Department of Computer Engineering, Faculty of Engineering, Ardahan University, 75000 Ardahan, Turkey
[8] Elazig Governorship, Interior Ministry, 23119 Elazig, Turkey
[9] Department of Digital Forensics Engineering, Technology Faculty, Firat University, 23119 Elazig, Turkey
* Correspondence: sdogan@firat.edu.tr; Tel.: +90-424-2370000-7634

Abstract: SARS-CoV-2 and Influenza-A can present similar symptoms. Computer-aided diagnosis can help facilitate screening for the two conditions, and may be especially relevant and useful in the current COVID-19 pandemic because seasonal Influenza-A infection can still occur. We have developed a novel text-based classification model for discriminating between the two conditions using protein sequences of varying lengths. We downloaded viral protein sequences of SARS-CoV-2 and Influenza-A with varying lengths (all 100 or greater) from the NCBI database and randomly selected 16,901 SARS-CoV-2 and 19,523 Influenza-A sequences to form a two-class study dataset. We used a new feature extraction function based on a unique pattern, HamletPat, generated from the text of Shakespeare's *Hamlet*, and a signum function to extract local binary pattern-like bits from overlapping fixed-length (27) blocks of the protein sequences. The bits were converted to decimal map signals from which histograms were extracted and concatenated to form a final feature vector of length 1280. The iterative Chi-square function selected the 340 most discriminative features to feed to an SVM with a Gaussian kernel for classification. The model attained 99.92% and 99.87% classification accuracy rates using hold-out (75:25 split ratio) and five-fold cross-validations, respectively. The excellent performance of the lightweight, handcrafted HamletPat-based classification model suggests that it can be a valuable tool for screening protein sequences to discriminate between SARS-CoV-2 and Influenza-A infections.

Keywords: Hamlet Pattern; protein sequence classification; SARS-CoV-2; bioinformatics

Citation: Erten, M.; Acharya, M.R.; Kamath, A.P.; Sampathila, N.; Bairy, G.M.; Aydemir, E.; Barua, P.D.; Baygin, M.; Tuncer, I.; Dogan, S.; et al. Hamlet-Pattern-Based Automated COVID-19 and Influenza Detection Model Using Protein Sequences. *Diagnostics* **2022**, *12*, 3181. https://doi.org/10.3390/diagnostics12123181

Academic Editors: Hiam Alquran and Wan Azani Mustafa

Received: 11 November 2022
Accepted: 12 December 2022
Published: 15 December 2022

1. Introduction

Understanding how viruses interact with host cells for their life cycle is essential for understanding viral pathogenesis. In addition, the viral genetic codes enable these interactions. The unique protein sequences that distinguish between viruses are a crucial cornerstone. For this purpose, medical laboratories make great efforts. On the other hand, in silico approaches cover a wider place in today's academic research than ever before. This study proposes a method based on viral protein sequences that can classify SARS-CoV-2, which has scared the world in recent years, and the Influenza-A virus, which causes difficulties in differential diagnosis [1].

SARS-CoV-2 is a new type of coronavirus that emerged in December 2019, causing severe acute respiratory distress syndrome [1]. It spread worldwide, resulting in a pandemic named COVID-19. Coronaviridae family members are enveloped by single-stranded

positive-sense Ribonucleic acid (RNA) viruses [2]. SARS-CoV-2 [3] shares a 50% similarity with the MERS-CoV genome. Additionally, it has 88% similarity to two SARS-like coronaviruses derived from bat species [4]. The SARS-CoV-2 genome contains 14 open reading frames (ORFs) that encode about 27 proteins [5], including four structural (surface (spike) (S), envelope (E), matrix (membrane) (M)) proteins, and a nucleocapsid (N) protein [5]. The S protein binds to host cells and exhibits tropism towards the angiotensin-converting enzyme receptors (ACE-2s) [6]. ACE-2 is highly expressed in alveolar type 2 epithelial cells, which explains the vulnerability to pneumonia [7]. The S protein consists of two subunits: S1 binds to the receptor on the host cell membrane, while S2 fuses the viral and host membranes [8]. Therefore, these proteins are potential therapeutic drug targets [8,9].

The influenza virus is an RNA virus belonging to the Orthomyxoviridae family that can infect a variety of human and animal hosts. Influenza B and C are more common in humans, while Influenza D infects animals such as goats and pigs [10]. The genomes of Influenza A and B viruses contain eight RNA segments that encode 18 different proteins. Influenza C and D lack the fifth segment responsible for neuraminidase. The fourth gene encodes hemagglutinin, and other genes encode viral polymerases, nucleoproteins, major matrix protein, ion channel matrix protein, and other nonstructural proteins [11]. Influenza-A viruses are named according to the structural differences of neuraminidase (N1-11) and hemagglutinin glycoproteins (H1–18), which are on the surface of the envelope [3]. These envelope proteins play an important step in viral pathogenesis by binding to the host cells [12]. This stage is also an important drug target [13].

The viral pathophysiology that we previously mentioned, and the etiopathogenesis of influenza explained in the prior section, should illustrate why the viral genomes are the crucial component of these diseases and their therapies. SARS-CoV-2 and Influenza-A can present with similar symptoms, and Influenza-A infection may be more problematic for patients during the current COVID-19 pandemic [14]. In addition, computer-aided diagnosis could help screen for the two pathological conditions requiring different treatment and isolation protocols [15].

Our main aim is to propose a new machine learning model to classify COVID-19 and Influenza-A diseases with high classification performance. Moreover, we propose a lightweight protein classification model. This model uses a new feature extraction technique named Hamlet Pattern. We in fact propose a new feature extraction methodology, and Hamlet Pattern is the first feature generator of the presented feature extraction methodology. This methodology is named text-based feature extractor creation. To take attention, we have used the popular text, *Hamlet*. In the Materials and Methods we present the main steps in the creation of our text-based feature extractor. Our main hypothesis is that literary texts have harmonies, which have been created using hidden patterns. In this paper we propose a new feature extraction methodology to obtain these hidden patterns. We used a piece of text from *Hamlet* to generate a new pattern, and the features were extracted using this pattern. We tested the feature generation ability of the presented feature extraction function using a protein sequence dataset to classify COVID-19 and Influenza-A.

Machine learning and artificial intelligence models are commonly used in the biomedical and bioinformatics sciences to solve classification problems [16–18]. Therefore, we were motivated to develop a computationally lightweight machine learning model for automated SARS-CoV-2 versus Influenza-A diagnosis. Feature engineering is an important aspect of machine learning [19,20]. One popular handcrafted feature generator is the local binary pattern (LBP) [21], which extracts local textual features based on the neighborhood relations of overlapping blocks. LBP possesses the advantages of simple application, low time complexity, ability to generate distinctive features, and fixed-size feature vector creation. We were motivated to develop an LBP-like feature extractor based on a novel pattern inspired by art. To this end, we selected text from a famous literary work, *Hamlet*, to create a novel pattern, combined with the feature selection function and a standard shallow classifier to form a handcrafted learning model.

The contributions of the proposed model are as follows:

- A novel feature extraction method based on a novel pattern that was inspired by a literary work. The presented feature extraction method is the first text-based feature extraction function creation methodology.
- Using protein sequences, a classification model incorporating the novel pattern was applied for the binary classification of SARS-CoV-2 versus Influenza-A diagnosis. The model attained excellent classification performance, supporting its potential use as an adjunctive screening tool for suspected viral respiratory infections in the current pandemic.

The paper is organized as follows: The dataset is shown in Section 2. Section 2 describes our proposed protein sequence classification model. Section 3 presents the results. Section 4 presents a discussion of the results. Section 5 outlines the conclusions.

2. Materials and Methods

2.1. Materials

In FASTA format, we downloaded viral protein sequences of SARS-CoV-2 and Influenza-A with lengths 100 or greater from the NCBI database [22]. Among these, we randomly selected 16,901 SARS-CoV-2 and 19,523 Influenza-A sequences to form a two-class study dataset with 36,424 observations.

2.2. Our Proposed Protein Sequence Classification Model

Shakespeare's *Hamlet* inspired our novel handcrafted feature engineering method. In Act 1 Scene 1, an apparition is before Horatio, who initially doubts but later acknowledges it to be the ghost of the recently deceased King Hamlet [23]. Horatio, the speaker of truth, i.e., the oracle, and confidant of the protagonist Prince Hamlet, later informed him of the encounter, thus setting the unfolding of subsequent tragic events in motion. By using letters from this text, HamletPat is applied to extract features from protein sequences. The protein sequences were coded with letters because the amino acids are named with text. We converted these texts to numerical values. HamletPat extracts a feature vector from these numerical values. An iterative Chi-square (IChi2) application [24] was then deployed to choose the most discriminative features to feed a support vector machine (SVM) [25,26] for classification using hold-out (75:25 split ratio) as well as 5-fold cross-validations (CVs) (see Figure 1).

The basic steps of the model are listed below (details are provided in the following sections):

Step 0: Load/read each protein sequence from the dataset and convert the amino acid sequence conventionally denoted by letters of the alphabet to a string of numbers.

Step 1: Extract features from each protein sequence using HamletPat.

Step 2: Select highly discriminative features from the generated feature vector by deploying the IChi2 feature selection function.

Step 3: Classify selected features by deploying the SVM classifier with two validation techniques, hold-out (split ratio 75:25) and 5-fold CVs.

2.2.1. Feature Extraction Using HamletPat

A novel text-generated pattern, HamletPat, was used to extract LBP-like features from overlapping fixed-length blocks (27) of protein sequences of different lengths. A block diagram of the proposed feature extraction function is shown in Figure 2.

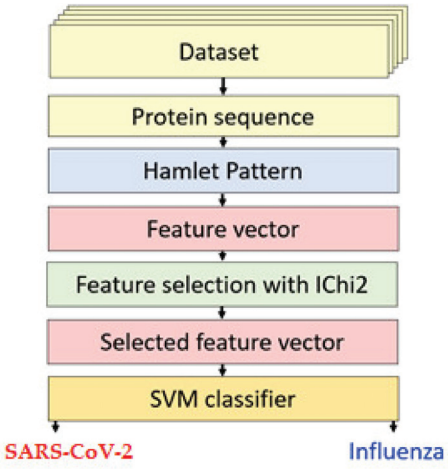


Figure 1. Schema of the proposed HamletPat-based model for binary classification of viral protein sequences.

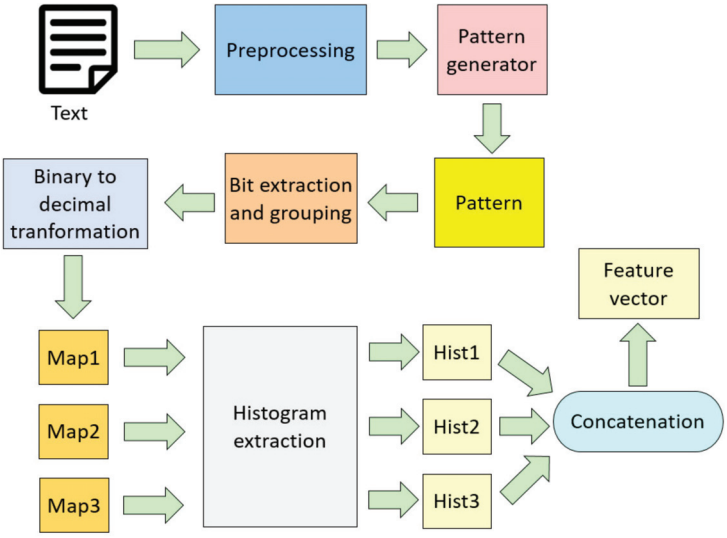


Figure 2. Block diagram of the proposed text-based feature extraction function generation model. We used *Hamlet* as a text in this paper. In the figure, Map defines feature map signals, and Hist is histogram.

First, text from the first page of *Hamlet* Act 1 Scene 1 was pre-processed by deconstructing it into letters of the English alphabet, with special characters deleted and all uppercase letters converted to lowercase. Next, the letters were enumerated from 1 to 26 using ASCII conversion and input to a pattern generator algorithm. The output, HamletPat, was then used to extract bits, similar to LBP feature extraction, from overlapping fixed-length blocks of protein sequences of varying lengths to construct the final feature vector. The feature engineering steps are detailed below:

1: Choose the text. In this model, we chose the text from *Hamlet* Act 1 Scene 1.
2: Remove all special characters in the text.
3: Transform all uppercase letters to lowercase.

4: Enumerate letters by using their ASCII code.

$$val = ascii(lc) - 96 \tag{1}$$

where val defines value and $ascii(.)$ is the ASCII value conversion function of lc, the lowercase character. The frequency histogram of the enumerated letters is shown in Figure 3.

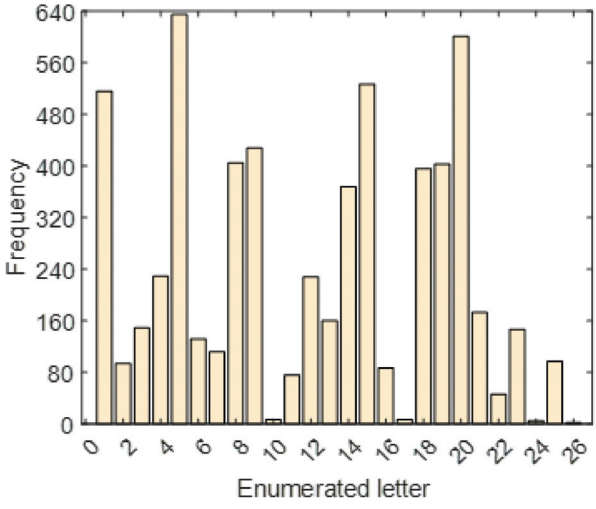


Figure 3. Frequency histogram of the enumerated letters used in the selected *Hamlet* text.

5: Generate a pattern by deploying Algorithm 1.

Algorithm 1. Pattern generator using enumerated letters.

Input: The calculated values of the letters
Output: Pattern

01: **for** i = 1 to 26 **do** // Assign counter
02: $counter(i) = 0$;
03: **end for i**
04: i = 1; j = 1; // Define variables
05: $sum = \sum_{i=1}^{26} counter(i)$;
06: **while** $sum < 26$ **do**
07: $v = val(i)$;
08: **if** $counter(v) = 0$ **then**
09: $counter(v) = 1$;
10: $pattern(j) = v$;
11: $j++$;
12: **end if**
13: $sum = \sum_{i=1}^{26} counter(i)$;
14: $i++$;
15: **end while**

Computed values or indexes in the pattern array, which are arranged in ascending order of the identifying number of the enumerated letters (Table 1), constituted the HamletPat.

Table 1. Pattern array of the computed Hamlet Pattern.

id	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
Ind.	2	5	18	14	1	4	15	23	8	19	20	6	3
id	14	15	16	17	18	19	20	21	22	23	24	25	26
Ind.	9	25	13	21	12	7	22	11	16	17	24	10	26

Here "id" represents the identifying number of the enumerated letter (e.g., "1" and "26" represent "a" and "z", respectively) and "Ind." represents the corresponding computed index.

6: Divide the signal/sequence into overlapping blocks with a length of 27 each.

$$b = s(j+i-1),\ j \in \{1,2,\ldots,27\},\ i \in \{1,2,\ldots,l-26\} \tag{2}$$

where s represents the utilized one-dimensional signal with a length of l and b represents the overlapping block with a size of 27. To create overlapping blocks, frameshift is defined as one (stride = 1). Therefore, $l-26$ overlapping blocks with a length of 27 were created from a one-dimensional signal/array with a length of l.

7: Choose the center value (14th value) as the center.

$$c = b(14) \tag{3}$$

where c is the center value.

8: Enumerate all other values, skipping the 14th value.

$$d(h) = b(h),\ h \in \{1,2,\ldots,13\} \tag{4}$$

$$d(k-1) = b(k),\ k \in \{15,16,\ldots,27\} \tag{5}$$

where d represents renumbered values with a length of 26.

9: Create bits using the HamletPat, renumbered values, center value, and signum function.

$$bf(i) = sign(center, d(pattern(i))),\ i \in \{1,2,\ldots,26\} \tag{6}$$

$$sign(q,w) = \begin{cases} 0,\ q-w \geq 0 \\ 1, q-w < 0 \end{cases} \tag{7}$$

where bf represents bits; $sign(.,.)$, signum function; and (q,w), parameters of the signum function.

10: Divide the generated bits into three groups with bit lengths 9, 8, and 9, respectively.

$$first(r) = bf(r),\ r \in \{1,2,\ldots,9\} \tag{8}$$

$$second(t) = bf(r+t),\ t \in \{1,2,\ldots,8\} \tag{9}$$

$$third(r) = bf(r+17) \tag{10}$$

11: Calculate three map signals from the three-bit groups using binary-to-decimal conversion.

$$map_1(i) = \sum_{j=1}^{9} first(j) \times 2^{9-j} \tag{11}$$

$$map_2(i) = \sum_{j=1}^{8} second(j) \times 2^{8-j} \tag{12}$$

$$map_3(i) = \sum_{j=1}^{9} third(j) \times 2^{9-j} \tag{13}$$

12: Extract histograms ($hist_1$, $hist_2$, $hist_3$) from the corresponding map_1, map_2, and map_3, which have lengths equal to 512 (=2^9), 256 (=2^8), and 512 (=2^9), respectively.

13: Merge the generated histograms to obtain the feature vector, which has a length of 1280.

$$ftvec(p) = hist_1(p),\ p \in \{1, 2, \ldots, 512\} \tag{14}$$

$$ftvec(z + 512) = hist_2(z),\ z \in \{1, 2, \ldots, 256\} \tag{15}$$

$$ftvec(p + 768) = hist_3(p) \tag{16}$$

2.2.2. Iterative Chi-Square Feature Selection

For feature selection, we deployed IChi2, which uses the parametric Chi-square function, one of the fastest in the literature [27], to compute qualified indexes of the features. IChi2 is efficient at iteratively selecting highly discriminative features using fewer features, effectively reducing the execution times of the classifiers. An iteration range is typically set, and then variable feature vectors are selected iteratively using a loss value calculator. Here we set the iteration range at (100–500), and an SVM with a Gaussian kernel (see Section 3.3 below) was deployed as the loss calculator using a 5-fold CV. In our experiments on the study dataset, the SVM calculated the loss values of 401 (=500 – 100 + 1) feature vectors, and IChi2 selected the optimal feature vector of length 340.

2.2.3. Classification

An SVM was deployed as a loss calculator (see Section 3.2 above) and classifier. The hyperparameters were set as: kernel function, Gaussian; kernel scale, 4.6; box constraint, one [25,26]. Both five-fold and hold-out CVs (split ratio 75:25, i.e., the dataset was randomly split into 75% and 25% for training and testing, respectively) were used for the classification task, whereas only the former was used for loss calculation during the IChi2 feature selection.

3. Results

3.1. Experimental Setup

The study dataset comprised two viral protein sequence classes, each typically notated as a string of letters corresponding to the individual amino acid's building blocks. An ASCII code table was used to transform these letters into integer values. The latter were then input to the proposed handcrafted HamletPat-based model. The model was computationally lightweight and was implemented in a MATLAB (2021b) environment on a personal computer with an Intel i9-9900 Processor (cache 16 M, clock speed 5 GHz) and 48 GB memory, using Microsoft Windows 10.1 Professional operating system. The presented Hamlet-Pattern-based model has linear time complexity. Therefore, more simply configured computers can be used for implementation. There is no need to use expensive hardware, for instance, a graphical processing unit (GPU) and a tensor processing unit (TPU). Therefore, this model can be implemented on any computer. Moreover, the proposed Hamlet Pattern model can be embedded in a card.

3.2. Evaluation Metrics

Standard metrics were used to evaluate the performance of the model for binary classification: accuracy (ac), sensitivity (sn), specificity (sp), precision (pr), F-measure ($f1$), and geometric mean (geomean). The metrics were calculated from the numbers of true positive (tp), true negative (tn), false positive (fp), and false negative (fn) results using Equations (17) to (22) [28,29].

$$ac = \frac{tp + tn}{tp + tn + fp + fn} \tag{17}$$

$$sn = recall = \frac{tp}{tp + fn} \tag{18}$$

$$sp = \frac{tn}{fp + tn} \tag{19}$$

$$pr = \frac{tp}{tp + fp} \tag{20}$$

$$f1 = 2\frac{pr \times sn}{pr + sn} \tag{21}$$

$$geomean = \sqrt{sp \times sn} \tag{22}$$

3.3. Performance of the Proposed Model

The model's performance metrics stratified by the validation schemes, i.e., hold-out versus five-fold CVs and virus type, are summarized in Table 2.

Table 2. Performance metrics for binary classification of viral protein sequences into SARS-CoV-2 versus Influenza-A using the HamletPat-based classification model.

Metric	Cross Validation	SARS-CoV-2	Influenza-A
Sensitivity (%)	5-fold CV	99.95	99.79
	75:25	100	99.86
Specificity (%)	5-fold CV	99.79	99.95
	75:25	99.86	100
Precision (%)	5-fold CV	99.76	99.96
	75:25	99.83	100
F1-score (%)	5-fold CV	99.86	99.87
	75:25	99.92	99.93
Overall accuracy (%)	5-fold CV	99.87	
	75:25	99.92	
Overall geometric mean (%)	5-fold CV	99.87	
	75:25	99.93	

The model attained 99.92% and 99.87% classification accuracy using hold-out and five-fold CVs, respectively. Of note, 100% sensitivity for SARS-CoV-2 detection was achieved by deploying a hold-out CV with a 75:25 split ratio.

ROC curves were added to evaluate this model. The ROC curves of both classes are demonstrated in Figure 4.

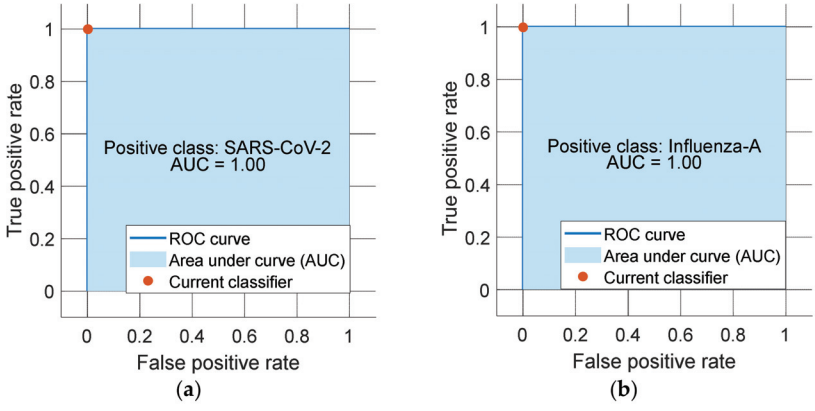


Figure 4. ROC curves of each class (**a**) SARS-CoV-2 and (**b**) Influenza-A.

As can be seen from Figure 4, the proposed model attained a 100% area under curve (AUC) value.

3.4. Time Complexity Analysis

We calculated the time complexity of the proposed model and describe the results in this section. The presented Hamlet Pattern is a handcrafted model. Therefore, the time burden of this feature extractor is equal to $O(n)$. Herein, n is the length of the signal. To choose features, the IChi2 feature selector was used. IChi2 uses Chi2, a loop, and a loss value calculator. Therefore, it is calculated as $O(s + ic)$. The used s, i, and c variables are the time burden coefficients of the feature selector, the number of loops, and the time burden coefficients of the classifier, respectively. In the classification phase, a shallow classifier is used, and its computational complexity is $O(c)$. In total, the time complexity of the presented Hamlet-Pattern-based classification model is $O(n + s + ic + c) \cong O(n + s + ic)$. This result demonstrates that the presented model has linear time complexity.

4. Discussion

SARS-CoV-2 and Influenza-A are very different pathogens that share important overlapping clinical features. In the current COVID-19 pandemic, SARS-CoV-2 has caused nearly six million deaths worldwide. The H1N1 Influenza-A virus was the cause of the Spanish flu that infected one-third of the world's population and killed millions from 1918 to 1920 [30]. Subsequent influenza-related pandemics had less-severe consequences [31]. According to the World Health Organization, about half a million people, mostly the elderly, die from seasonal influenza cases each year [3]. Both viruses are transmitted through the respiratory tract, and it is possible to be protected with non-pharmacological interventions such as masks [32]. Measures were taken amid the COVID-19 pandemic, and a reduction in global travel has caused a decrease in the number of seasonal influenza cases. However, it is expected that influenza cases will rise as the level of international travel is slowly being restored. Distinguishing infection from SARS-CoV-2 vs. Influenza-A is clinically difficult in the early stages of infection. It is also important to keep in mind the possibility of co-infection, which can exacerbate the clinical prognosis. During the influenza season, it is difficult to confidently secure clinical diagnosis due to similar symptom presentations, such as fever, cough, and dyspnea. In this context, real-time polymerase chain reaction and nucleic acid tests can be performed on suspected patients to determine the viral etiology and institute appropriate treatment and, if applicable, isolation procedures. Our main motivation in this study was to create an accurate algorithm that can be used to classify infection due to SARS-CoV-2 versus Influenza-A at the highest level.

We presented a new handcrafted text-based feature generation model that could accurately discriminate between SARS-CoV-2 and Influenza-A. The novel *Hamlet* [33,34] Pattern is a local texture feature extractor with low time complexity, $O(n)$. IChi2 selected the most valuable 340 features among the 1280 features created with HamletPat, thereby effectively reducing the execution time. Using two validation techniques, these selected features were then classified using an SVM with a Gaussian kernel. The calculated confusion matrices according to the validation technique are shown in Figure 5.

The performance of our model for SARS-CoV-2 versus Influenza-A diagnosis is compared with another study in the literature that involved the binary classification of SARS-CoV-2 versus human immunodeficiency virus [9] in Table 3. The HamletPat-based model outperformed the other model on a larger dataset.

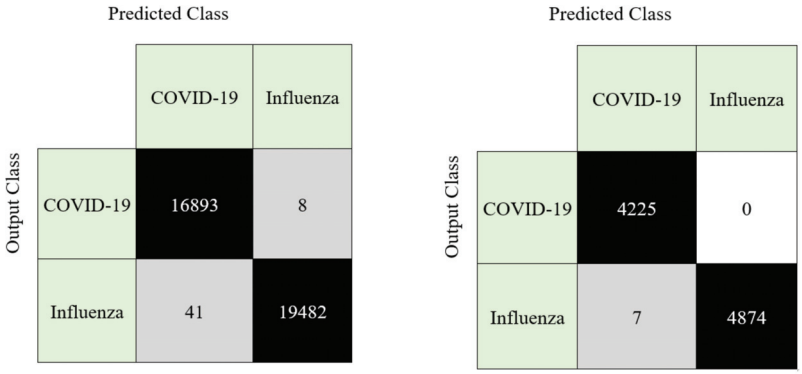


Figure 5. Confusion matrices of the HamletPat-based classification model using hold-out (split ratio 75:25) versus 5-fold cross-validations (CVs).

Table 3. Comparative results (%).

Model	Dataset	Number of Observations	Method	Result
Afify and Zanaty [9]	NCBI	18,476 protein sequences: 9238 COVID-19 9238 HIV	Conjoint triad feature extraction and Random Forest classification with hold-out validation (80:20)	Accuracy: 99.80%
Our model	NCBI	36,424 protein sequences: 16,901 COVID-19 19,523 Influenza-A	HamletPat feature extraction, IChi2 feature selection, and SVM classification with hold-out validation (75:25) and 5-fold CV	Accuracy: hold-out: 99.92% 5-fold CV: 99.87%

Table 3 shows that Afify and Zanaty [9] used a balanced protein sequence dataset with two classes, HIV and COVID-19. They achieved a 99.80% accuracy. We utilized an unbalanced dataset, and our classes are COVID-19 and Influenza-A (which have similar symptoms). Our proposal attained a 99.92% accuracy with a five-fold CV. Moreover, this model is the first text-based feature extractor generation methodology. The high classification capability of the features generated using HamletPat is clearly demonstrated. Moreover, we used a larger dataset than that of Afify and Zanaty's [9]. This result demonstrates the robust feature generation ability of HamletPat.

Moreover, using the presented HamletPat-based classification model, a decision-support system was created. By incorporating the 340 features selected, the decision support-system could be implemented. A graphical representation of the decision tree (pruning level = 10) for basic decision support is demonstrated in Figure 6.

By only using these nine rules (see Figure 6), 96.84% classification accuracy was attained on the used dataset. These nine rules demonstrated the explainability of the proposed system. By using HamletPat, features were created. Hidden rules were extracted by deploying the created features and a decision tree. These rules were created using features 167, 127, 210, 324, 9, 1, 242, 8, and 17 (see Figure 6). The proposed HamletPat is useful for creating a cognitive model to detect COVID-19. Figure 6 shows that an explainable artificial intelligence (XAI) model was proposed in this work. We were able to create a decision-support system by using these rules (see Figure 6), and this decision-support system can be embedded in a card to develop an embedded system.

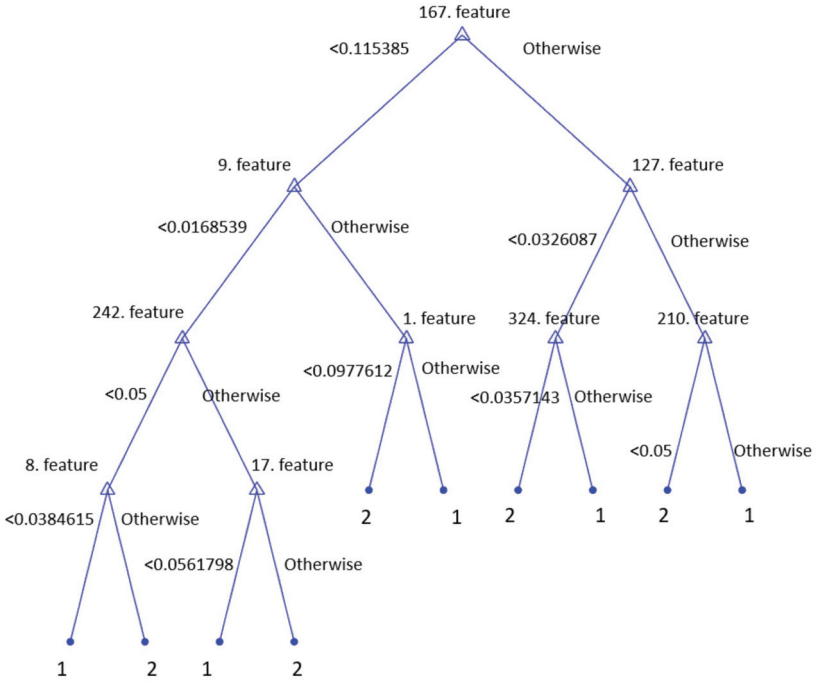


Figure 6. Rules of a basic decision-support system using our selected features. Herein, the symbol 1 denotes COVID-19, while 2 denotes influenza.

The MATLAB codes of the presented pattern generation are given in Appendix A (Tables A1 and A2).

The advantages and disadvantages of the novel HamletPat-based protein sequence classification model are listed below.

Advantages:

- Influenza and COVID-19 share similar symptoms, and clinical discrimination is difficult. Therefore, an automated protein-sequence-based model was developed to differentiate the disorders automatically.
- To our knowledge, HamletPat is the first text-based pattern utilized to create a new feature extraction function.
- The novel HamletPat-based classification model was trained on a two-class dataset and attained 99.87% and 99.92% accuracy rates by deploying a five-fold CV and hold-out (split ratio 75:25) CV, respectively.
- The model is simple, it has a low time complexity of $O(n)$, and is easy to implement.

Limitations:

- The model used overlapping blocks with a fixed length of 27. Therefore, the minimum length of the studied protein sequence should be 27 (we used a protein sequence with a length of 100 or greater in the study).
- We used the SVM classifier with default hyperparameters in the study. The hyperparameters can be further optimized using a metaheuristic optimization model.

5. Conclusions

To discriminate between SARS-CoV-2 and Influenza-A infections, we developed a new automated detection model for the binary classification of protein sequences. The handcrafted classification model used a textual-based pattern to extract 1280 features. IChi2 was used to select the 340 most discriminative features, and an SVM was used for

classification using two validation strategies, hold-out (75:25 split ratio) and five-fold CV, which attained 99.92% and 99.87% accuracy rates, respectively. These results suggest that the proposed model has discriminative utility for the binary classification of SARS-CoV-2 vs. Influenza-A on the basis of protein sequences.

Developments in personalized or precision medicine have grown apace in recent years, and medical understanding is evolving with the use of new technologies. As a result, computer-assisted algorithms have become increasingly more relevant and valuable. We hope that our new textual-based feature generation methodology can lend assistance in this regard, and that new-generation explainable artificial intelligence models can be built to aid clinicians [35]. For now, the salutary results of our lightweight, handcrafted classification model suggest that it can be applied as a useful adjunctive screening tool to discriminate between these two important viral respiratory conditions.

Author Contributions: Conceptualization, M.E., M.R.A., A.P.K., N.S., G.M.B., E.A., P.D.B., M.B., I.T., S.D. and T.T.; methodology, M.E., M.R.A., A.P.K., N.S., G.M.B., E.A., P.D.B., M.B., I.T., S.D. and T.T.; software, S.D. and TT.; validation, M.E., M.R.A., A.P.K., N.S., G.M.B., E.A. and P.D.B.; formal analysis, M.E., M.R.A., A.P.K., N.S. and G.M.B.; investigation, M.E., M.R.A., A.P.K., N.S., G.M.B., E.A. and P.D.B.; resources, M.E., M.R.A., A.P.K., N.S., G.M.B., E.A., P.D.B., M.B., I.T., S.D. and T.T.; data curation M.E., M.R.A., A.P.K., N.S., G.M.B. and E.A.; writing—original draft preparation, M.E., M.R.A., A.P.K., N.S., G.M.B., E.A., P.D.B., M.B., I.T., S.D. and T.T.; writing—review and editing, M.E., M.R.A., A.P.K., N.S., G.M.B., E.A., P.D.B., M.B., I.T., S.D. and T.T.; visualization, M.E., M.R.A., A.P.K., N.S. and G.M.B.; supervision, S.D. and T.T.; project administration, S.D. and T.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used in this study were downloaded from [22].

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

We implemented the proposed model using the MATLAB (2021b) programming environment. The novelty of this paper is HamletPat. The MATLAB codes of the presented pattern generation of the text-based feature generation methodology and HamletPat are given in Tables A1 and A2, respectively.

Table A1. Pattern generation code of the text-based feature generation methodology.

```
clc,clear all,close all
text = 'bernardowhostherefranciscoнayanswermestandandun-
foldyourselfbernardolonglivethekingfranciscobernardobernardohefrancisco-
youcomemostcarefullyuponyourhourbernardotisnowstrucktwelvegettheeto-
bedfranciscofranciscoforthisreliefmuchthankstisbittercoldandiamsickatheartber-
nardohaveyouhadquietguardfranciscoнotamousestirringbernardowellgoodnight-
ifyoudomeethoratioandmarcellustherivalsofmywatchbidthemmakehastefrancisco-
ithinkihearthemstandhowhosthereenterhoratioandmarcellushoratiofriendstothisg-
roundmarcellusandliegementothedanefranciscogiveyougoodnightmarcellusofarew-
ellhonestsoldierwhohathrelievedyoufranciscobernardohasmyplacegiveyougoodnig-
htexitmarcellushollabernardobernardosaywhatishoratiotherehoratioapieceofhimber-
nardowelcomehoratiowelomegoodmarcellusmarcelluswhathasthisthingappeardagai-
ntonightbernardoihaveseennothingmarcellushoratiosaystisbutourfantasyandwillnotl-
etbelieftakeholdofhimtouchingthisdreadedsighttwiceseenofusthereforeihaveentreatedh-
imalongwithustowatchtheminutesofthisnightthatifagainthisapparitioncomehemayappro-
veoureyesandspeaktoithoratiotushtushtwillnotappearbernardositdownawhileandletuson-
ceagainassailyourearsthataresofortifiedagainstourstorywhatwehavetwonightsseenhorati-
owellsitwedownandletushearbernardospeakofthisbernardolastnightofallwhenyondsame-
starthatswestwardfromthepolehadmadehiscoursetoillumethatpartofheavenwherenowitbu-
```

Table A1. *Cont.*

rnsmarcellusandmyselfthebellthenbeatingoneenterghostmarcelluspeacebreaktheeofflook- whereitcomesagainbernardointhesamefigurelikethekingthatsdeadmarcellusthouartascholar- speaktoithoratiobernardolooksitnotlikethekingmarkithoratiohoratiomostlikeitharrowsmew- ithfearandwonderbernardoitwouldbespoketomarcellusquestionithoratiohoratiowhatartthou- thatusurpstthistimeofnighttogetherwiththatfairandwarlikeforminwhichthemajestyofburiedd- enmarkdidsometimesmarchbyheavenichargetheespeakmarcellusitisoffendedbernardoseeitsta- lksawayhoratiostayspeakspeakichargetheespeakexitghostmarcellustisgoneandwillntanswerb- ernardohownowhoratioyoutrembleandlookpaleisnotthissomethingmorethanfantasywhatthink- youonthoratiobeforemygodimightnotthisbelievewithoutthesensibleandtrueavouchof- mineowneyesmarcellusisitnotlikethekinghoratioasthouarttothyselfsuchwastheveryarmou- rhehadonwhenhetheambitiousnorwaycombatedsofrowndheoncewheninanangryparlehe- smotethesleddedpolacksontheicetisstrangemarcellusthustwicebeforeandjumpatthisdeadh- ourwithmartialstalkhathhegonebyourwatchhoratioinwhatparticularthoughttoworkiknow- notbutinthegrossandscopeofmyopinionthisbodessomestrangeeruptiontoourstatemarcellu- sgoodnowsitdownandtellmehethatknowswhythissamestrictandmostobservantwatchsonig- htlytoilsthesubjectofthelandandwhysuchdailycastofbrazencannonandforeignmartforimpl- ementsofwarwhysuchimpressofshipwrightswhosesoretaskdoesnotdividethesundayfromth- eweekwhatmightbetowardthatthissweatyhastedothmakethenightjointlabourerwiththeday- whoistthatcaninformmehoratiothatcaniatleastthewhispergoessoourlastkingwhoseimagee- venbutnowappeardtouswasasyouknowbyfortinbrasofnorwaytheretoprickdonbyamostemu- latepridedaredtothecombatinwhichourvalianthamletforsothissideofourknownworldes- teemdhimdidslaythisfortinbraswhobyasealdcompactwellratifiedbylawandheraldrydidforfe- itwithhislifeallthosehislandswhichhestoodseizedoftotheconqueroragainstthewhichamoietycom- petentwasgagedbyourkingwhichhadreturndtotheinheritanceoffortinbrashadhebeenvanqui- sherasbythesamecovenantandcarriageofthearticledesigndhisfelltohamletnowsiryoungfortin- brasofunimprovedmettlehotandfullhathintheskirtsofnorwayhereandtheresharkdupalisto- flawlessresolutesforfoodanddiettosomeenterprisethathathastomachintwhichisnootherа- sitdothwellappearuntoourstatebuttorecoverofusbystronghandandtermscompulsatoryth- oseforesaidlandssobyhisfatherlostandthisitakeitisthemainmotiveofourpreparationsthesо- urceofthisourwatchandthechiefheadofthisposthasteandromageinthelandbernardoithink- itbenootherbuteensowellmayitsortthatthisportentousfigurecomesarmedthroughourwatch- solikethekingthatwasandisthequestionofthesewarshoratioamoteitistotroublethemindseye- inthemosthighandpalmystateofromealittleerethemightiestjuliusfellthegravesstoodtena- ntlessandthesheeteddeaddidsqueakandgibberintheromanstreetsasstarswithtrainsoff- ireanddewsofblooddisastersinthesunandthemoiststaruponwhoseinfluenceneptunesem- pirestandswassickalmosttodoomsdaywitheclipseandeventhelikeprecurseoffierceeven- tsasharbingersprecedingstillthefatesandprologuetothеomencomingonhaveheavenan- dearthtogetherdemonstrateduntoourclimaturesandcountrymenbutsoftbeholdlowherei- tcomesagainreenterghostillcrossitthoughitblastmestayillusionifthouhastanysoundo- ruseofvoicespeaktomeiftherebeanygoodthingtobedonethatmaytotheedoeasandgrac- etomespeaktomecockcrowsifthouartprivytothycountrysfatewhichhappilyforeknowing- mayavoidospeakorifthouhastuphoardedinthylifeextortedtreasureinthewombofearthf- orwhichtheysayyouspiritsoftwalkindeathspeakofitstayandspeakstopitmarcel- lusmarcellusshallistrikeatitwithmypartisanhoratiodoifitwillnotstandbernardotish- erehoratiotisheremarcellustisgoneexitghostwedoitwrongbeingsomajesticaltoofferit- theshowofviolenceforitisastheairinvulnerableandourvainblowsmaliciousmockeryb- ernardoitwasabouttospeakwhenthecockcrewhoratioandthenitstartedlikeaguiltythingu- ponafearfulsummonsihaveheardthecockthatisthetrumpettothemorndothwithhisloftya- ndshrillsoundingthroatawakethegodofdayandathiswarningwhetherinseaorfireinearthor- airtheextravagantanderringspirithiestohisconfineandofthetruthhereinthispresentobjec- tmadeprobationmarcellusitfadedonthecrowingofthecocksomesaythatevergainstthatseas- oncomeswhereinoursaviourbirthiscelebratedthebirdofdawningsingethallnightlon- gandthentheysaynospiritdaresstirabroadthenightsarewholesomethennoplanetsstriken- ofairytakesnorwitchhathpowertocharmsohallowdandsograciousisthetimehoratiosoha- veiheardanddoinpartbelieveitbutlookthemorninrussetmantlecladwalksoerthedewofy- onhigheastwardhillbreakweourwatchupandbymyadviceletusimpartwhatwehaveseentо- nightuntoyounghamletforuponmylifethisspiritdumbtouswillspeaktohimdoyouconsen- tweshallacquainthimwithitasneedfulinourlovesfittingourdutymarcellusletsdotipra- yandithismorningknowwhereweshallfindhimmostconveniently’;

Table A1. *Cont.*

```
number = double(text)-96;
histo = zeros(1,26);
for j = 1:length(number)
histo(number (j)) = histo(number (j)) + 1;
end
plot(histo)
% Pattern Generation
counter = zeros(1,26);
summ = sum(counter);
i = 1; j = 1;
while(summ < 26)
sy = number(i);
if (counter(sy) == 0)
counter(sy) = 1;
pattern(j) = sy;
j = j + 1;
end
summ = sum(counter);
i = i + 1;
end
```

Table A2. The proposed HamletPat.

```
function histo = hamlet_pat(sinyal)
h1 = zeros(1512); h2 = zeros(1256); h3 = h1;
for i = 1:length(sinyal)-26
blok = sinyal(i:i + 26);
m = blok(14);
deger(1:13) = blok(1:13);
deger(14:26) = blok(15:27);
for j = 1:26
bit(j) = deger(j) >= m;
end
b1(1:9) = bit(1:9); b2(1:8) = bit(10:17); b3(1:9) = bit(18:26);
m1(i) = 0; m2(i) = 0; m3(i) = 0;
for j = 1:9
m1(i) = m1(i) + b1(j)*2^(j-1);
m3(i) = m3(i) + b3(j)*2^(j-1);
end
for j = 1:8
m2(i) = m2(i) + b2(j)*2^(j-1);
end
h1(m1(i) + 1) = h1(m1(i) + 1) + 1; h2(m2(i) + 1) = h2(m2(i) + 1) + 1;
h3(m3(i) + 1) = h3(m3(i) + 1) + 1;
end
histo = [h1 h2 h3];
```

References

1. Zhou, P.; Yang, X.-L.; Wang, X.-G.; Hu, B.; Zhang, L.; Zhang, W.; Si, H.-R.; Zhu, Y.; Li, B.; Huang, C.-L. A pneumonia outbreak associated with a new coronavirus of probable bat origin. *Nature* **2020**, *579*, 270–273. [CrossRef]
2. Su, S.; Wong, G.; Shi, W.; Liu, J.; Lai, A.C.; Zhou, J.; Liu, W.; Bi, Y.; Gao, G.F. Epidemiology, genetic recombination, and pathogenesis of coronaviruses. *Trends Microbiol.* **2016**, *24*, 490–502. [CrossRef] [PubMed]
3. Khorramdelazad, H.; Kazemi, M.H.; Najafi, A.; Keykhaee, M.; Emameh, R.Z.; Falak, R. Immunopathological similarities between COVID-19 and influenza: Investigating the consequences of Co-infection. *Microb. Pathog.* **2021**, *152*, 104554.
4. Lu, R.; Zhao, X.; Li, J.; Niu, P.; Yang, B.; Wu, H.; Wang, W.; Song, H.; Huang, B.; Zhu, N. Genomic characterisation and epidemiology of 2019 novel coronavirus: Implications for virus origins and receptor binding. *Lancet* **2020**, *395*, 565–574. [CrossRef] [PubMed]
5. Wu, A.; Peng, Y.; Huang, B.; Ding, X.; Wang, X.; Niu, P.; Meng, J.; Zhu, Z.; Zhang, Z.; Wang, J. Genome composition and divergence of the novel coronavirus (2019-nCoV) originating in China. *Cell Host Microbe* **2020**, *27*, 325–328. [CrossRef]

6. Ge, X.-Y.; Li, J.-L.; Yang, X.-L.; Chmura, A.A.; Zhu, G.; Epstein, J.H.; Mazet, J.K.; Hu, B.; Zhang, W.; Peng, C. Isolation and characterization of a bat SARS-like coronavirus that uses the ACE2 receptor. *Nature* **2013**, *503*, 535–538. [CrossRef]
7. He, X.; Yang, X.; Zhang, S.; Zhao, J.; Zhang, Y.; Xing, E.; Xie, P. Sample-efficient deep learning for COVID-19 diagnosis based on CT scans. *medRxiv* **2006**. [CrossRef]
8. Li, F. Structure, function, and evolution of coronavirus spike proteins. *Annu. Rev. Virol.* **2016**, *3*, 237–261. [CrossRef]
9. Afify, H.M.; Zanaty, M.S. A Comparative Study of Protein Sequences Classification-Based Machine Learning Methods for COVID-19 Virus against HIV-1. *Appl. Artif. Intell.* **2021**, *35*, 1733–1745. [CrossRef]
10. Long, J.S.; Mistry, B.; Haslam, S.M.; Barclay, W.S. Host and viral determinants of influenza A virus species specificity. *Nat. Rev. Microbiol.* **2019**, *17*, 67–81. [CrossRef]
11. Vasin, A.; Temkina, O.; Egorov, V.; Klotchenko, S.; Plotnikova, M.; Kiselev, O. Molecular mechanisms enhancing the proteome of influenza A viruses: An overview of recently discovered proteins. *Virus Res.* **2014**, *185*, 53–63. [CrossRef] [PubMed]
12. Kumlin, U.; Olofsson, S.; Dimock, K.; Arnberg, N. Sialic acid tissue distribution and influenza virus tropism. *Influenza Other Respir. Viruses* **2008**, *2*, 147–154. [CrossRef] [PubMed]
13. Robson, B. Bioinformatics studies on a function of the SARS-CoV-2 spike glycoprotein as the binding of host sialic acid glycans. *Comput. Biol. Med.* **2020**, *122*, 103849. [CrossRef] [PubMed]
14. Jones, T.C.; Mühlemann, B.; Veith, T.; Biele, G.; Zuchowski, M.; Hofmann, J.; Stein, A.; Edelmann, A.; Corman, V.M.; Drosten, C. An analysis of SARS-CoV-2 viral load by patient age. *medRxiv* **2012**. [CrossRef]
15. Li, D.; Wang, D.; Dong, J.; Wang, N.; Huang, H.; Xu, H.; Xia, C. False-negative results of real-time reverse-transcriptase polymerase chain reaction for severe acute respiratory syndrome coronavirus 2: Role of deep-learning-based CT diagnosis and insights from two cases. *Korean J. Radiol.* **2020**, *21*, 505–508. [CrossRef]
16. Baygin, M.; Yaman, O.; Barua, P.D.; Dogan, S.; Tuncer, T.; Acharya, U.R. Exemplar Darknet19 feature generation technique for automated kidney stone detection with coronal CT images. *Artif. Intell. Med.* **2022**, *127*, 102274.
17. Barua, P.D.; Dogan, S.; Tuncer, T.; Baygin, M.; Acharya, U.R. Novel automated PD detection system using aspirin pattern with EEG signals. *Comput. Biol. Med.* **2021**, *137*, 104841.
18. Kobat, M.A.; Kivrak, T.; Barua, P.D.; Tuncer, T.; Dogan, S.; Tan, R.-S.; Ciaccio, E.J.; Acharya, U.R. Automated COVID-19 and Heart Failure Detection Using DNA Pattern Technique with Cough Sounds. *Diagnostics* **2021**, *11*, 1962. [CrossRef]
19. Dong, G.; Liu, H. *Feature Engineering for Machine Learning and Data Analytics*; CRC Press: New York, NY, USA, 2018.
20. Zheng, A.; Casari, A. *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2018.
21. Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987. [CrossRef]
22. NCBI Virus. 2022. Available online: https://www.ncbi.nlm.nih.gov (accessed on 3 January 2022).
23. Shakespeare, W.; Scene, I. Elsinore. A Platform before the Castle. Available online: https://shakespeare.mit.edu/hamlet/hamlet.1.1.html (accessed on 3 January 2022).
24. Baygin, M.; Yaman, O.; Tuncer, T.; Dogan, S.; Barua, P.D.; Acharya, U.R. Automated accurate schizophrenia detection system using Collatz pattern technique with EEG signals. *Biomed. Signal Process. Control* **2021**, *70*, 102936. [CrossRef]
25. Vapnik, V. The support vector method of function estimation. In *Nonlinear Modeling*; Springer: New York, NY, USA, 1998; pp. 55–85.
26. Vapnik, V. *The Nature of Statistical Learning Theory*; Springer Science & Business Media: New York, NY, USA, 2013.
27. Liu, H.; Setiono, R. Chi2: Feature selection and discretization of numeric attributes. In Proceedings of the 7th IEEE International Conference on Tools with Artificial Intelligence, Herndon, VA, USA, 5–8 November 1995; pp. 388–391.
28. Powers, D.M. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv* **2020**, arXiv:2010.16061.
29. Warrens, M.J. On the equivalence of Cohen's kappa and the Hubert-Arabie adjusted Rand index. *J. Classif.* **2008**, *25*, 177–183. [CrossRef]
30. Taubenberger, J.K.; Kash, J.C.; Morens, D.M. The 1918 influenza pandemic: 100 years of questions answered and unanswered. *Sci. Transl. Med.* **2019**, *11*, eaau5485. [CrossRef] [PubMed]
31. Jester, B.; Uyeki, T.; Jernigan, D. Readiness for responding to a severe pandemic 100 years after 1918. *Am. J. Epidemiol.* **2018**, *187*, 2596. [CrossRef] [PubMed]
32. Solomon, D.A.; Sherman, A.C.; Kanjilal, S. Influenza in the COVID-19 Era. *JAMA* **2020**, *324*, 1342–1343. [CrossRef] [PubMed]
33. Islam, M.M.; Iqbal, T. Hamlet: A hierarchical multimodal attention-based human activity recognition algorithm. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 25–29 October 2020; pp. 10285–10292.
34. Ren, F.; Zhang, Z.; Yan, Y.; Wang, Z.; Su, S.; Philip, S.Y. HAMLET: Hierarchical Attention-based Model with muLti-task sElf-Training for user profiling. In Proceedings of the 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 15–18 December 2021; pp. 500–509.
35. Loh, H.W.; Ooi, C.P.; Seoni, S.; Barua, P.D.; Molinari, F.; Acharya, U.R. Application of Explainable Artificial Intelligence for Healthcare: A Systematic Review of the Last Decade (2011–2022). *Comput. Methods Programs Biomed.* **2022**, *226*, 107161. [CrossRef] [PubMed]

diagnostics

MDPI

Article

Ensemble Learning Based on Hybrid Deep Learning Model for Heart Disease Early Prediction

Ahmed Almulihi [1], Hager Saleh [2,*], Ali Mohamed Hussien [3], Sherif Mostafa [2], Shaker El-Sappagh [4,5], Khaled Alnowaiser [6], Abdelmgeid A. Ali [7] and Moatamad Refaat Hassan [3]

[1] Department of Computer Science, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia
[2] Faculty of Computers and Artificial Intelligence, South Valley University, Hurghada 84511, Egypt
[3] Department of Computer Science, Faculty of Science, Aswan University, Aswan 81528, Egypt
[4] Faculty of Computer Science and Engineering, Galala University, Suez 34511, Egypt
[5] Information Systems Department, Faculty of Computers and Artificial Intelligence, Benha University, Banha 13518, Egypt
[6] College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al Kharj 11942, Saudi Arabia
[7] Faculty of Computers and Information, Minia University, Minia 61519, Egypt
* Correspondence: hager.saleh@fcih.svu.edu.eg

Abstract: Many epidemics have afflicted humanity throughout history, claiming many lives. It has been noted in our time that heart disease is one of the deadliest diseases that humanity has confronted in the contemporary period. The proliferation of poor habits such as smoking, overeating, and lack of physical activity has contributed to the rise in heart disease. The killing feature of heart disease, which has earned it the moniker the "silent killer," is that it frequently has no apparent signs in advance. As a result, research is required to develop a promising model for the early identification of heart disease using simple data and symptoms. The paper's aim is to propose a deep stacking ensemble model to enhance the performance of the prediction of heart disease. The proposed ensemble model integrates two optimized and pre-trained hybrid deep learning models with the Support Vector Machine (SVM) as the meta-learner model. The first hybrid model is Convolutional Neural Network (CNN)-Long Short-Term Memory (LSTM) (CNN-LSTM), which integrates CNN and LSTM. The second hybrid model is CNN-GRU, which integrates CNN with a Gated Recurrent Unit (GRU). Recursive Feature Elimination (RFE) is also used for the feature selection optimization process. The proposed model has been optimized and tested using two different heart disease datasets. The proposed ensemble is compared with five machine learning models including Logistic Regression (LR), Random Forest (RF), K-Nearest Neighbors (K-NN), Decision Tree (DT), Naïve Bayes (NB), and hybrid models. In addition, optimization techniques are used to optimize ML, DL, and the proposed models. The results obtained by the proposed model achieved the highest performance using the full feature set.

Keywords: machine learning; deep learning; ensemble learning; heart disease

Citation: Almulihi, A.; Saleh, H.; Hussien, A.M.; Mostafa, S.; El-Sappagh, S.; Alnowaiser, K.; Ali, A.A.; Refaat Hassan, M. Ensemble Learning Based on Hybrid Deep Learning Model for Heart Disease Early Prediction. *Diagnostics* **2022**, *12*, 3215. https://doi.org/10.3390/diagnostics12123215

Academic Editors: Wan Azani Mustafa and Hiam Alquran

Received: 9 November 2022
Accepted: 14 December 2022
Published: 18 December 2022

1. Introduction

Heart disease is among the most common illnesses that persisted in the past and have increased and spread in our present. The reasons for the increase in its rates are varied, especially in our modern age. Diabetes, hypertension, cholesterol, erratic heartbeat, and many more clinical signs are some biological markers and risk factors that are needed to diagnose heart disease. World Health Organization (WHO) claims that one of the main and highly-ranked causes of death worldwide is heart disease, which can have several forms such as ischemic, hypertensive, and vascular heart disease [1], and it has been shown that cardiovascular illnesses kill 17.9 million patients each year. In addition, unhealthy behavior that results in being overweight, obesity, and hypertension raises the risk of heart

disease [1]. In addition, the heart is one of the essential organs of the human body. It is primarily responsible for the continuity of pumping the blood needed for the work of the rest of the human body. However, it is difficult for the heart to maintain the same efficiency throughout a person's life. The heart is exposed to many problems that can occur because of several different reasons, such as bad health and nutritional habits or aging [2]. Therefore, finding methods and techniques that allow for the early detection or even prediction of potential heart problems has become inevitable. This can help doctors and healthcare organizations to reduce the problems and complications of the disease.

Artificial intelligence (AI) based on machine learning (ML) and deep learning (DL) has conducted key roles in evaluating medical data to assist in illness diagnosis to determine the appropriate treatment. It is used to find patterns automatically from the clinical data and then reason about clinical data to predict the early risk for patients such as heart disease [3], cancer disease [4,5], and COVID-19 [6,7]. Recently, deep learning algorithms such LSTM, GRU, CNN, and hybrid models of these algorithms have played an important role in strengthening and enhancing the level of heart disease prediction using various layers that could collect deeper features [8–11] Recently, authors have used ensemble learning to enhance the performance of these models in the healthcare domain [12]. Ensemble learning combines the decisions of various base classifiers using many techniques such as voting or averaging to improve the final decision [13]. Ensemble algorithms can be categorized into three branches: boosting [14], stacking [15], and bagging [16]. Stacking ensemble is considered as the best technique for building ensemble models because it is based on a meta-learner, which learns from data how to weight the base classifiers and combine them in the best way to optimize the performance of the resulting model. Ensemble stacking optimizes a set of heterogeneous base models and combines their decisions using a meta-learner [15].

In this study, we proposed an optimized ensemble stacking model that merged the two pre-trained hybrid models of CNN-LSTM and CNN-GRU with a meta-learner (SVM) to enhance the performance of heart disease prediction. In addition, Recursive Feature Elimination (RFE) has been used to choose the most informative features from two heart disease datasets. Our contributions can be summarized as follows:

- We proposed two hybrid models with heterogeneous architectures: CNN-LSTM and CNN-GRU were proposed and optimized.
- We proposed a stacking ensemble model that merged the previous pre-trained hybrid models of CNN-LSTM and CNN-GRU. The best meta-learner classifier has been selected based on the experimental results. The SVM algorithm achieved the best results as the meta-classifier to determine the best weights of the base classifiers;
- We compared the proposed model with different ML models using two benchmark heart disease datasets;
- The proposed model significantly outperformed all other models and achieved the best results.

The remainder of the paper is structured as follows: Section 2 discussed heart disease-related works. The section describes the main phases and approaches Section 3 of predicting heart disease. Section 4 describes the results and discussion results. Finally, the paper is concluded in Section 5.

2. Related Work

Machine learning and deep learning have been used to predict heart disease. For example Kavitha M. et al. [17] suggested a hybrid model that combines DT and RF to predict heart disease using the Cleveland dataset. They contrasted the hybrid model's performance with that of DT and RF. Ishaq A. et al. [18] applied different ML algorithms: SVM, DT, LR, NB, Adaptive boosting (AdaBoost), Stochastic Gradient Descent (SGD), RF, Gradient Boosting Machine (GBM), and Extra Tree Classifier (ETC) using the Cleveland heart disease dataset to analyzes the heart failure. The results showed that ETC gave the best performance and outperformed other models. Ansarullah, S. I. et al. [19] used ML algorithms to predict heart disease: NB, RF, DT, K-NN, and SVM. The dataset was gathered

in Kashmir from many heterogeneous data sources (India). The results showed that RF has the best model performance.

Many authors applied feature selection methods with ML and DL models to predict heart disease. For example, Spencer R. et al. [20] used Chi2, ReliefF, symmetrical uncertainty (SU), and PCA feature selection methods to extract the important features from four heart-disease datasets. They applied BayesNet, Logistic, Stochastic Gradient Descent (SGD), and KNN Adaboost to the full and selected features. The result showed that the BayesNet model was recorded as the best performer using the Chi-2 feature selection compared with other models. Bharti R. et al. [21] used the Lasso algorithm to select features from the heart disease dataset. They applied ML and DL models: LR, KNN, SVM, RF, DT, and ANN, respectively. The results showed that ANN has the best performance compared to ML models. Gokulnath C. B. et al. [22] used KNN, MLP, SVM, and J48 for heart disease detection. The datasets were gathered from a variety of sources. The authors applied various feature selection strategies, including the extra tree classifier, gradient boosting classifier, random forest, recursive feature removal, and XG boost classifier. In the study by Amin, M. S. et al. [23], in order to increase the prediction accuracy, the authors proposed a voting hybrid model based on NB and LR. They used k-NN, DT, NB, LR, SVM, Neural Network (NN), and the hybrid model to choose meaningful characteristics from the Cleveland heart disease dataset. The hybrid model was given the best performance compared to other models. Bashir S. et al. [24] used DT, LR, NB, SVM, and RF models with feature extraction methods with the Cleveland heart disease dataset to predict heart disease. The results showed that LR and SVM with feature selection methods had better accuracy than the other models. Javid I. et al. [25] developed model-based GRU and RF (GRU-RF) for heart disease detection. The GRU-RF was compared with RF, GRU, KNN, and DNN algorithms and achieved the best performance. Chae M. et al. [26] proposed a hybrid model, LSTM–GRU, and compared it with DT, RF, LR, LSTM, and GRU to predict heart disease. They used the dataset from Soonchunhyang University Cheonan Hospital in Korea to train and test the models. They improved the performance models based on hyperparameter adjustment, the quantity of primary patient data, and input parameters. The results indicate that when compared to other models, the GRU model outperforms the others. In the study by Narmadha, S. et al. [27], the authors used LSTM and GRU hyperparameter tuning to enhance the performance of the algorithms. The outcomes demonstrated that the GRU provides better accuracy than the LSTM across the board.

The authors have used ensemble models to predict heart disease. For example, Adhikari, B. et al. [28] applied LR, SVM, DT, K-NN, GNB, and ensemble models using a dataset collected from the UCI heart disease dataset. They used the voting and averaging ensemble models built by combining the ML above models. The results showed that the ensemble model was the best performer compared with other models. Javid, I. et al. [29] used RF, SVM, K-NN, LSTM, Hard Voting Ensemble Model, and GRU for heart disease prediction. The results showed that the Hard Voting Ensemble Model recorded higher accuracy compared to other models.

Ghosh P. et al. [30] proposed hybrid models that integrated boosting and bagging with traditional ML models: KNN, DT, and RF. The hybrid models: K-NN Bagging Method (KNNBM), DT-Bagging Method (DTBM), AdaBoost (AB), and Random Forest Bagging Method (RFBM) were applied to heart disease datasets. Relief, Least Absolute Shrinkage, and Selection Operator were the three feature selection approaches they used (LASSO). When compared to other models, the RFBM model showed the best performance.

Previous studies do not use ensemble stacking based on heterogeneous hybrid deep learning models to predict heart disease. In addition, most previous studies have used the Cleveland heart disease database to perform this experiment. In our work, we used a new large heart disease dataset, and we proposed ensemble stacking models based on optimizing different heterogeneous hybrid models: CNN-LSTM and GRU-LSTM.

3. Methodology

In this study, we evaluate three approaches: the classical machine learning approach, the hybrid models approach, and a proposed model. These models are applied to the full feature set and selected feature set. The proposed model for predicting heart disease has several steps including data collection, data preprocessing, data splitting, feature selection, and evaluation models, as shown in Figure 1. Each phase is described in detail as follows.

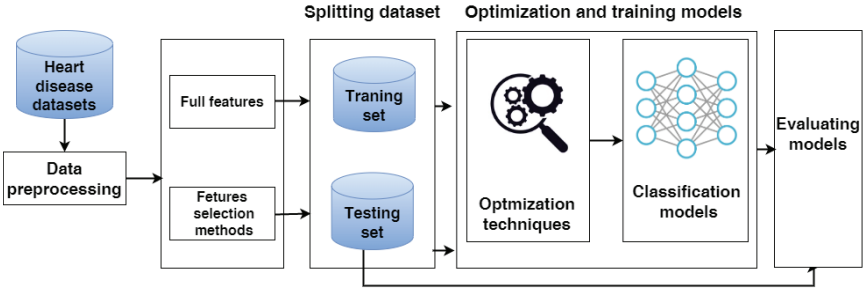


Figure 1. The phases of predicting heart disease.

3.1. Heart Disease Datasets

In our work, we used two heart disease datasets.

3.1.1. Dataset 1

We used the large heart disease dataset (Heart Disease) [31]. This data includes 18 independent features and one dependent variable as the class label for predicting heart disease. The class label includes two values: 0 represents the healthy class label, and 1 represents the heart disease class label. Table 1 presents the number of medical records for each class in the training and testing sets. The description of each feature is described in a Supplementary File.

3.1.2. Cleveland Dataset

The Cleveland dataset [32] includes 13 independent variables as features and one dependent variable as the class label used to diagnose heart disease. The class label includes two values: 0 represents the healthy class label, and 1 represents the heart disease class label. Table 1 presents the number of medical records for each class in the training and testing sets of the Cleveland heart disease dataset. The description of each feature is described in the Supplementary File.

Table 1. The number of medical records for each class in the heart disease datasets.

Dataset	Classes	Training Set	Testing Set	Total
Dataset 1	Heart disease	21,898	5475	27,373
	Healthy	24,000	6000	30,000
	Total	45,898	11,475	57,373
Cleveland Dataset	Heart disease	421	105	526
	Healthy	399	100	499
	Total	820	205	1025

3.2. Data Pre-Processing

The first heart disease dataset includes 14 numeric features and four categorical features. The data was preprocessed after collection as follows: removing duplicate records and encoding category data into numerical data such as smoking and skin cancer.

3.3. Data Splitting

The two datasets are divided into two sets using a stratified sampling method: 80% training sets and 20% testing sets. Models are trained and optimized using training data. The test set is employed to assess and test the model.The stratified sampling method is one way of splitting the dataset used to get samples that accurately reflect the distribution of classes in the population. It separates the dataset into homogeneous subsets; each subset contains the same percentage of every class. [33,34]. This method has been used in studies of different fields of healthcare [35–37]

3.4. Feature Selection Methods

In our work, we use the Recursive Feature Elimination (RFE) feature selection method to extract the most informative features from each dataset. The RFE determines the essential features by figuring a high correlation between features and the target [38]. It assigns one value as ranking for features if the features have high collaboration with the target. A novel RFE strategy is recently presented that used RF and SVM to evaluate features rather than classification performance and selects the minor significant features for deletion [39,40]

3.5. Machine Learning Approach

3.5.1. ML Algorithms

We tested many classical ML models from different families including SVM [41–44], Logistic Regression (LR) [45,46], Nave Bayes (NB) [47], Decision tree (DT) [48], Random Forest (RF) [49,50], and K-nearest Neighbors (k-NN) [51].

3.5.2. Optimization Techniques for Classical Models

Grid search is employed to fine-tune hyperparameters of different classical ML models by generating discrete grids within the hyperparameter domain and select the list of parameters that give the best performance [52]. Data is split into two segments using the cross-validation technique: one is used to train and validate the models (training set), and the other is utilized for model testing (testing set) [19]. The training set has been used to validate the models using the k-fold cross validation technique.

3.6. The Hybrid Models

3.6.1. The Hybrid Model Architectures

We proposed two hybrid models: CNN-LSTM and CNN-GRU for predicting heart disease. The structures of hybrid models are illustrated in Figure 2.

- The first model is CNN-LSTM, which combines CNN with LSTM, consisting of a convolutional layer, a max-pooling layer, an LSTM layer, a flatten layer, a fully connected, and an output layer;
- The second model is CNN-GRU, which combines CNN with GRU. The architecture consists of a convolutional layer, a max-pooling layer, an GRU layer, a flatten layer, a fully connected, and an output layer.

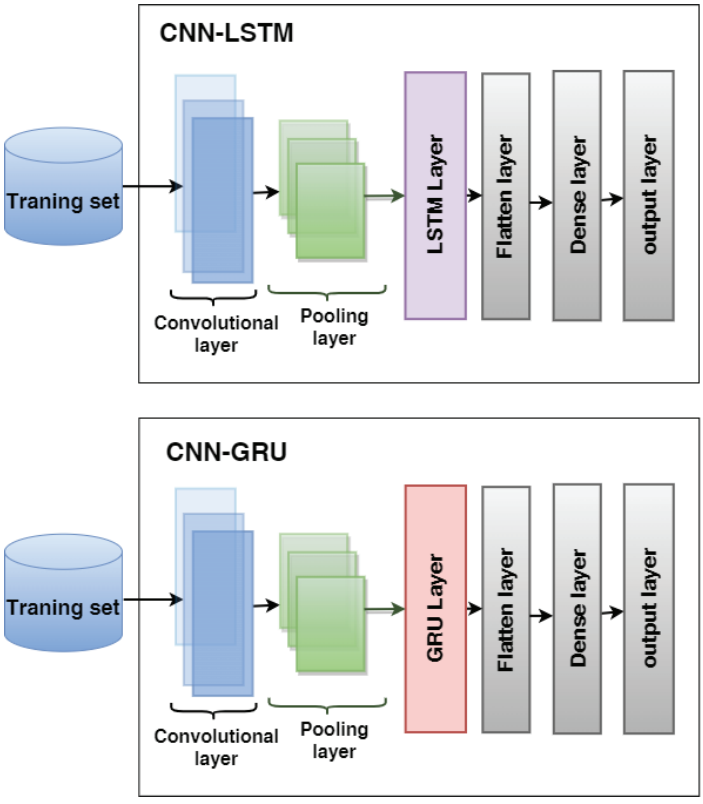


Figure 2. The architecture of the hybrid models CNN-LSTM and CNN-GRU used to predict heart disease.

3.6.2. Optimization Techniques for Hybrid Models

The Bayesian optimizer is used to optimize the hybrid models. This search technique quickly generates the search space and locates the best hyperparameter values for the models [53]. We adopt the parameter settings for CNN-LSTM and CNN-GRU, as shown in Table 2.

Table 2. Setting values of the parameters.

Parameters	Values
filters	[16,128]
Kernel_size	[2,3,4,5]
Pool_Size	[2,3,4,5]
Unit_LSTM	between 20 and 500
Unit_GRU	between 20 and 500
Unit_Dense	between 20 and 500

3.7. The Proposed Stacking Ensemble Model

In this work, our model is developed using two levels: Level-1 and Level-2, as shown in Figure 3. Level-1 begins by loading the pre-trained models of hybrid models CNN-LSTM and CNN-GRU, and the layers of the models are frozen except for the last layers. The models anticipate the training set's output probabilities and subsequently integrate them into stacking training. Secondly, the models estimate the output probabilities of the testing set and aggregate them in stacking testing. At Level 2, SVM, as a meta-learner, is trained

and optimized using stacking training and Grid search, respectively, while producing the final results using stacking testing.

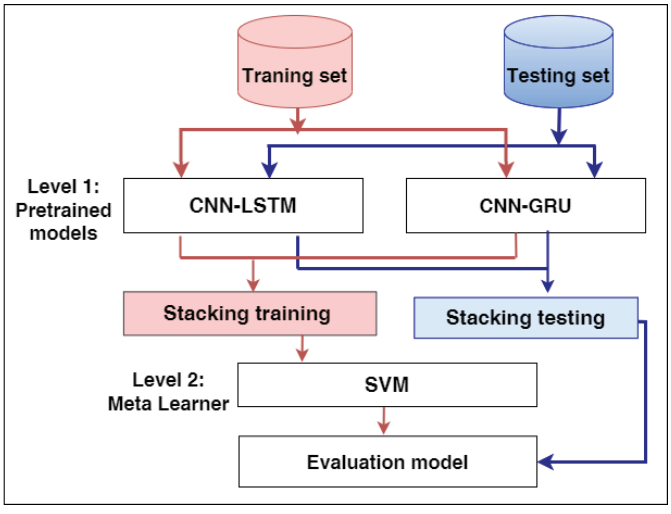


Figure 3. The proposed model for predicting heart disease.

3.8. Evaluating Models

The metrics for classification performance that are most frequently employed are accuracy (ACC), precision (PRE), recall (REC), and *F*1-score (*F*1). In contrast to the True Positive (*TP*), which denotes that the person is ill and the test is positive, the True Negative (*TN*) shows that the person is healthy and the result is negative. False positives are tests that come back positive even when the subject is healthy (*FP*). When a test is negative, but the subject is ill, it is known as a false negative (*FN*).

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}. \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1\text{-}score = \frac{2 \cdot precision \cdot recall}{precision + recall} \quad (4)$$

4. Experimental Results

In this section, we describe the rank of features after applying the RFE to the two datasets. Moreover, we describe the results of the performance of using ML models (SVM, LR, RF, NB, and KNN), the hybrid models (CNN-LSTM, CNN-GRU), and the proposed model to full and selected features.

4.1. Experimental Setup

The experiments in this paper are implemented using Google Colab with Python libraries such as Scikit-learn, TensorFlow, and others. We used grid-search and the Bayesian optimizer to optimize the ML and hybrid models. We used RFE technique to identify the best features from the two datasets. The two datasets are separated into two sets: 80% training and 20% testing set using the stratified methods. The models are trained and tested by utilizing the training and testing sets, respectively.

4.2. Results of Dataset1

4.2.1. Feature Selection Results

In the experiments, we used the RFE to extract the important features from the heart disease dataset by assigning ranking for every feature. The critical features are ranked 1, and the least important features are ranked 8. The features ranking is shown in Figure 4. We can see that the most significant 10 features have a ranking of 1: BMI, Stroke, PhysicalHealth, MentalHealth, DiffWalking, AgeCategory, Race, Diabetic, GenHealth, and SleepTime. The lowest important feature has a ranking of 8, which is AlcoholDrinking.

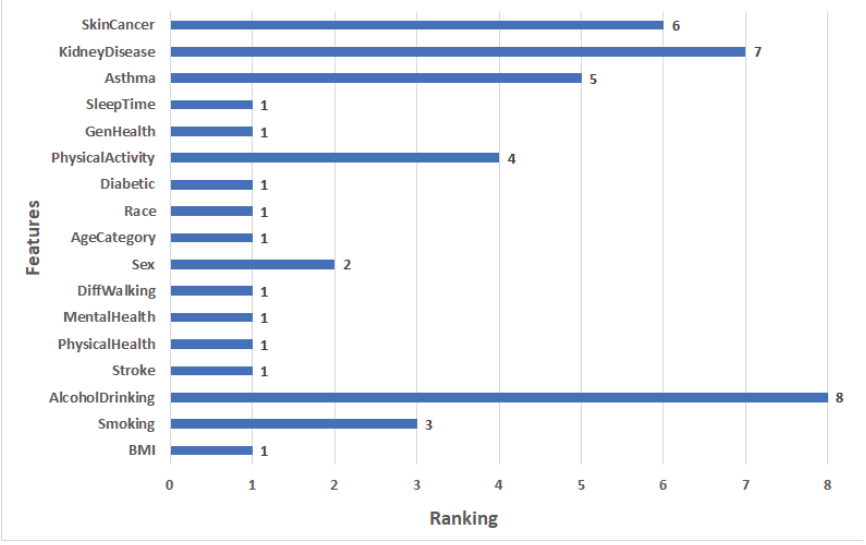


Figure 4. The ranking features for heart disease Dataset 1.

4.2.2. Results of Applying Models

This section presents the ACC, PRE, REC, and F1 of ML, hybrid models, and the proposed model for Dataset 1. In the hybrid models CNN-LSTM and CNN-GRU some parameters were adapted: batch_size of 500, epoch = 50, learning rate = 0.00004, and the optimizer used is Adam. Some of the best values of CNN-LSTM and CNN-GRU hyperparameters that were selected by KerasTuner are shown in Table 3.

Table 3. The best values of the parameters for CNN-LSTM and CNN-GRU.

Dataset	Models	Parameters	Full Features	Selected Features
Dataset 1	CNN-LSTM	filters	128	16
		Kernel_size	4	4
		Pool_Size	2	2
		Unit_LSTM	380	40
		Unit_Dense	140	50
	CNN-GRU	filters	128	16
		Kernel_size	4	4
		Pool_Size	2	2
		Unit_GRU	100	320
		Unit_Dense	100	200

Table 4 shows the results of applying ML, hybrid models, and the proposed model with full features and selected features by RFE to the heart disease Dataset 1.

- Results of the full features:
  For ML models, RF and LR register approximately the same highest scores (75.32% of ACC, 75.44% of PRE, 75.32% of REC, 75.33% of F1) and (75.60% of ACC, 75.60% of PRE, 75.60% of REC, 75.60% of F1), respectively. NB records the worst scores (60.87% of ACC, 64.98% of PRE, 60.87% of REC, 56.69% of F1). KNN registers the second-highest scores (73.16% of ACC, 73.47% of PRE, 73.16% of REC, 73.16% of F1).
  For hybrid models, CNN-LSTM has the highest scores (76.64% of ACC, 76.9% of PRE, 76.64% of REC, and 76.65% of F1). CNN-GRU records the lowest scores (75.63% of ACC, 75.65% of PRE, 75.63% of REC, 75.58% of F1).
  The proposed model records the highest scores (ACC = 78.81%, 78.1% of PRE, 78.81% of REC, and 78.81% of F1) compared to other models. It improves ACC by 2.17, PRE by 1.2, REC by 2.17, and F1 by 2.16 compared to CNN-LSTM.
- Results of the selected features:
  For ML models, RF and LR register approximately the same highest scores (73.02% of ACC, 73.06% of PRE, 73.02% of REC, 73.03% of F1) and (73.58% of ACC, 73.60% of PRE, 73.58% of REC, = 73.59% of F1), respectively. NB records the worst scores (60.84% of ACC, 64.97% of PRE, 60.84% of REC, F1 = 56.63%). KNN registers the second-highest scores (72.59% of ACC, 72.92% of PRE, 72.59% of REC, F1 = 72.59%).
  The top scores for hybrid models belong to CNN-LSTM (75.22% of ACC, 75.42% of PRE, 75.22% of REC, and 75.22% of F1). The lowest scores are recorded by CNN-GRU (74.07% of ACC, 74.23% of PRE, 74.07% of REC, and 74.08% of F1).
  In comparison to other models, the proposed model achieves the greatest scores (77.42% of ACC, 77.99% of PRE, 77.42% of REC, and 77.39% of F1). In comparison to CNN-LSTM, it enhances ACC by 2.2%, PRE by 2.57%, REC by 2.2%, and F1 by 2.17%.

Table 4. Result of applying models with full features and the selected features for Dataset 1.

Approaches	Models	Features	Matrix Performance			
			ACC	PRE	REC	F1
Regular ML approach	RF	Full features	75.32	75.44	75.32	75.33
		Selected features	73.02	73.06	73.02	73.03
	LR	Full features	75.60	75.60	75.60	75.60
		Selected features	73.58	73.60	73.58	73.59
	DT	Full features	67.28	67.26	67.28	67.27
		Selected features	65.76	65.76	65.76	65.7
	NB	Full features	60.87	64.98	60.87	56.69
		Selected features	60.84	64.97	60.84	56.63
	KNN	Full features	73.16	73.47	73.16	73.16
		Selected features	72.59	72.92	72.59	72.59
The hybrid models	CNN-LSTM	Full features	76.64	76.9	76.64	76.65
		Selected features	75.22	75.42	75.22	75.22
	CNN-GRU	Full features	75.63	75.65	75.63	75.58
		Selected features	74.07	74.23	74.07	74.08
The proposed model	Stacking SVM	Full features	78.81	78.1	78.81	78.81
		Selected features	77.42	77.99	77.42	77.39

4.3. Results of the Cleveland Dataset

4.3.1. Feature Selection Results

In the experiments, we used the RFE to extract the important features from the Cleveland dataset. It assigns features a value of ranking, with the critical features having a ranking of 1, and the least important features having a ranking of 8. The features ranking is shown in Figure 5. We can see that the 8 most significant features have a ranking of 1: age, cp, thalach, oldpeak, ca, and thal. The least important feature has a ranking of 8, which is fbs.

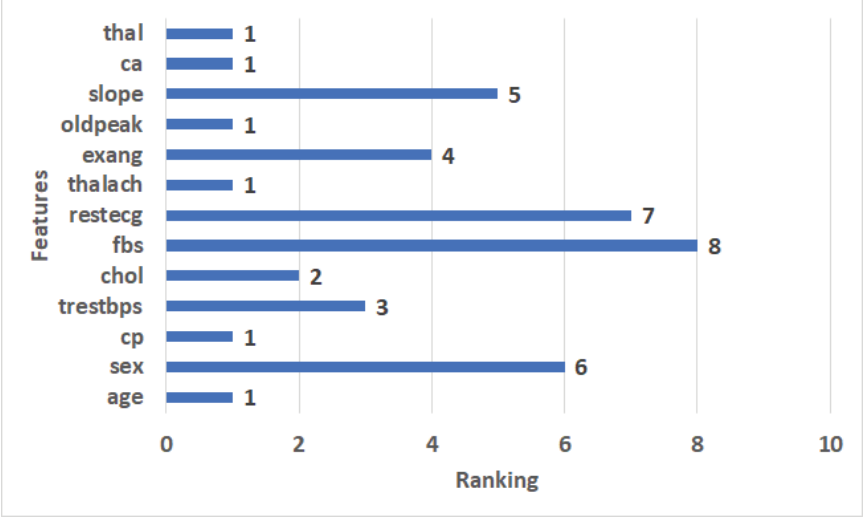


Figure 5. The ranking features for the Cleveland dataset.

4.3.2. Results of the Applied Models

This section presents the setting of values parameters for models and the results of applied ML, hybrid models, and the proposed model with the full and selected features for the Cleveland dataset. The following settings were modified for CNN-LSTM and CNN-GRU hybrid models: batch size = 50, epoch = 50, learning rate = 0.00004, and the optimizer used is Adam. Some of the best CNN-LSTM and CNN-GRU hyperparameter values as determined by KerasTuner are shown in Table 5.

Table 6 shows the results of applying ML, hybrid models, and the proposed model with full features and selected features by RFE to the Cleveland dataset.

- Full features
  For ML models, RF has the highest scores (86.34% of ACC, 86.34% of PRE, 86.34% of REC, and 86.34% of F1). NB records the lowest scores (60.00% of ACC, 60.05% of PRE, 60.00% of REC, 59.74% of F1). DT registers the second-highest scores (82.44% of ACC, 82.46% of PRE, 82.44% of REC, 82.44% of F1).
  For hybrid models, CNN-LSTM has the highest scores (89.76% of ACC, 89.96% of PRE, REC = 89.76% of REC, F1 = 89.75%). CNN-GRU records the lowest scores (88.29% of ACC, 89.06% of PRE, REC = 88.29% of REC, 88.26% of F1).
  The proposed model records the highest scores (97.17% of ACC, 97.42% of PRE, 97.17% of REC, 97.15% of F1) compared to the other models. It improves ACC by 7.41, PRE by 7.46, REC by 7.41, and F1 by 7.4 compared to CNN-LSTM.
- Selected features
  For ML models, RF has the highest scores (82.93% of ACC, 82.99% of PRE, 82.93% of REC, 82.91% of F1). NB records the lowest scores (64.88% of ACC, 64.90% of PRE, 64.88% of REC, 64.88% of F1). DT registers the second-highest scores (81.95% of ACC,

PRE = 82.01%, 81.95% of REC, 81.93% of F1).
For hybrid models, CNN-LSTM has the highest scores (86.34% of ACC, 86.41% of PRE, 86.34% of REC, and 86.34% of F1). CNN-GRU records the lowest scores (85.85% of ACC, 86.92% of PRE, 85.85% of REC, 85.78% of F1).
The proposed model records the highest scores (91.22% of ACC, 91.29% of PRE, 91.22% of REC, 91.22% of F1) compared to other models. It improves ACC by 4.88, PRE by 4.88, REC by 4.88, and F1 by 4.88 compared to CNN-LSTM.

Table 5. The best values of the parameters for the Cleveland dataset.

Datasets	Models	Parameters	Full Features	Selected Features
Cleveland dataset	CNN-LSTM	filters	128	16
		Kernel_size	4	5
		Pool_Size	2	2
		Unit_LSTM	360	60
		Dense Unit	160	20
	CNN-GRU	filters	64	16
		Kernel_size	4	5
		Pool_Size	2	2
		Unit_GRU	440	80
		Unit_Dense	160	40

Table 6. Result of applying models with full features and selected features for the Cleveland dataset.

Approaches	Models	Features	Matrix Performance			
			ACC	PRE	REC	F1
Regular ML approach	RF	Full features	86.34	86.34	86.34	86.34
		Selected features	82.93	82.99	82.93	82.91
	LR	Full features	67.32	67.43	67.3	67.18
		Selected features	73.17	73.19	73.17	73.14
	DT	Full features	82.44	82.46	82.44	82.44
		Selected features	81.95	82.01	81.95	81.93
	NB	Full features	60.00	60.05	60.00	59.74
		Selected features	64.88	64.90	64.88	64.88
	KNN	Full features	60.00	60.25	60.00	59.92
		Selected features	66.34	66.62	66.34	66.29
The hybrid models	CNN-LSTM	Full features	89.76	89.96	89.76	89.75
		Selected features	86.34	86.41	86.34	86.34
	CNN-GRU	Full features	88.29	89.06	88.29	88.26
		Selected features	85.85	86.92	85.85	85.78
The proposed model	Stacking SVM	Full features	97.17	97.42	97.17	97.15
		Selected features	91.22	91.29	91.22	91.22

4.4. Discussion

We used two heart disease datasets downloaded from Kaggle. We applied RFE feature selection methods to select the essential features. The proposed model, in all cases, has achieved the highest score compared with the other models.

4.4.1. Dataset1

Figures 6 and 7 show the best models for applying models with full features and selected features. We can see that the proposed model has achieved the highest scores with full features at ACC = 78.81%, PRE = 78.81%, REC = 78.81%, and F1 = 78.81% compared to other models with full features and selected features, and It improves ACC by 2.17, PRE by 1.2, REC by 2.17, and F1 by 2.16 compared to CNN-LSTM. In addition, it has the highest scores with selected features at (ACC = 77.42%, PRE = 77.99%, REC = 77.42%, F1 = 77.39%, and it improves ACC by 2.2%, PRE by 2.57%, REC by 2.2%, and F1 by 2.17%. LR has the lowest scores with full features and selected features.

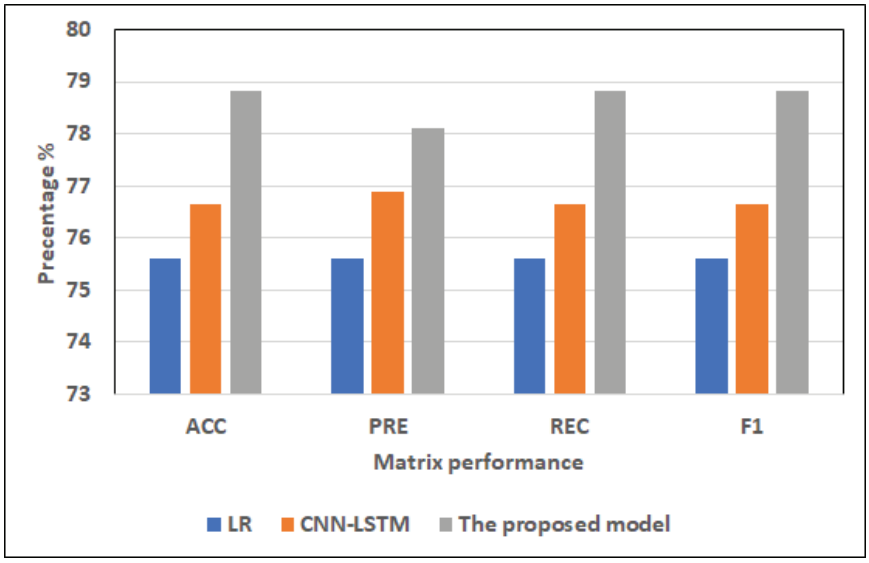


Figure 6. The best models for applying models with full features for Dataset 1.

4.4.2. Cleveland Dataset

Figures 8 and 9 show the best models for applying models with full features and selected features. We can see that the proposed model has achieved the highest scores with full features at ACC = 98.17%, PRE = 98.42%, REC = 98.17%, and F1 = 98.15% compared to other models with full features and selected features, and it improves ACC by 3.41, PRE by 3.46, REC by 3.41, and F1 by 3.4 compared to CNN-LSTM. In addition, it has the highest scores with selected features at (ACC = 91.22%, PRE = 91.29%, REC = 91.22%, F1 = 91.22%, and it improves ACC by 4.88, PRE by 4.88, REC by 4.88 and F1 by 4.88 compared to CNN-LSTM. RF has the lowest scores with full features, and LR has the lowest scores with the selected features.

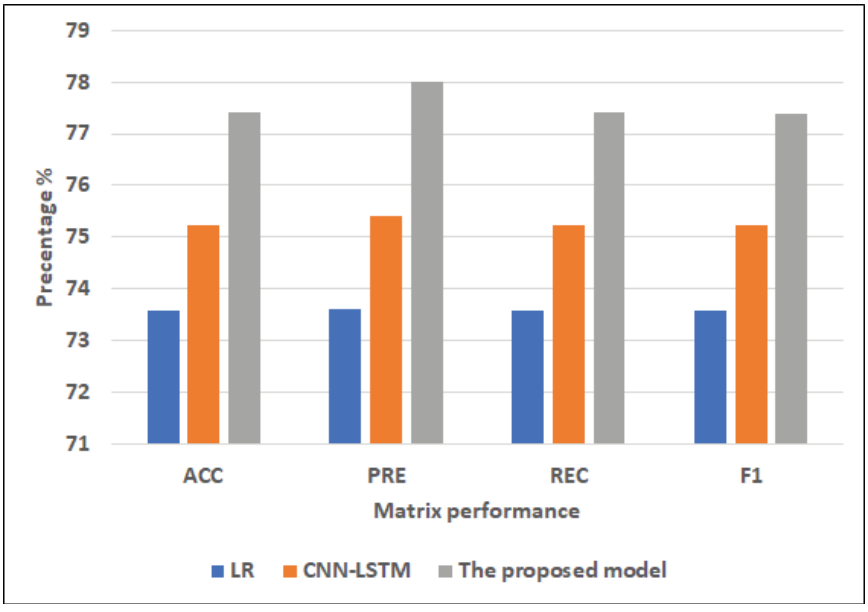


Figure 7. The best models for applying models with selected features for Dataset 1.

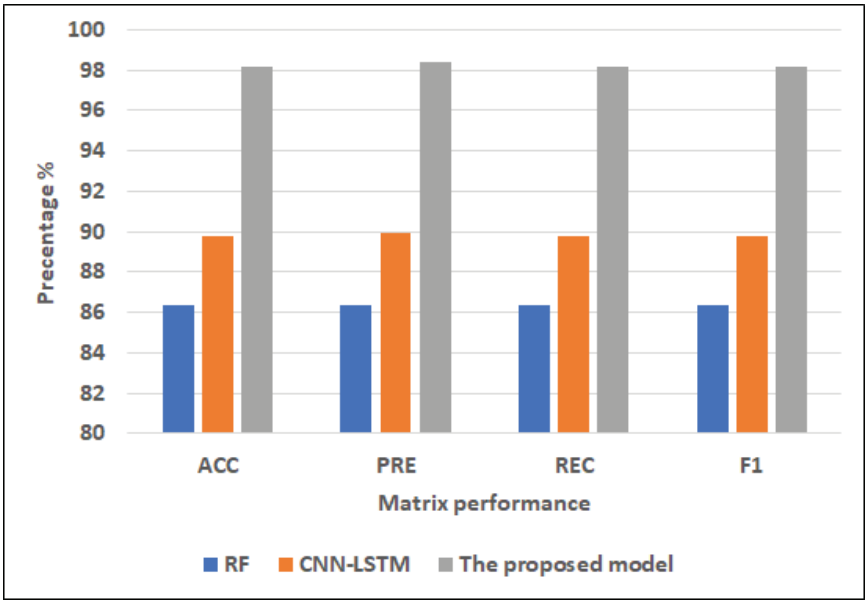


Figure 8. The best models for applying models with full features for Dataset 2.

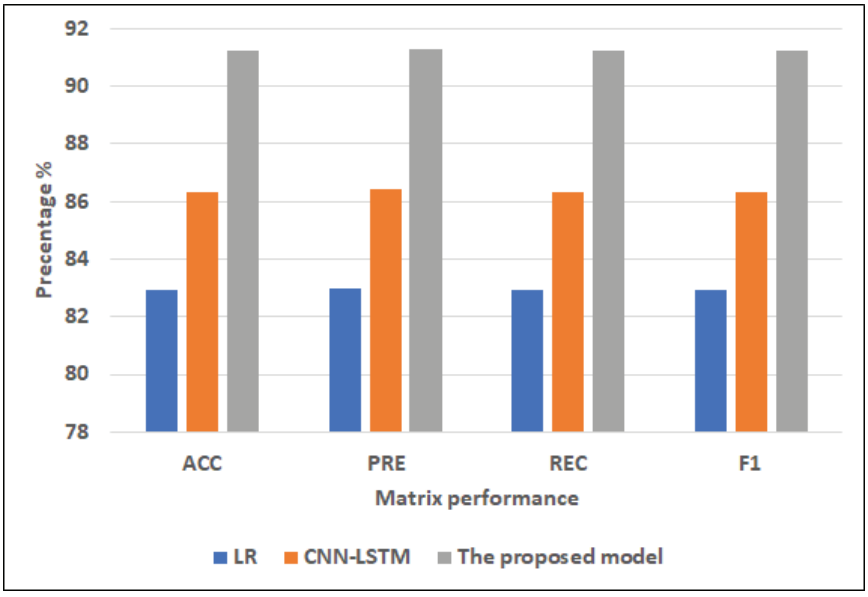


Figure 9. The best models for applying models with selected features for Dataset 2.

4.4.3. Comparison with Literature Studies

By assessing the developed model against the current models we could observe that our approach enhanced the scores more than other models. We compared our approach with the approach by authors who used the Cleveland Dataset, as shown in Table 7. The authors of Ref. [17] used a hybrid model combining DT and RF, which recorded 88.7% of ACC. The authors in Refs. [20,22–24,29], used various models, none of which were accurate to more than 90%, which recorded 85%, 88.34%, 87.41%, 84.85%, and 85.71%, respectively. While in Ref. [18,21,28], the authors achieved an accuracy of over 90%. The proposed model has achieved the highest ACC at 98.41% compared to the ACC values in these studies.

Table 7. Comparison between previous studies and the proposed model for the Cleveland dataset.

Papers	Models	Datasets	Accuracy
[17]	hybrid model that combines DT and RF	Cleveland Dataset.	88.7%
[18]	DT, AdaBoost, LR, SGD, RF, GBM, ETC, GNB, SVM	Cleveland Dataset.	92.62%
[20]	BayesNet, LR, SGD, IBK(k = 21), AdaB(DS), AdaB(Logistic), RF	Cleveland Dataset.	85%
[21]	LR, KNN, SVM, RF, DT, DL	Cleveland Dataset.	94.2%
[22]	KNN, MLP, SVM, and J48	Cleveland Dataset.	88.34%
[23]	K-NN, DT, NB, LR, SVM, NN, Vote	Cleveland Dataset.	87.41%
[24]	DT, LR, RF, NB, LR (SVM)	Cleveland Dataset.	84.85%
[28]	Ensemble Voting,	Cleveland Dataset.	96.43%

Table 7. *Cont.*

Papers	Models	Datasets	Accuracy
[29]	Hard Voting Ensemble Model	Cleveland Dataset.	85.71%
Our work	The proposed model	Cleveland Dataset.	98.41

5. Conclusions

The study proposed a deep staking ensemble to improve the performance of heart disease prediction. The proposed model was based on the integration of two pre-trained and optimized deep hybrid models: CNN-LSTM and CNN-GRU. The SVM classifier has been used as the meta-learner model. The first hybrid model was the CNN-LSTM model, which combined CNN and LSTM layers. The second hybrid model was the CNN-GRU model, which combined CNN with GRU models. RFE was used to choose the most important features from two heart disease datasets. The proposed models were compared with five classical ML models, including LR, RF, K-NN, DT, NB, and hybrid models (i.e., CNN-LSTM and CNN-GRU). Results were collected with the full feature set and a selected feature set. Compared to other models, the result generated by the proposed model had the optimum performance with all the features. For the first dataset, the proposed model had the highest ACC of 78.81%, PRE of 78.1%, REC of 78.81%, and F1 of 78.81. For the Cleveland dataset, the proposed model had the highest ACC of 97.17%, PRE of 97.42%, REC of 97.17%, and F1 of 97.15%. In addition, the proposed model achieved better results than the literature. As a result, the proposed model can improve the disease prediction and can improve the quality of life of the heart disease patients. In the future, we will test the performance of the proposed model with other datasets. We will extend the model by adding other modalities such as images and EEG data. We will provide interpretability features to the proposed model.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/diagnostics12123215/s1, Table S1: Features information and description of heart disease dataset1; Table S2: Features information and description of Cleveland heart disease dataset 2016.

Author Contributions: Methodology, H.S.; Software, H.S.; Data curation, A.M.H.; Writing—original draft, H.S., S.M. and S.E.-S.; Writing—review & editing, A.A., H.S., A.M.H., S.M., S.E.-S., K.A., A.A.A. and M.R.H.; Visualization, H.S. and S.E.-S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The direct link in the dataset citations will take you to all of the datasets that were utilized to support the study's assertions.

Acknowledgments: We deeply acknowledge Taif University for Supporting This research through Taif University Researchers Supporting Project number (TURSP-2020/344), Taif University, Taif, Saudi Arabia.

Conflicts of Interest: All authors declare that they have no conflict of interest.

References

1. Cardiovascular Diseases (CVDs). Available online: http://www.who.int/cardiovascular_diseases/en/ (accessed on 10 October 2022).
2. Hall, J.E.; Hall, M.E. *Guyton and Hall Textbook of Medical Physiology e-Book*; Elsevier Health Sciences: Amsterdam, The Netherlands, 2020.
3. Bhowmick, A.; Mahato, K.D.; Azad, C.; Kumar, U. Heart Disease Prediction Using Different Machine Learning Algorithms. In Proceedings of the 2022 IEEE World Conference on Applied Intelligence and Computing (AIC), Sonbhadra, India, 17–19 June 2022; pp. 60–65.
4. Saleh, H.; Alyami, H.; Alosaimi, W. Predicting Breast Cancer Based on Optimized Deep Learning Approach. *Comput. Intell. Neurosci.* **2022**, *2022*, 1820777. [CrossRef] [PubMed]

5. Cardoso, M.R.; Santos, J.C.; Ribeiro, M.L.; Talarico, M.C.R.; Viana, L.R.; Derchain, S.F.M. A metabolomic approach to predict breast cancer behavior and chemotherapy response. *Int. J. Mol. Sci.* **2018**, *19*, 617. [CrossRef] [PubMed]
6. Spagnuolo, G.; De Vito, D.; Rengo, S.; Tatullo, M. COVID-19 outbreak: An overview on dentistry. *Int. J. Environ. Res. Public Health* **2020**, *17*, 2094. [CrossRef] [PubMed]
7. Alouffi, B.; Alharbi, A.; Sahal, R.; Saleh, H. An Optimized Hybrid Deep Learning Model to Detect COVID-19 Misleading Information. *Comput. Intell. Neurosci.* **2021**, *2021*, 9615034. [CrossRef]
8. Mitchell, T.; Buchanan, B.; DeJong, G.; Dietterich, T.; Rosenbloom, P.; Waibel, A. Machine learning. *Annu. Rev. Comput. Sci.* **1990**, *4*, 417–433. [CrossRef]
9. Chan, S.R.; Torous, J.; Hinton, L.; Yellowlees, P. Mobile tele-mental health: Increasing applications and a move to hybrid models of care. *Healthcare* **2014**, *2*, 220–233. [CrossRef]
10. Sharma, S.; Parmar, M. Heart diseases prediction using deep learning neural network model. *Int. J. Innov. Technol. Explor. Eng. (IJITEE)* **2020**, *9*, 124–137. [CrossRef]
11. Weissler, E.H.; Naumann, T.; Andersson, T.; Ranganath, R.; Elemento, O.; Luo, Y.; Freitag, D.F.; Benoit, J.; Hughes, M.C.; Khan, F.; et al. The role of machine learning in clinical research: Transforming the future of evidence generation. *Trials* **2021**, *22*, 1–15.
12. Melin, P.; Monica, J.C.; Sanchez, D.; Castillo, O. Multiple ensemble neural network models with fuzzy response aggregation for predicting COVID-19 time series: The case of Mexico. *Healthcare* **2020**, *8*, 181. [CrossRef]
13. Sagi, O.; Rokach, L. Ensemble learning: A survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2018**, *8*, e1249. [CrossRef]
14. Freund, Y.; Schapire, R.E. Experiments with a new boosting algorithm. *ICML Citeseer* **1996**, *6*, 148–156.
15. Rajagopal, S.; Kundapur, P.P.; Hareesha, K.S. A stacking ensemble for network intrusion detection using heterogeneous datasets. *Secur. Commun. Netw.* **2020**, *2020*, 4586875. [CrossRef]
16. Bühlmann, P. Bagging, boosting and ensemble methods. In *Handbook of Computational Statistics*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 985–1022.
17. Kavitha, M.; Gnaneswar, G.; Dinesh, R.; Sai, Y.R.; Suraj, R.S. Heart disease prediction using hybrid machine learning model. In Proceedings of the 2021 6th International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 20–22 January 2021; pp. 1329–1333.
18. Ishaq, A.; Sadiq, S.; Umer, M.; Ullah, S.; Mirjalili, S.; Rupapara, V.; Nappi, M. Improving the prediction of heart failure patients' survival using SMOTE and effective data mining techniques. *IEEE Access* **2021**, *9*, 39707–39716. [CrossRef]
19. Ansarullah, S.I.; Saif, S.M.; Kumar, P.; Kirmani, M.M. Significance of visible non-invasive risk attributes for the initial prediction of heart disease using different machine learning techniques. *Comput. Intell. Neurosci.* **2022**, *2022*, 9580896. [CrossRef]
20. Spencer, R.; Thabtah, F.; Abdelhamid, N.; Thompson, M. Exploring feature selection and classification methods for predicting heart disease. *Digit. Health* **2020**, *6*, 2055207620914777. [CrossRef]
21. Bharti, R.; Khamparia, A.; Shabaz, M.; Dhiman, G.; Pande, S.; Singh, P. Prediction of heart disease using a combination of machine learning and deep learning. *Comput. Intell. Neurosci.* **2021**, *2021*, 8387680. [CrossRef]
22. Gokulnath, C.B.; Shantharajah, S. An optimized feature selection based on genetic approach and support vector machine for heart disease. *Clust. Comput.* **2019**, *22*, 14777–14787. [CrossRef]
23. Amin, M.S.; Chiam, Y.K.; Varathan, K.D. Identification of significant features and data mining techniques in predicting heart disease. *Telemat. Informatics* **2019**, *36*, 82–93. [CrossRef]
24. Bashir, S.; Khan, Z.S.; Khan, F.H.; Anjum, A.; Bashir, K. Improving heart disease prediction using feature selection approaches. In Proceedings of the 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, 8–12 January 2019; pp. 619–623.
25. Javid, I.; Ghazali, R.; Zulqarnain, M.; Husaini, N.A. Deep Learning GRU Model and Random Forest for Screening Out Key Attributes of Cardiovascular Disease. In *International Conference on Soft Computing and Data Mining*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 160–170.
26. Chae, M.; Gil, H.W.; Cho, N.J.; Lee, H. Machine Learning-Based Cardiac Arrest Prediction for Early Warning System. *Mathematics* **2022**, *10*, 2049. [CrossRef]
27. Narmadha, S.; Gokulan, S.; Pavithra, M.; Rajmohan, R.; Ananthkumar, T. Determination of various deep learning parameters to predict heart disease for diabetes patients. In Proceedings of the 2020 International Conference on System, Computation, Automation and Networking (ICSCAN), Pondicherry, India, 3–4 July 2020; pp. 1–6.
28. Adhikari, B.; Shakya, S. Heart Disease Prediction Using Ensemble Model. In *Proceedings of Second International Conference on Sustainable Expert Systems*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 857–868.
29. Javid, I.; Alsaedi, A.K.Z.; Ghazali, R. Enhanced accuracy of heart disease prediction using machine learning and recurrent neural networks ensemble majority voting method. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*. [CrossRef]
30. Ghosh, P.; Azam, S.; Jonkman, M.; Karim, A.; Shamrat, F.J.M.; Ignatious, E.; Shultana, S.; Beeravolu, A.R.; De Boer, F. Efficient prediction of cardiovascular disease using machine learning algorithms with relief and LASSO feature selection techniques. *IEEE Access* **2021**, *9*, 19304–19326. [CrossRef]
31. Heart Disease Prediction. Available online: https://www.kaggle.com/code/andls555/heart-disease-prediction/data/ (accessed on 8 November 2022).
32. Heart Disease Dataset. Available online: https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset (accessed on 8 November 2022).

33. Sechidis, K.; Tsoumakas, G.; Vlahavas, I. On the stratification of multi-label data. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 145–158.
34. Liberty, E.; Lang, K.; Shmakov, K. Stratified sampling meets machine learning. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016; pp. 2320–2329.
35. Naz, H.; Ahuja, S. Deep learning approach for diabetes prediction using PIMA Indian dataset. *J. Diabetes Metab. Disord.* **2020**, *19*, 391–403. [CrossRef] [PubMed]
36. Prusty, S.; Patnaik, S.; Dash, S.K. SKCV: Stratified K-fold cross-validation on ML classifiers for predicting cervical cancer. *Front. Nanotechnol.* **2022**, *4*, 972421. [CrossRef]
37. Fonarow, G.C.; Adams, K.F.; Abraham, W.T.; Yancy, C.W.; Boscardin, W.J.; Committee, A.S.A. Risk stratification for in-hospital mortality in acutely decompensated heart failure: Classification and regression tree analysis. *JAMA* **2005**, *293*, 572–580. [CrossRef] [PubMed]
38. Srinivasan, B.; Pavya, K. Feature selection techniques in data mining: A study. *Int. J. Sci. Dev. Res. (IJSDR)* **2017**, *2*, 594–598.
39. Saeys, Y.; Inza, I.; Larranaga, P. A review of feature selection techniques in bioinformatics. *Bioinformatics* **2007**, *23*, 2507–2517. [CrossRef] [PubMed]
40. Liu, H.; Yu, L. Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 491–502.
41. Noble, W.S. What is a support vector machine? *Nat. Biotechnol.* **2006**, *24*, 1565–1567. [CrossRef]
42. Hearst, M.A.; Dumais, S.T.; Osuna, E.; Platt, J.; Scholkopf, B. Support vector machines. *IEEE Intell. Syst. Their Appl.* **1998**, *13*, 18–28. [CrossRef]
43. Pisner, D.A.; Schnyer, D.M. Support vector machine. In *Machine Learning*; Elsevier: Amsterdam, The Netherlands, 2020; pp. 101–121.
44. Steinwart, I.; Christmann, A. *Support Vector Machines*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2008.
45. Nusinovici, S.; Tham, Y.C.; Yan, M.Y.C.; Ting, D.S.W.; Li, J.; Sabanayagam, C.; Wong, T.Y.; Cheng, C.Y. Logistic regression was as good as machine learning for predicting major chronic diseases. *J. Clin. Epidemiol.* **2020**, *122*, 56–69. [CrossRef]
46. Rymarczyk, T.; Kozłowski, E.; Kłosowski, G.; Niderla, K. Logistic regression for machine learning in process tomography. *Sensors* **2019**, *19*, 3400. [CrossRef] [PubMed]
47. John, G.H.; Langley, P. Estimating continuous distributions in Bayesian classifiers. *arXiv* **2013**, arXiv:1302.4964.
48. Charbuty, B.; Abdulazeez, A. Classification based on decision tree algorithm for machine learning. *J. Appl. Sci. Technol. Trends* **2021**, *2*, 20–28. [CrossRef]
49. Chen, W.; Xie, X.; Wang, J.; Pradhan, B.; Hong, H.; Bui, D.T.; Duan, Z.; Ma, J. A comparative study of logistic model tree, random forest, and classification and regression tree models for spatial prediction of landslide susceptibility. *Catena* **2017**, *151*, 147–160. [CrossRef]
50. Rodriguez-Galiano, V.F.; Chica-Rivas, M. Evaluation of different machine learning methods for land cover mapping of a Mediterranean area using multi-seasonal Landsat images and Digital Terrain Models. *Int. J. Digit. Earth* **2014**, *7*, 492–509. [CrossRef]
51. Arjun, P. *Machine Learning and AI for Healthcare: Big Data for Improved Health Outcomes*; Apress: London, UK, 2019.
52. Fayed, H.A.; Atiya, A.F. Speed up grid-search for parameter selection of support vector machines. *Appl. Soft Comput.* **2019**, *80*, 202–210. [CrossRef]
53. Malley, T.O.; Bursztein, E.; Long, J.; Chollet, F.; Jin, H.; Invernizzi, L. Hyperparameter Tuning with Keras Tuner. 2019. Available online: https://github.com/keras-team/keras-tuner (accessed on 8 November 2022).

diagnostics

MDPI

Article

Advanced Time-Frequency Methods for ECG Waves Recognition

Ala'a Zyout [1], Hiam Alquran [1], Wan Azani Mustafa [2,3,*] and Ali Mohammad Alqudah [1]

[1] Department of Biomedical Systems and Informatics Engineering, Yarmouk University, Irbid 21163, Jordan
[2] Faculty of Electrical Engineering & Technology, Campus Pauh Putra, Universiti Malaysia Perlis, Arau 02000, Perlis, Malaysia
[3] Advanced Computing, Centre of Excellence (CoE), University Malaysia Perlis (UniMAP), Arau 02000, Perlis, Malaysia
* Correspondence: wanazani@unimap.edu.my

Abstract: ECG wave recognition is one of the new topics where only one of the ECG beat waves (P-QRS-T) was used to detect heart diseases. Normal, tachycardia, and bradycardia heart rhythm are hard to detect using either time-domain or frequency-domain features solely, and a time-frequency analysis is required to extract representative features. This paper studies the performance of two different spectrum representations, iris-spectrogram and scalogram, for different ECG beat waves in terms of recognition of normal, tachycardia, and bradycardia classes. These two different spectra are then sent to two different deep convolutional neural networks (CNN), i.e., Resnet101 and ShuffleNet, for deep feature extraction and classification. The results show that the best accuracy for detection of beats rhythm was using ResNet101 and scalogram of T-wave with an accuracy of 98.3%, while accuracy was 94.4% for detection using iris-spectrogram using also ResNet101 and QRS-Wave. Finally, based on these results we note that using deep features from time-frequency representation using one wave of ECG beat we can accurately detect basic rhythms such as normal, tachycardia, and bradycardia.

Keywords: ECG; iris-spectrogram; scalogram; CNN; ResNet101; ShuffleNet; heart rhythm

Citation: Zyout, A.; Alquran, H.; Mustafa, W.A.; Alqudah, A.M. Advanced Time-Frequency Methods for ECG Waves Recognition. *Diagnostics* **2023**, *13*, 308. https://doi.org/10.3390/diagnostics13020308

Academic Editor: Sameer Antani

Received: 11 December 2022
Revised: 7 January 2023
Accepted: 10 January 2023
Published: 13 January 2023

1. Introduction

The heart is a muscle that pumps blood throughout the body, and contracts rhythmically. The atrial sine node, which functions as a natural pacemaker, initiates this contraction, which then spreads to the rest of the muscle. There is a pattern to the way that this electrical pulse spreads [1]. This action causes fluctuations in the skin's surface's electrical potential by producing electric currents on the body's surface. Electrodes and proper tools can be used to record or measure these signals, known as an electrocardiogram (ECG) [2].

An ECG signal is composed of three major components explained in Figure 1 [3]: P-wave; QRS complex, which contains three waves, i.e., Q, R, and S; and the T-wave [4]. The P-wave is a small flexure wave indicating atrial depolarization, ventricular depolarization is represented by the QRS complex, and the T-wave is indicative of ventricular repolarization (atrial repolarization is hidden by the large QRS complex) [5]. The amplitudes and frequencies of these waves are shown in Table 1 below [4,6].

When there is no disease or abnormality in the waveform of the ECG signal, the heart's regular rhythm is known as a normal sinus rhythm (NSR). Typically, the heart rate of NSRs ranges from 60 to 100 beats per minute. The breathing cycle causes a small change in the R-R interval's regularity. Sinus tachycardia is the name for the rhythm when the heart rate rises above 100 beats per minute and the R-R interval decreases. This is the heart's normal response to the need for increased blood circulation; it is not an arrhythmia. However, overly rapid heartbeats result in incomplete filling of the ventricles before contraction, which lowers pumping effectiveness and negatively impacts perfusion. Bradycardia, which

occurs when the heartbeat is extremely slow, can have a significant negative impact on important organs and the heart rate drops down to 60 beats per minute, and the R-R interval increases [7].

The paper is organized as follows: Section 2 provides details about the state-of-the-art-related works. Section 3 presents an explanation of the used dataset, the preprocessing and segmentation of the ECG, iris spectrogram, and scalogram. Section 4 shows the results, including the performance of the proposed CNN classifiers, and the discussion about the proposed method results. Finally, Section 5 represents the conclusion of the work.

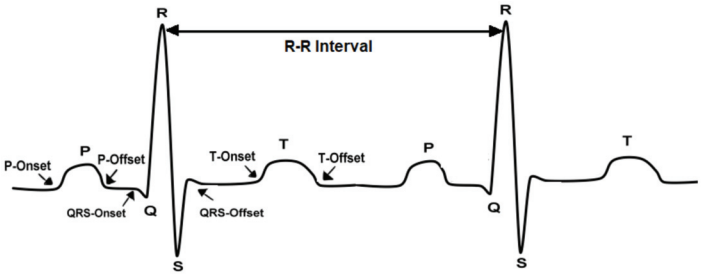


Figure 1. ECG signal components show the onset and offset of each wave.

Table 1. The amplitudes and frequencies of ECG waves.

ECG Waves	Amplitude	Frequency
P-Wave	0.25 mV	5–30 Hz
QRS-Complex	The amplitude for the largest wave R is 1.6 mV	8–50 Hz
T-Wave	0.1–0.5 mV	0–10 Hz

2. Literature Review

Classification of normal and arrhythmia-associated ECG is an important goal to achieve better detection and proper identification of various cardiovascular diseases (CVDs). However, the small amplitude and duration of the ECG arrhythmia can make it difficult to classify. With the rise of deep learning techniques, several recent studies have used very deep networks for ECG classification. Here, we will attempt to detail the latest related works using time-frequency methods for ECG classification.

Rashed Al-Mahfuz et al. proposed a novel ECG beat classifier using a customized VGG16-based Convolution Neural Network (CNN) with two advanced time-frequency representation techniques, Continuous Wavelet Transforms (CWT) and Hilbert-Huang transform (HHT), to identify the best time-frequency representation of ECG beats. The proposed adopted CNN with CWT scalogram achieved 100% classification accuracies on MIT-BIH arrhythmia database for 2–4 classes and 99.90% for 5 classes, and the CWT scalogram outperformed the HHT spectrum in all the cases [8].

Swain et al. introduced an automated identification of myocardial infarction (MI) using a modified Stockwell transform (MST)-based time-frequency analysis and a phase information distribution pattern method. Both healthy and MI ECG signals are collected from the PTB diagnostic ECG database with 12 lead ECG signals; the results of the proposed method can detect the MI successfully with an accuracy, sensitivity, and specificity of 99.93%, 99.97%, and 99.30% respectively [9]. Additionally, Lekhal et al. introduced an ECG beat classifier system based on features observed in time–frequency analysis using a variant of the Stockwell transform, and then the SVM with asymmetric costs (AS3VM) was applied for assessment of the feature performance. The proposed method has been evaluated on the MIT-BIT arrhythmia database, using four types: normal beats (N), left and right bundle branch blocks (L and R), and premature ventricular contractions (V). The obtained results

show accuracies of 99.35%, 98.73%,98.57%, and 99.44% respectively, for N, L, R, and V beats [10].

However, a suitable method for telemedicine systems provided by Kayikcioglu et al. to classify ST segment using time-frequency distribution based on features from multi-lead ECG signals of four-class and tested them on three different databases, MIT-BIH Arrhythmia database, European ST-T database, and Long-Term ST database. The weighted k-NN algorithm achieved the best average performance with an accuracy of 94.23%, a sensitivity of 95.72%, and a specificity of 98.15% using the Choi–Williams time-frequency distribution features, in addition to the other classification algorithms SVM and Ensemble [11].

Kłosowski et al. proposed an effective method for ECG classification using the deep neural long-short-term memory (LSTM) network and feature extraction consists of converting the ECG signal into a series of spectral images using short-term Fourier transformation. Then, the images were converted using Fourier transform again to two signals, which include instantaneous frequency and spectral entropy, which are used to train the LSTM network [12].

In 2021, Wang et al. provided a simple and accurate method, which can be used as a clinical auxiliary diagnostic tool, and is an automatic ECG classification method based on Continuous Wavelet Transform (CWT) to obtain different time-frequency components and Convolutional Neural Network (CNN) to extract features from the 2D scalogram composed of the time-frequency components. The method achieved an accuracy of 98.74%, a sensitivity of 67.47%, an F1-score of 68.76%, which compared with existing methods is increased by 4.75~16.85%, and a positive predictive value of 70.75% [13].

In the same year, Hussein et al. presented a novel method to extract ST and PR features from the Choi–Williams time-frequency distribution proposed for myocardial ischemia identification. With the use of these extracted features, a multi-class SVM classifier is trained to detect unknown circumstances and assess whether they are ischemic or normal. Improved detection performance is the result of using multi-lead ECG for classification and 1 min intervals rather than beats or frames. The proposed strategy produced a final result that had an overall accuracy, sensitivity, and specificity of 99.09%, 99.49%, and 98.44%, respectively [14].

Furthermore, in 2022, Alqudah et al. published a paper in which they present a method that is efficient, simple, fast, and deployable on mobile devices. A deep learning methodology was developed to detect up to 17 classes of cardiac arrhythmia based on analyzing a single ECG beat and calculating the iris spectrogram to feed the convolutional neural network. The results show that the proposed methodology has an overall recognition accuracy of 99.13% $\pm$ 0.25, 98.223% $\pm$ 0.85, and 97.494% $\pm$ 1.26 for 13, 15, and 17 arrhythmia classes, respectively. The training/testing is performed using tenfold cross-validation [15].

Faraget et al. provided a short-time Fourier Transform (STFT) Convolutional Neural Network (CNN) model for ECG classification in real-time at the edge. To extract the spectrogram from the input ECG signal, they developed an STFT-based 1D convolutional (Conv1D) layer and then reshaped it into a 2D heat-map image to feed the 2D convolutional (Conv2D) neural network (CNN) for classification. The proposed classifier achieved 99.1% accuracy and a 95% F1-score at the edge with a maximum model size of 90 KB, an average inference time of 9 ms, and a maximum memory usage of 12 MB [16].

This study aims to propose a comparison between two different advanced time-frequency methods, i.e., iris spectrogram and scalogram, to categorize the previous types of ECG, i.e., Normal, Tachycardia, and Bradycardia, using deep learning with Resnet101 and ShuffleNet convolutional neural networks.

3. Materials and Methods

The methodology of this research can be concentrated on the design and implementation of the procedure steps to achieve the overall approach. The first step is preparing ECG data with different types to be ready for the system implementation. The implemented approach for ECG wave recognition is passed into three main steps, as shown in Figure 2,

the Preprocessing and Segmentation process, Time-Frequency methods, and diagnosis and Recognition using Deep Learning.

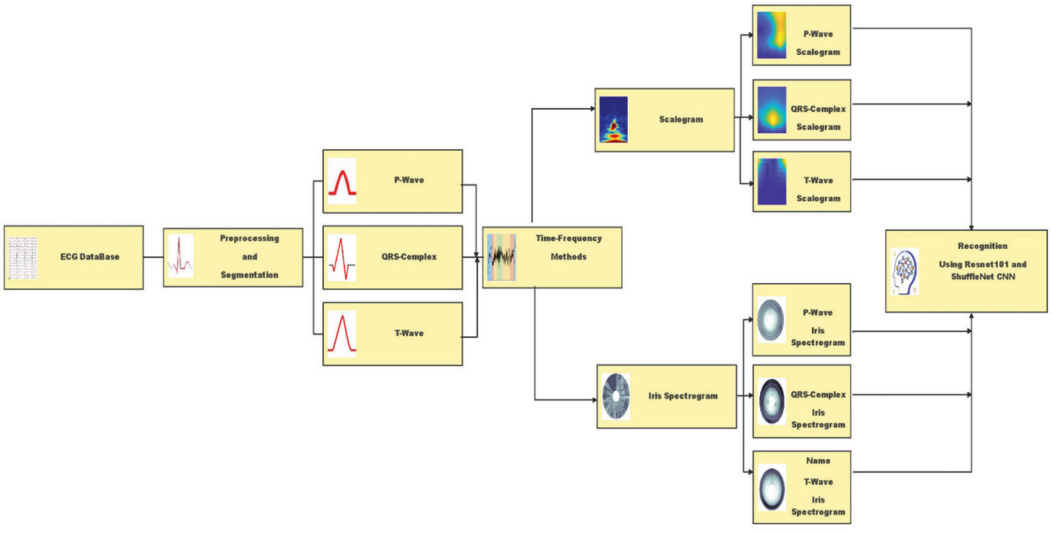


Figure 2. Block diagram of the proposed methodology.

3.1. Dataset

ECG-ID Database is used in this study for normal ECG type; the database contains 310 ECG recordings, obtained from 90 persons, and was created and contributed by Tatiana Lugovaya, who used it in her master's thesis [17]. Additionally, the Challenge 2015 Training Sets is used for tachycardia and a bradycardia type; the training set contains 750 recordings for five ECG categories. However, two types were selected [18]. The length of each ECG signal segment is 10 s. The applied segmentation methodology (discussed in the next section) results in 316 normal beats, 138 Bradycardia, and 326 Tachycardia images for all methods, with a total of 780 images.

3.2. ECG Preprocessing and Segmentation

The main objective of this processing is to distinguish the P, QRS, and T waves and detect the characteristic points P-Onset, P-Offset, QRS-Onset, QRS-Offset, and T-Onset, T-Offset for each cycle. After identifying the ECG waves, their respective amplitudes are measured concerning the baseline. Before analysis, an ECG signal is typically the first bandpass filtered using several frequency ranges. Bandpass filtering is widely used to remove low- and high-frequency noise components, baseline wander muscle noise, and power line interference [19]. The frequency range used is 0.5–40 Hz [20,21].

For de-noising and baseline wandering removal, different types of wavelet transforms were usually applied to ECG signals. Advanced signal processing methods, such as the stationary wavelet transform de-noising technique, should be employed to eliminate various noise types that contain muscle artefacts and electrode moving artefacts [22]. Baseline wanders removal and de-noising were achieved by multiresolution wavelet transform [23]. In this study, the noisy signal was decomposed into nine levels by using the Daubechies wavelet db8. The de-noised signal was recovered by taking the inverse discrete wavelet transform of the resulting coefficients. The next step in the preprocessing stage used is amplitude normalization, which is optional but useful for visually comparing data from various patients and datasets [24]. To ensure the signal starts with P-wave and ends with T-wave, find the peaks of the R-wave. The onset of the P-wave for the signal is the half peak-to-peak distance between the two first peaks and the offset of the T-wave for the

signal is also the half peak-to-peak distance between the two last peaks. Figure 3 below show the preprocessing stages for all ECG data.

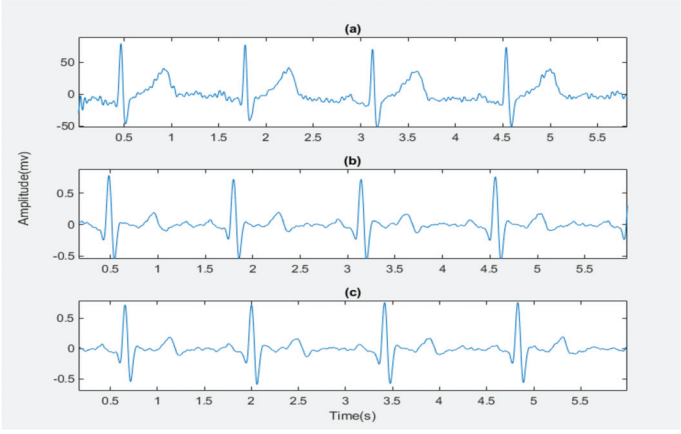


Figure 3. The preprocessing stages for normal ECG signal for example: (**a**) The Original ECG signal; (**b**) The Preprocessed ECG signal; (**c**) The ECG signal starting with the P-wave and ending with the T-wave.

In this work, we have adopted the QRS-waves detection algorithm, which is developed by Faruk U, who proposed a new thresholding method on the Pan–Tompkins algorithm [25] and found the maximum peaks in the left and the right of the R-wave within a window to detect the P-wave and the T-wave of the same beat. Furthermore, in this work, we have depended on the zero crossing to detect the onset and offset of each wave; the P-wave and T-wave onset and offset was detected by beginning from the waves' peaks and by searching backwards and forward, respectively, to find zero crossing, which is the onset and offset of the two waves, respectively. From the P-wave offset to Q-wave, a forward search is carried out in this window to find the last zero crossing point. This point is the QRS onset. To find the QRS offset, we take a window from the S-wave to T-wave onset and search forward in this window to find the first zero crossing. Figures 4 and 5 illustrates the process of segmentation.

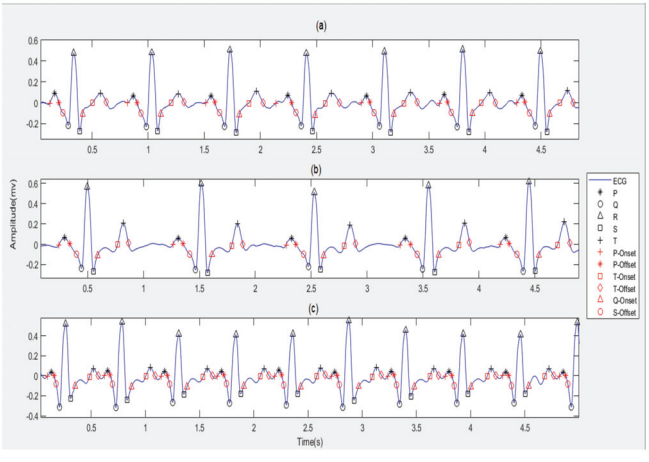


Figure 4. The peaks and intervals detection for ECG types: (**a**) The Normal ECG; (**b**) The Bradycardia ECG; (**c**) The Tachycardia ECG.

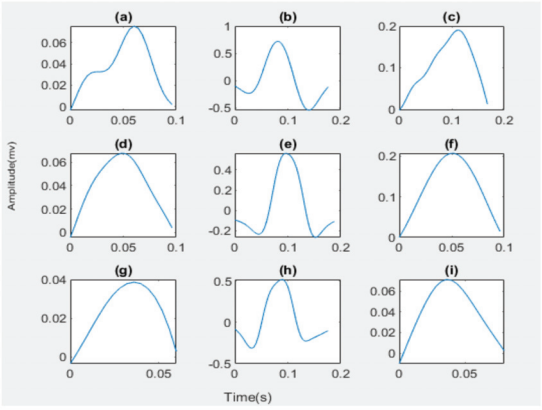


Figure 5. The wave segmentation for ECG types: (**a**) The Normal P-wave; (**b**) The Normal QRS-waves; (**c**) The Normal T-wave; (**d**) The Bradycardia P-wave; (**e**) The Bradycardia QRS-waves; (**f**) The Bradycardia T-wave; (**g**) The Tachycardia P-wave; (**h**) The Tachycardia QRS-waves; (**i**) The Tachycardia T-wave.

For segmenting, ECG waves are considered at an extracted onset and offset of each wave separately from the ECG signal for all ECG types as shown in Figure 5. A total of 320 for normal, 140 for bradycardia, and 325 for tachycardia P-waves, QRS-waves, and T-waves are extracted.

3.3. Time-Frequency Representations

Time-frequency representations define the frequency content of a signal as a function of time [26]. Time-frequency analysis and representation are used in the fields of signal and image processing, data analysis, measurements, acoustics and vibration, machinery diagnosis, seismology, etc. for the analysis of signals and data, fundamental frequency detection, instantaneous frequency determination, etc. [27]. Two of the most commonly used time–frequency analysis tools include the short-time Fourier transform (STFT) and continuous wavelets transform (CWT) [28].

STFT is a series of Fourier transforms of a windowed signal. When a signal's frequency components change over time, the STFT gives time-localized frequency information, whereas the conventional Fourier transform provides frequency information averaged across the whole signal time interval. The spectrogram, which is an intensity representation of STFT magnitude over time, is frequently used to visualize STFT [29]. The STFT is given by:

$$X(n,\omega) = \sum_{m=-\infty}^{\infty} \omega[n-m]x[m]e^{-j\omega n} \quad (1)$$

m is a "dummy" time argument whereas n represents the location of the short segment of the original time function as it is obtained by the window $\omega[n-m]$ which moves along the m-axis according to the value of n.

Wavelet decomposition uses a scale rather than a frequency to translate a signal onto a time-scale plane. The time-frequency plane of the STFT is the same as this, and each scale of the time-scale plane corresponds to a certain frequency range of the time-frequency plane. When comparing the wavelet with the Fourier transform, the wavelet decomposes the signal into shifted or scaled shapes from a mother wavelet, whereas the Fourier analysis decomposes the signal into sinusoids of various frequencies [30].

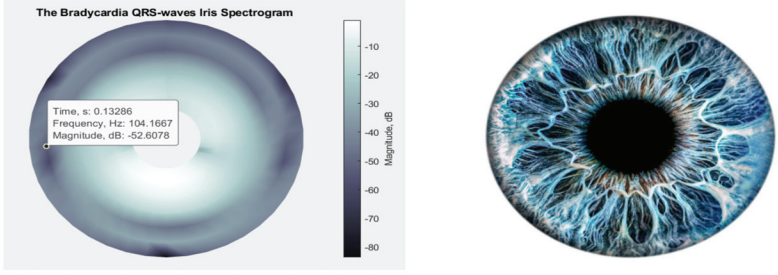
The CWT is the sum of the signal $x(t)$ multiplied by shifted and scaled shapes from a mother wavelet $\varphi(t)$ [30]:

$$CWT(scale, position) = \int_{-\infty}^{+\infty} x(t) * \varphi(scale, position, t)dt \quad (2)$$

At this time, frequency–time analysis allows us to know how the signal is distributed with frequency and phase, so complex signals can be expressed concisely and analyzed easily. However, by visually representing signals at various scales and various frequencies through CWT, hidden features can be seen in the time-frequency domain [30].

3.3.1. Irisgram ECG Representation

In 2018, H. Zhivomirov proposed an innovative method for visualizing the outcomes of time-frequency analysis called an "iris-spectrogram" or "irisgram" for its resemblance to a human iris, as shown in Figure 6. The irisgram is a circular representation of the traditional spectrogram in which the signal strength is indicated axially with color and time increases circumferentially in a clockwise direction. To pinpoint the precise location of a given point in the time-frequency domain, one must utilize a Data Cursor tool because the circular design of the irisgram does not permit annotations to be placed on the time and frequency axes [27].



(**a**) (**b**)

Figure 6. (**a**) An irisgram of QRS-waves; (**b**) Human eye iris.

Times and frequencies values converted into the polar coordinates to generate an irisgram, using the following Equations (3)–(5), are applied [31]:

$$\theta = \left\{ -\pi : \frac{2\pi}{T-1} : \pi \right\}, \textit{ where } T \textit{ is the length of time vector.} \tag{3}$$

$$\rho = \frac{\max(f)}{3} + f, \textit{ where } f \textit{ is the frequency vector.} \tag{4}$$

$$X = \rho \times \cos\theta \textit{ , and } Y = \rho \times \sin\theta. \tag{5}$$

Then, compute the spectrogram of power spectral density (PSD) and convert to amplitude spectrum in dB within the range −120 dB to 120 dB, which are the values of the Z plane [32]. Using the (surf) function, the values of X and Y are plotted against Z to create a 3D surface plot. This sort of plot contains solid edge colors and face colors, and the color of the surface varies according to the heights that Z specifies. This function is used to plot the values in matrix Z as heights (weights) above a grid in the X-Y plane [15].

In this research, for all types of segmented ECG waves, the irisgram representation has been generated using the function *irisgram*, accessed in MATLAB® Central File Exchange [32], and the resulting images are stored with their relevant classes, as shown in Figure 7. Then, the deep learning models will be applied to the stored images, as will be shown next.

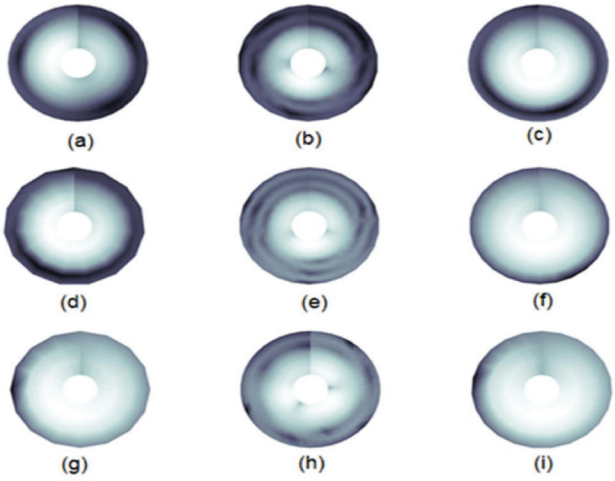


Figure 7. The irisgrams for ECG types: (**a**) The Normal P-wave irisgram; (**b**) The Normal QRS-waves irisgram; (**c**) The Normal T-wave irisgram; (**d**) The Bradycardia P-wave irisgram; (**e**) The Bradycardia QRS-waves irisgram; (**f**) The Bradycardia T-wave irisgram; (**g**) The Tachycardia P-wave irisgram; (**h**) The Tachycardia QRS-waves irisgram; (**i**) The Tachycardia T-wave irisgram.

3.3.2. Scalogram ECG Representation

The scalogram (SG) is a time-frequency representation of the signal constructed by a wavelet transformation, where coefficient values at specific time-frequency locations can be indicated by color [33]. The SG of x is defined by the following equation:

$$S(x) = \| W_a x(b) \| \sqrt{\int_{-\infty}^{+\infty} x(t)\varphi(\frac{t-B}{A})} da \quad (6)$$

Which represents the energy of $W_a x(b)$ at the scale A. B defines the translation of the mother wavelet $\varphi(t)$. The SG enables the detection of the signal's most representative scales (or frequencies), or those that contribute the most to the signal's overall energy. By integrating (4) between these values, we may define the appropriate windowed SG if we are only interested in a specific time window (t_0; t_1). The three axes are time (x), scales (y), and coefficient value (z) [34,35]. The ECG wave SG sample is shown in Figure 8.

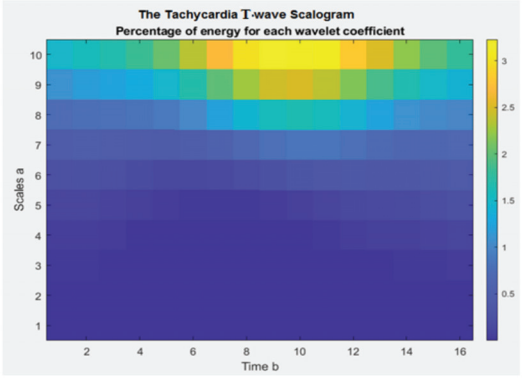


Figure 8. The Tachycardia T-wave SG.

Additionally, in this research, for all types of segmented ECG waves, the SG representation has been generated using the function *wscalogram*, in MATLAB®, with different scaling in each wave, as shown in Table 1, and the resulting images are stored with their relevant classes, as shown in Figure 9. Then, the deep learning models will be applied to the stored images, as will be shown next.

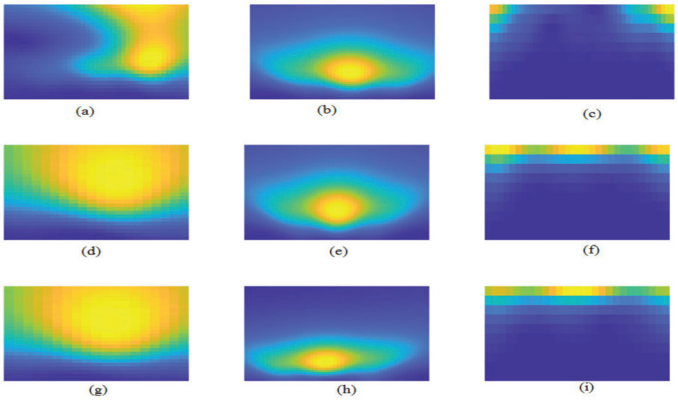


Figure 9. The Scalogram for ECG types: (**a**) The Normal P-wave SG; (**b**) The Normal QRS-waves SG; (**c**) The Normal T-wave SG; (**d**) The Bradycardia P-wave SG; (**e**) The Bradycardia QRS-waves SG; (**f**) The Bradycardia T-wave SG; (**g**) The Tachycardia P-wave SG; (**h**) The Tachycardia QRS-waves SG; (**i**) The Tachycardia T-wave SG.

3.4. Deep Learning

Usually, deep learning models need a large dataset to train and achieve robust results. Therefore, many researchers have started to employ transfer learning techniques to tune the pre-trained deep learning structures to perform the intended task.

The first convolutional neural network is the residual neural network, which is distinguished by its residual block property. This feature enhances the performance of classification by overcoming the problems of vanishing or exploding gradients due to deep learning layers. ResNet allows forming a skipping connection which enables activating a layer to further layers by skipping some layers in between. There are various versions of ResNet, such as ResNet-18, -34, -50, and -101. These versions are based on the number of deep layers. The architecture ResNet is stacking of such residual blocks. The input size of these networks is 224 × 224 × 3 [36].

The second convolutional neural network is ShuffleNet, which is one of the most effective networks utilized in mobile applications. To obtain a high accuracy level, ShuffleNet performs two types of convolutions: point-wise group convolution and channel convolution, which makes its performance efficient and fast. It consists of a stacking of ShuffleNet blocks, each one consisting of two grouped convolutional layers, channel shuffle layer, in addition to depth-wise convolutional layers. The output from each block maps using the ReLU layer. The designed input layer is compatible with image size 224 × 224 × 3 [37,38].

4. Results

The resulting images were utilized to build deep learning models either using ResNet101 or ShuffleNet. ECG signal is segmented into three waves P, QRS, and T. Each segment proceeds with irisgram and scalogram, separately. For each wave, there are two generated colored images; one for scalogram and the other for irisgram. The labeled data are recognized based on ECG diagnosis, normal, bradycardia, or tachycardia. For each class, six datasets are achieved; three ECG segments for each category in both signals' representations, i.e., irisgrams and scalograms. The classification is performed using two pre-trained deep-learning structures ResNet and ShuffleNet. The resulting representation images are

divided into 70% training and 30% testing. The corresponding sections demonstrate the analysis of the results.

4.1. Irisgram Representation

4.1.1. ResNet

The iris image classification is executed using pre-trained ResNet101 structures, and the corresponding matrices illustrate its performance. The first one shows the performance of the irisgram of the P-waves, and the second one illustrates the capabilities of the irisgram of the T-waves. The third represents the performance of the irisgram of the QRS waves.

For the irisgram P-wave, as shown in Figure 10a, the sensitivity of bradycardia is 85.4%, where 35 out of 41 cases are classified correctly. The precision of bradycardia is 94.6%. Meanwhile, 87 segments were discriminated from 95 P-waves for normal subjects, with a sensitivity of 91.6% and a positive predictive value of 98.9%. A true positive rate of tachycardia is the highest value at 98.9%. On the other hand, its precision is the lowest because 14 cases were misrecognized as tachycardia. The overall accuracy is 92.7% for all classes.

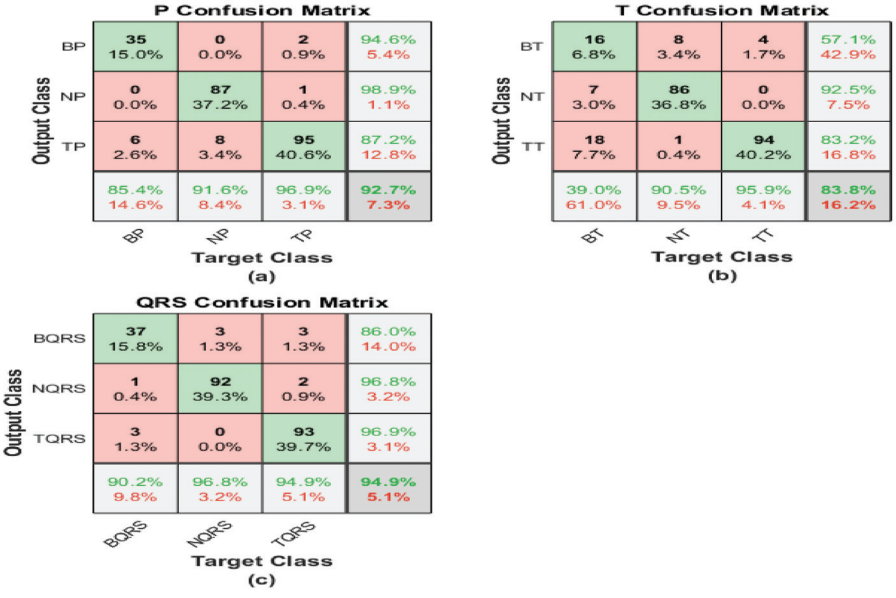


Figure 10. Confusion matrices of ECG irisgram using ResNet101: (**a**) P-waves irisgram; (**b**) T-waves irisgram; (**c**) QRS waves irisgram.

The confusion matrix describes the outputs of T-waves. The results are not promising. Seven segments are misclassified as healthy and 18 cases of bradycardia are misclassified as tachycardia. Therefore, the sensitivity is too low for bradycardia cases, at 39%. Moreover, 14 cases from other classes are misclassified as bradycardia by the worst precision of 57%. The performance of ResNet is better regarding normal class recognition. Nine samples are misclassified, which is one as bradycardia and the rest as tachycardia, with a sensitivity of 90.5% and a misclassification rate of 9.5%. Regarding discrimination between tachycardia and normal classes, there are seven classes of bradycardia classified as normal and the precision is 92.5%. Tachycardia's sensitivity reaches 95.9%, and the misclassification rate is 4.1%. Furthermore, 19 cases of bradycardia cases are recognized as tachycardia. Therefore, the precision was reduced to 82.2%. The overall accuracy is the lowest, and it does not exceed 83.3%.

For the QRS wave irisgram, results show an improvement in discrimination between three classes in terms of sensitivity, accuracy, and precision. In the bradycardia class, 37 cases are distinguished correctly from 41 and six misclassified samples as bradycardia. That is why the precision reduced to 86% from what it was in the P-waves case. Recall and PPV are the highest for healthy segments by almost 97% for both performance terms. Tachycardia obtains a high level of precision using the QRS-waves, in which just three samples from the whole data were misclassified as tachycardia, and the sensitivity is 94.9%. The overall accuracy of the proposed approach regarding QRS-waves irisgram is 94.9%.

4.1.2. ShuffleNet

The irisgram images are proceed using ShuffleNet. The process is started by splitting the dataset into 70% training and 30% tests. That operation is executed on each ECG segment. The following confusion matrices characterize the performance of the test phase of the whole database.

The first confusion matrix in Figure 11a represents the P-waves irisgram images. The sensitivity of bradycardia is 90.2%, where 37 cases are classified correctly from 41. The precision of bradycardia is 78.7%. Meanwhile, 94 segments were distinguished from 95 p-waves for normal subjects, with the highest sensitivity of 98.9% and a best positive predictive value reaching 100%. A true positive rate of tachycardia is 89.8%. On the other hand, its precision is the lowest because 14 cases were misrecognized as tachycardia. The overall accuracy is 93.6% for all classes.

Figure 11. Confusion matrices of ECG irisgram using ShuffleNet: (**a**) P-waves irisgram; (**b**) T-waves irisgram; (**c**) QRS-waves irisgram.

The second confusion matrix represents the outputs of T-waves representation. The results are not adequate. Nine segments are misclassified as healthy, and seven cases of bradycardia are misclassified as tachycardia. Therefore, the sensitivity is too low for bradycardia cases at 61%. However, nine cases from other classes are misrecognized as bradycardia by the worst precision of 73.5%. The accomplishment of ShuffleNet is better than normal class discrimination. Seven samples are misclassified as bradycardia, and zero samples as tachycardia, with high sensitivity of 92.9%, and a misclassification rate of 7.1%. There are seven classes of bradycardia classified as normal for discrimination between

tachycardia and normal classes, and the precision does not exceed 92.5%. Tachycardia's sensitivity reaches 95.9%, and the misclassification rate is 4.1%. Moreover, 19 cases of bradycardia cases are classified as tachycardia. Therefore, the precision was reduced to 82.2%. The overall accuracy is the lowest, and it does not exceed 83.3%

For the QRS waves irisgram, results show an improvement in discrimination between three classes in terms of sensitivity, accuracy, and precision. In the bradycardia class, 31 cases are classified correctly from 41, and four misclassified samples as bradycardia. That is why the precision reduced to 88.6% from what it was in the P waves case. Recall and PPV are the highest for normal ECG segments by almost 94.8% for both performance evaluation terms. Tachycardia obtains a high level of precision using the QRS-waves, in which just five segments from all the data were misclassified as tachycardia, and the highest sensitivity reaches 100%. The overall accuracy of the proposed approach regarding QRS-waves irisgram is 94.0%.

4.2. Scalogram Representation

4.2.1. ResNet

The scalogram image recognition is performed utilizing pre-trained ResNet101 architecture, and the following matrices represent its performance. The first matrix describes the accomplishment of the scalogram of the P-waves, while the second one shows the capability level of the scalogram of the T-waves. On the other hand, the third one demonstrates the performance of the scalogram of the QRS-waves.

The first confusion matrix in Figure 12a indicates P-waves scalogram images. The sensitivity of bradycardia is 85.4%, where 35 cases are discriminated correctly from 41. The precision of bradycardia is almost moderate by 87.5%. While 93 segments were distinguished from 95 P-waves for healthy subjects, with the highest sensitivity of 97.9% and a best positive predictive value reaching 100%. The true positive rate of tachycardia is 94.9%. The precision is 92.1% because 14 cases were misclassified as tachycardia. The overall accuracy is 94.4% for all classes.

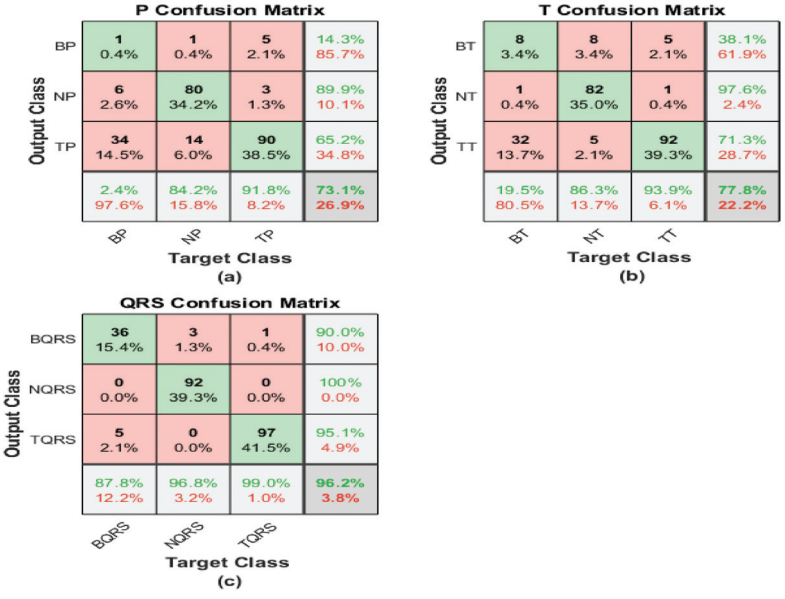


Figure 12. Confusion matrices of ECG SGs using ResNet101: (**a**) P-waves SG; (**b**) T-waves SG; (**c**) QRS-waves SG.

The second confusion matrix describes the output of T-waves representation. The results are better than the P-waves and QRS-waves scalogram. Two segments are misclassified as healthy, and no cases of bradycardia are misclassified as tachycardia. Therefore, the sensitivity is too high for bradycardia cases at 95.1%. However, two cases from other classes are misrecognized as bradycardia by the highest precision of 95.1%. The performance of ShuffleNet is better than normal class discrimination. Two samples are misrecognized as bradycardia, and zero samples as tachycardia, with high sensitivity of 97.9%, and a misclassification rate of 2.1%. For discrimination between tachycardia and normal classes, there are no misclassification segments with the highest precision reaches to 100%. Tachycardia's sensitivity is the best by 100%. Moreover, only two cases of bradycardia cases are misclassified as tachycardia. Therefore, the precision is high at 98%. The overall accuracy is the highest and reaches 98.3%.

QRS-waves SG is presented in the third confusion matrix. In the bradycardia class, 18 cases are classified correctly from 41, and 19 segments are misclassified as bradycardia. That is why the precision is too low reaching 48.6% from what it was in the P-waves case. Recall and PPV are the highest for normal ECG segments by almost 92.8% for both performance evaluation terms. Tachycardia obtains an acceptable level of precision using QRS-waves, in which 18 segments from the whole data were misclassified as tachycardia, while the sensitivity reaches 83.7%%. The overall accuracy of the proposed approach regarding the QRS-waves SG is 81.0%.

4.2.2. ShuffleNet

The scalogram image recognition is executed using a pre-trained ShuffleNet structure, and the corresponding matrices illustrate its performance. The first matrix represents the performance of the scalogram of the P-waves, while the second one shows the capability level of the scalogram of the T-waves. On the other hand, the last confusion demonstrates the performance of the scalogram of the QRS waves.

The first confusion matrix in Figure 13a indicates P-waves scalogram images. The sensitivity of bradycardia is 2.4%, where just one case is discriminated correctly from 41. The precision of bradycardia is too low at 14.3%. While 80 segments were distinguished from 95 P-waves for healthy subjects, with moderate sensitivity of 84.2% and a positive predictive value reaching 89.9%. A true positive rate of tachycardia is 91.8%. The precision is 65.2%, because eight cases were misclassified as tachycardia. The overall accuracy is 73.1% for all classes.

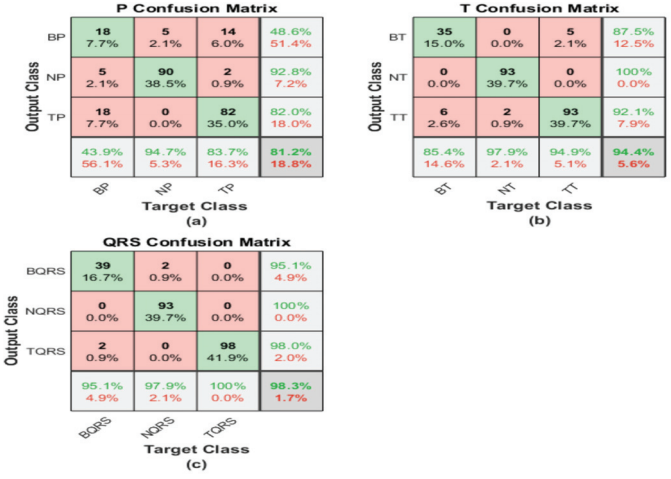


Figure 13. Confusion matrices of ECG SGs using ShuffleNet: (**a**) P-waves SG; (**b**) T-waves SG; (**c**) QRS-waves SG.

The second confusion matrix describes the outputs of T-waves representation. The results are almost like a P-waves SG. Eight segments are misclassified as healthy and five cases of bradycardia are misclassified as tachycardia. Therefore, the sensitivity is too low for bradycardia cases at 80.5%. However, 13 cases from other classes are misrecognized as bradycardia by the lowest precision of 38.1%. The performance of ShuffleNet is better than normal class discrimination. Eight samples are misrecognized as bradycardia and five samples as tachycardia, with moderate sensitivity of 86.3% and a misclassification rate of 13.7%. For discrimination between tachycardia and normal classes, there are five misclassification segments and 32 segments misclassified for bradycardia with a low precision reach of 71.1%. Tachycardia's sensitivity is the best at 93.9%. Moreover, only five cases of bradycardia cases are misclassified as tachycardia. Therefore, the precision is high at 98%. The overall accuracy is the highest and reaches 77.6%. Tables 2–5 summarize the obtained results. Exploiting the scalogram representation of T-waves and the pre-trained ResNet101 yields a high accuracy of 98.3%.

Table 2. Results of Irisgram and ResNet101.

	P-Wave		**QRS-Wave**		**T-Wave**	
	Sensitivity	**Precision**	**Sensitivity**	**Precision**	**Sensitivity**	**Precision**
Bradycardia	85.40%	94%	90%	86%	39%	57.10%
Normal	91.60%	98.90%	96.80%	96.80%	90.50%	92.50%
Tachycardia	96.90%	87.20%	94.90%	96.40%	95.90%	83.20%
	Accuracy = 92.7%		**Accuracy = 94.9%**		Accuracy = 83.8%	

Table 3. Results of Irisgram and ShuffleNet.

	P-Wave		**QRS-Wave**		**T-Wave**	
	Sensitivity	**Precision**	**Sensitivity**	**Precision**	**Sensitivity**	**Precision**
Bradycardia	90.20%	79%	76%	89%	61%	73.50%
Normal	98.90%	100.00%	95.80%	94.80%	97.90%	91.20%
Tachycardia	89.80%	94.60%	100.00%	95.10%	92.90%	92.90%
	Accuracy = 93.6		**Accuracy = 94**		Accuracy = 89.3	

Table 4. Results of Scalogram and ResNet101.

	P-Wave		**QRS-Wave**		**T-Wave**	
	Sensitivity	**Precision**	**Sensitivity**	**Precision**	**Sensitivity**	**Precision**
Bradycardia	85.40%	88%	44%	49%	95%	95.10%
Normal	97.90%	100.00%	94.70%	92.80%	97.90%	100.00%
Tachycardia	94.90%	92.10%	83.70%	82.00%	100.00%	96.00%
	Accuracy = 94.4%		Accuracy = 81.2%		**Accuracy = 98.3%**	

Table 5. Results of Scalogram and ShuffleNet.

	P-Wave		**QRS-Wave**		**T-Wave**	
	Sensitivity	**Precision**	**Sensitivity**	**Precision**	**Sensitivity**	**Precision**
Bradycardia	2.40%	14%	88%	90%	20%	38.10%
Normal	84.20%	84.90%	96.80%	100.00%	86.50%	97.60%
Tachycardia	91.80%	65.20%	99.00%	95.10%	93.90%	71.30%
	Accuracy = 73.1%		**Accuracy = 96.2%**		Accuracy = 77.8%	

QRS-waves SG is presented in the third confusion matrix. In the bradycardia class, 36 cases are classified correctly from 41, and 4 segments are misclassified as bradycardia. That is why the precision is almost high reaching 90.0% from what it was in the P-wave case. Recall and PPV are the highest for normal ECG segments by almost 96.8% and 100%, respectively. Tachycardia obtains the best level of precision using the QRS-waves, in which just one segment from the whole data was misclassified as tachycardia, while the sensitivity is the best too, reaching 99.0%. The overall accuracy of the proposed approach regarding the QRS-waves SG is 96.2%.

To check the validity of features extracted using the models, we check the class activation maps (CAMs) and we find that all the trained models have selected the most significant regions of the scalogram and irisgram. Figures 14–16 show a sample if CAM using ShuffleNet using the last ReLU layer and scalogram for the three classes Normal, Bradycardia, and Tachycardia, respectively.

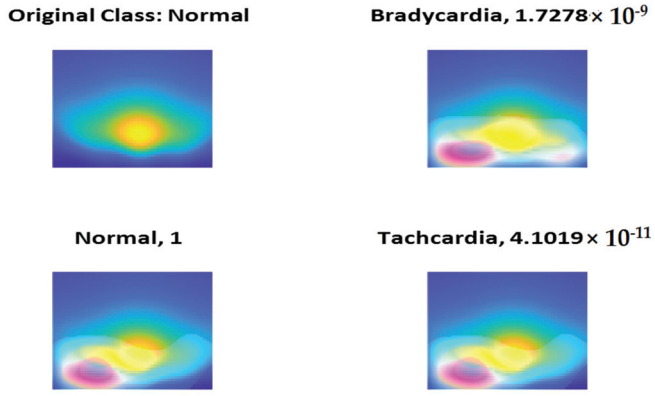


Figure 14. Normal Class CAM using ShuffleNet and Scalogram.

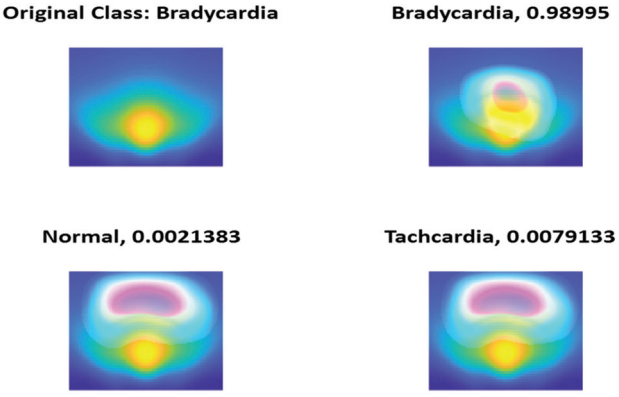


Figure 15. Bradycardia Class CAM using ShuffleNet and Scalogram.

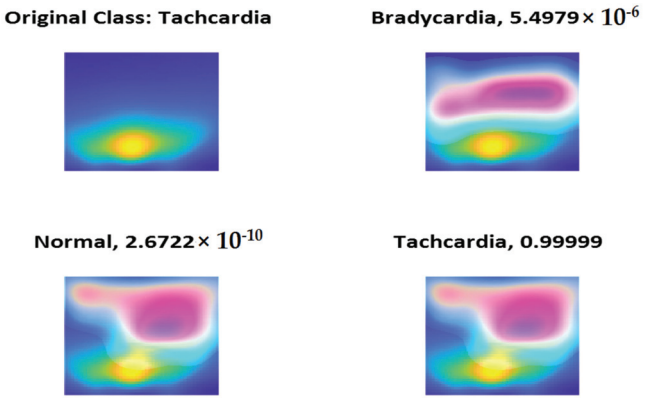


Figure 16. Tachycardia Class CAM using ShuffleNet and Scalogram.

4.3. K-Fold Results

To ensure the validity of the proposed methodology, the datasets' evaluation was performed using a 5 K-fold technique. These techniques were applied on the highest two results gained using each ShuffleNet and ResNet101. The overall confusion matrix and ROC of ResNet101 with Scalogram T-waves are shown in Figure 17. The overall confusion matrix and ROC of ResNet101 with Scalogram T-waves are shown in Figure 18.

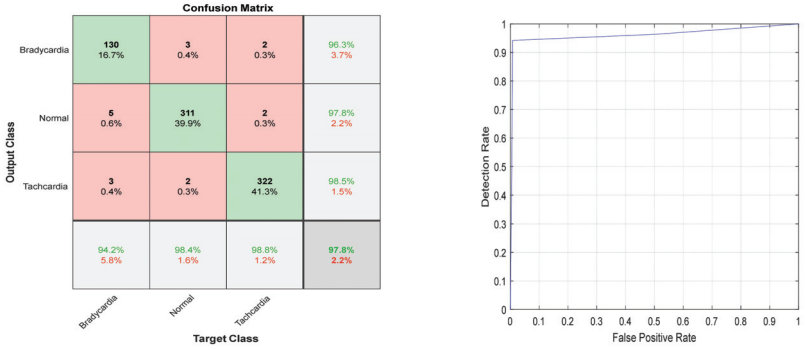


Figure 17. (**A**) Confusion matrix and (**B**) ROC for ResNet101 with Scalogram T-waves.

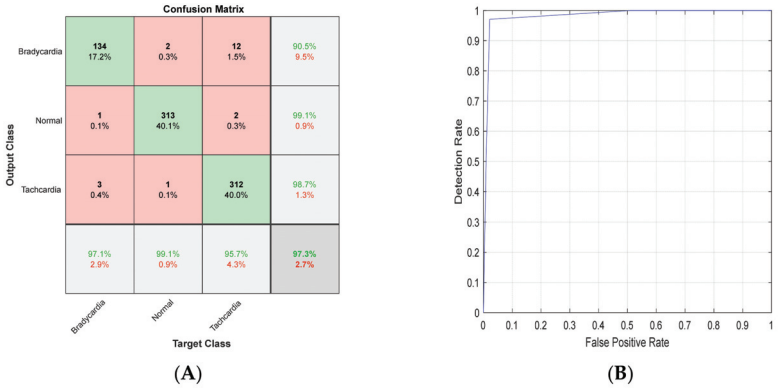


Figure 18. (**A**) Confusion matrix and (**B**) ROC for ShuffleNet with Scalogram QRS-waves.

Then, the performance results of 5 K-fold using the two scenarios are shown in Table 6. Using these results, we can conclude that the performance of the proposed model is stable over different sets of training and testing, which make it robust.

Table 6. Results of 5 K-fold using two scenarios.

	T-Wave with ResNet101	QRS-Wave with ShuffleNet
Sensitivity	97.13 ± 0.95%	97.29 ± 1.30%
Precision	97.52 ± 0.23%	96.11 ± 0.17%
Accuracy	97.82 ± 0.65%	97.31 ± 0.50%

5. Conclusions

This paper introduced a comparison between different ECG waves (P-QRS-T) spectrum representations (iris-spectrogram and scalogram) and two widely used CNN architectures (ResNet101 and ShuffleNet) for classifying three main heart rhythms (Normal, Tachycardia, and Bradycardia). The paper mainly focused on how different spectrum representations of other ECG beat waves combined with one of the two used CNN architectures perform on the classification of arrhythmias to ensure their ability to be applied later using embedded systems. The proposed methodology addressed the main concern of providing high performance with the lowest computational complexity on preprocessing, feature extraction, and classification for classifying ECG beats arrhythmias. The suggested combination methods achieved, in general, high-performance rates with generated images from these spectrums and were fed to CNN architectures. The main advantages of the proposed system are the ability to employ the methodology in embedded systems in the future. The disadvantages are the limited number of records on the used dataset, where a larger dataset is required for further evaluation of the method, yet the current results are very promising.

Author Contributions: Conceptualization, A.Z. and H.A.; methodology, H.A.; software, A.Z. and H.A.; validation, H.A., A.Z. and A.M.A.; formal analysis, A.Z.; investigation, A.Z., H.A., A.M.A. and W.A.M.; resources, A.Z.; data curation, A.Z.; writing—original draft preparation, A.Z., H.A., W.A.M. and A.M.A.; writing—review and editing, W.A.M.; visualization, A.Z. and H.A.; supervision, H.A.; project administration, H.A.; funding acquisition, H.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding. However, thank you to Wan Azani Mustafa for funding dealing and arrangement.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data Availability from the PhysioNet PhysioBank archive: ECG-ID set is used for normal ECG type, and the Challenge 2015 Training Sets is used for tachycardia and bradycardia types. "ECG-ID Database v1.0.0, www.physionet.org/content/ecgiddb/1.0.0", accessed on 19 November 2022. and "PhysioNet/CinC Challenge 2015: Training Sets, archive.physionet.org/physiobank/database/challenge/2015", accessed on 19 November 2022.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Spach, M.S.; Kootsey, J.M. The nature of electrical propagation in cardiac muscle. *Am. J. Physiol.-Heart Circ. Physiol.* **1983**, *244*, H3–H22. [CrossRef]
2. Luz, E.J.D.S.; Schwartz, W.R.; Cámara-Chávez, G.; Menotti, D. ECG-based heartbeat classification for arrhythmia detection: A survey. *Comput. Methods Programs Biomed.* **2016**, *127*, 144–164. [CrossRef]
3. Barros, A.; Resque, P.; Almeida, J.; Mota, R.; Oliveira, H.; Rosário, D.; Cerqueira, E. Data improvement model based on ECG biometric for user authentication and identification. *Sensors* **2020**, *20*, 2920. [CrossRef] [PubMed]
4. Al-Ani, M.S. ECG waveform classification based on P-QRS-T wave recognition. *UHD J. Sci. Technol.* **2018**, *2*, 7–14. [CrossRef]

5. Klabunde, R.E. Cardiac electrophysiology: Normal and ischemic ionic currents and the ECG. *Adv. Physiol. Educ.* **2017**, *41*, 29–37. [CrossRef] [PubMed]
6. Tereshchenko, L.G.; Josephson, M.E. Frequency Content and Characteristics of Ventricular Conduction. *J. Electrocardiol.* **2015**, *48*, 933–937. [CrossRef]
7. Sanamdikar, S.T.; Hamde, S.T.; Asutkar, V.G. A literature review on arrhythmia analysis of ECG signal. *Int. Res. J. Eng. Technol.* **2015**, *2*, 307–312.
8. Rashed-Al-Mahfuz, M.; Moni, M.A.; Lio, P.; Islam, S.M.S.; Berkovsky, S.; Khushi, M.; Quinn, J.M. Deep convolutional neural networks based on ECG beats classification to diagnose cardiovascular conditions. *Biomed. Eng. Lett.* **2021**, *11*, 147–162. [CrossRef]
9. Swain, S.S.; Patra, D.; Singh, Y.O. Automated detection of myocardial infarction in ECG using modified Stockwell transform and phase distribution pattern from time-frequency analysis. *Biocybern. Biomed. Eng.* **2020**, *40*, 1174–1189. [CrossRef]
10. Lekhal, R.; Zidelmal, Z.; Ould-Abdesslam, D. Optimized time–frequency features and semi-supervised SVM to heartbeat classification. *Signal Image Video Process.* **2020**, *14*, 1471–1478. [CrossRef]
11. Kayikcioglu, I.; Akdeniz, F.; Köse, C.; Kayikcioglu, T. Time-frequency approach to ECG classification of myocardial infarction. *Comput. Electr. Eng.* **2020**, *84*, 106621. [CrossRef]
12. Kłosowski, G.; Rymarczyk, T.; Wójcik, D.; Skowron, S.; Cieplak, T.; Adamkiewicz, P. The use of time-frequency moments as inputs of lstm network for ecg signal classification. *Electronics* **2020**, *9*, 1452. [CrossRef]
13. Wang, T.; Lu, C.; Sun, Y.; Yang, M.; Liu, C.; Ou, C. Automatic ECG classification using continuous wavelet transform and convolutional neural network. *Entropy* **2021**, *23*, 119. [CrossRef]
14. Hussein, A.F.; Hashim, S.J.; Rokhani, F.Z.; Wan Adnan, W.A. An automated high-accuracy detection scheme for myocardial ischemia based on multi-lead long-interval ECG and Choi-Williams time-frequency analysis incorporating a multi-class SVM classifier. *Sensors* **2021**, *21*, 2311. [CrossRef] [PubMed]
15. Alqudah, A.M.; Alqudah, A. Deep learning for single-lead ECG beat arrhythmia-type detection using novel iris spectrogram representation. *Soft Comput.* **2022**, *26*, 1123–1139. [CrossRef]
16. Farag, M.M. A Self-Contained STFT CNN for ECG Classification and Arrhythmia Detection at the Edge. *IEEE Access* **2022**, *10*, 94469–94486. [CrossRef]
17. "ECG-ID Database v1.0.0." ECG-ID Database v1.0.0. Available online: www.physionet.org/content/ecgiddb/1.0.0 (accessed on 19 November 2022).
18. "PhysioNet/CinC Challenge 2015: Training Sets." PhysioNet/CinC Challenge 2015: Training Sets. Available online: archive.physionet.org/physiobank/database/challenge/2015 (accessed on 19 November 2022).
19. Berkaya, S.K.; Uysal, A.K.; Gunal, E.S.; Ergin, S.; Gunal, S.; Gulmezoglu, M.B. A survey on ECG analysis. *Biomed. Signal Process. Control.* **2018**, *43*, 216–235. [CrossRef]
20. Wahabi, S.; Pouryayevali, S.; Hari, S.; Hatzinakos, D. On evaluating ECG biometric systems: Session-dependence and body posture. *IEEE Trans. Inf. Secur.* **2014**, *9*, 2002–2013. [CrossRef]
21. Kamath, C. ECG beat classification using features extracted from Teager energy functions in time and frequency domains. *IET Signal Process.* **2011**, *5*, 575–581. [CrossRef]
22. Ebrahimpour, R.; Sadeghnejad, N.; Sajedin, A.; Mohammadi, N. Electrocardiogram beat classification via coupled boosting by filtering and preloaded mixture of experts. *Neural Comput. Appl.* **2013**, *23*, 1169–1178. [CrossRef]
23. Rai, H.M.; Trivedi, A.; Shukla, S. ECG signal processing for abnormalities detection using multi-resolution wavelet transform and Artificial Neural Network classifier. *Measurement* **2013**, *46*, 3238–3246. [CrossRef]
24. Donoso, F.I.; Figueroa, R.L.; Lecannelier, E.A.; Pino, E.J.; Rojas, A.J. Atrial activity selection for atrial fibrillation ECG recordings. *Comput. Biol. Med.* **2013**, *43*, 1628–1636. [CrossRef] [PubMed]
25. Chaitanya, N.K.; Radhakrishnan, A.; Reddy, G.R.; Manikandan, M.S. A simple and robust QRS detection algorithm for wireless medical body area network. In Proceedings of the 2011 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), Udaipur, India, 22–24 April 2011; pp. 153–158.
26. Jones, D.L.; Touvannas, J.S.; Lander, P.; Albert, D.E. Advanced time-frequency methods for signal-averaged ECG analysis. *J. Electrocardiol.* **1992**, *25*, 188–194. [CrossRef] [PubMed]
27. Zhivomirov, H. A novel visual representation of the signals in the time-frequency domain. *UPB Sci. Bull. Ser. C Electr. Eng. Comput. Sci.* **2018**, *80*, 75–84.
28. Gentry, C.; Liao, C.T.; You, W.; Ryan, S.A.; Varner, B.A.; Shi, X.; Cating-Subramanian, E. Super-resolved time–frequency measurements of coupled phonon dynamics in a, 2D quantum material. *Sci. Rep.* **2022**, *12*, 1–8. [CrossRef] [PubMed]
29. "Digital Signal Processing System Design." Digital Signal Processing System Design | ScienceDirect. Available online: www.sciencedirect.com:5070/book/9780123744906/digital-signal-processing-system-design (accessed on 30 November 2022).
30. Byeon, Y.H.; Pan, S.B.; Kwak, K.C. Intelligent deep models based on scalograms of electrocardiogram signals for biometrics. *Sensors* **2019**, *19*, 935. [CrossRef] [PubMed]
31. Analysis Mathematical Physics by Triebel—AbeBooks. Available online: https://www.abebooks.com/book-search/title/analysis-mathematiCal-physics/Author/Triebel/ (accessed on 29 November 2022).
32. "Spectrogram Visualization with Matlab." Spectrogram Visualization with Matlab–File Exchange—MATLAB Central. Available online: www.mathworks.com/matlabcentral/fileexchange/64882-spectrogram-visualization-with-matlab (accessed on 3 December 2018).

33. Ren, Z.; Qian, K.; Zhang, Z.; Pandit, V.; Baird, A.; Schuller, B. Deep scalogram representations for acoustic scene classification. *IEEE/CAA J. Autom. Sin.* **2018**, *5*, 662–669. [CrossRef]
34. Mourad, K.; Fethi, B.R. Efficient automatic detection of QRS complexes in ECG signal based on reverse biorthogonal wavelet decomposition and nonlinear filtering. *Measurement* **2016**, *94*, 663–670. [CrossRef]
35. Lee, J.N.; Kwak, K.C. Personal identification using a robust eigen ECG network based on time-frequency representations of ECG signals. *IEEE Access* **2019**, *7*, 48392–48404. [CrossRef]
36. Alquran, H.; Alsalatie, M.; Mustafa, W.A.; Abdi, R.A.; Ismail, A.R. Cervical Net: A Novel Cervical Cancer Classification Using Feature Fusion. *Bioengineering* **2022**, *9*, 578. [CrossRef]
37. Alquran, H.; Al-Issa, Y.; Alsalatie, M.; Mustafa, W.A.; Qasmieh, I.A.; Zyout, A. Intelligent Diagnosis and Classification of Keratitis. *Diagnostics* **2022**, *12*, 1344. [CrossRef] [PubMed]
38. Alawneh, K.; Alquran, H.; Alsalatie, M.; Mustafa, W.A.; Al-Issa, Y.; Alqudah, A.; Badarneh, A. LiverNet: Diagnosis of Liver Tumors in Human CT Images. *Appl. Sci.* **2022**, *12*, 5501. [CrossRef]

diagnostics

MDPI

Article

H. pylori Related Atrophic Gastritis Detection Using Enhanced Convolution Neural Network (CNN) Learner

Yasmin Mohd Yacob [1,2], Hiam Alquran [3,4], Wan Azani Mustafa [2,5,*], Mohammed Alsalatie [6], Harsa Amylia Mat Sakim [7] and Muhamad Safiih Lola [8]

[1] Faculty of Electronic Engineering & Technology, Pauh Putra Campus, Universiti Malaysia Perlis (UniMAP), Arau 02600, Perlis, Malaysia
[2] Centre of Excellence for Advanced Computing, Pauh Putra Campus, Universiti Malaysia Perlis (UniMAP), Arau 02600, Perlis, Malaysia
[3] Department of Biomedical Systems and Informatics Engineering, Yarmouk University, Irbid 21163, Jordan
[4] Department of Biomedical Engineering, Jordan University of Science and Technology, Irbid 22110, Jordan
[5] Faculty of Electrical Engineering & Technology, Pauh Putra Campus, Universiti Malaysia Perlis (UniMAP), Arau 02600, Perlis, Malaysia
[6] King Hussein Medical Center, Royal Jordanian Medical Service, The Institute of Biomedical Technology, Amman 11855, Jordan
[7] School of Electrical and Electronic Engineering, Engineering Campus, Universiti Sains Malaysia, Nibong Tebal 11800, Penang, Malaysia
[8] Faculty of Ocean Engineering Technology and Informatics, Universiti Malaysia Terengganu, Kuala Terengganu 21030, Terengganu, Malaysia
* Correspondence: wanazani@unimap.edu.my

Abstract: Atrophic gastritis (AG) is commonly caused by the infection of the *Helicobacter pylori* (*H. pylori*) bacteria. If untreated, AG may develop into a chronic condition leading to gastric cancer, which is deemed to be the third primary cause of cancer-related deaths worldwide. Precursory detection of AG is crucial to avoid such cases. This work focuses on *H. pylori*-associated infection located at the gastric antrum, where the classification is of binary classes of normal versus atrophic gastritis. Existing work developed the Deep Convolution Neural Network (DCNN) of GoogLeNet with 22 layers of the pre-trained model. Another study employed GoogLeNet based on the Inception Module, fast and robust fuzzy C-means (FRFCM), and simple linear iterative clustering (SLIC) superpixel algorithms to identify gastric disease. GoogLeNet with Caffe framework and ResNet-50 are machine learners that detect *H. pylori* infection. Nonetheless, the accuracy may become abundant as the network depth increases. An upgrade to the current standards method is highly anticipated to avoid untreated and inaccurate diagnoses that may lead to chronic AG. The proposed work incorporates improved techniques revolving within DCNN with pooling as pre-trained models and channel shuffle to assist streams of information across feature channels to ease the training of networks for deeper CNN. In addition, Canonical Correlation Analysis (CCA) feature fusion method and ReliefF feature selection approaches are intended to revamp the combined techniques. CCA models the relationship between the two data sets of significant features generated by pre-trained ShuffleNet. ReliefF reduces and selects essential features from CCA and is classified using the Generalized Additive Model (GAM). It is believed the extended work is justified with a 98.2% testing accuracy reading, thus providing an accurate diagnosis of normal versus atrophic gastritis.

Keywords: *H. pylori*; atrophic gastritis; deep learning; convolution neural network; ShuffleNet; feature fusion; Canonical Correlation Analysis; ReliefF; generalized additive model

Citation: Yacob, Y.M.; Alquran, H.; Mustafa, W.A.; Alsalatie, M.; Sakim, H.A.M.; Lola, M.S. *H. pylori* Related Atrophic Gastritis Detection Using Enhanced Convolution Neural Network (CNN) Learner. *Diagnostics* **2023**, *13*, 336. https://doi.org/10.3390/diagnostics13030336

Academic Editor: Costin Teodor Streba

Received: 21 November 2022
Revised: 9 January 2023
Accepted: 12 January 2023
Published: 17 January 2023

1. Introduction

Artificial intelligence (AI) has been important in steering various medical imaging-related problems. Although consultancy or diagnosis from medical practitioners has been the gold standard, computer-aided systems play a key role in assisting the limited number

of busy-scheduled medical practitioners. Moreover, enhancing medical imaging techniques via AI is useful for screenings, medical precision, and risk assessment.

Otherwise, AI implementation in medical imaging enables physicians to detect medical-related conditions faster, thus promoting early intervention. AI enables colorectal cancer to be detected and diagnosed by analyzing tissue scans [1]. Furthermore, AI can identify fractures [2], neurological diseases [3], and thoracic complications [4]. Recent works have highlighted that AI may be better or at least on par with pathologists. It is observed that AI can be a tool to assist pathologists in keeping up with the rising demands for expert services. Despite pathologists regularly assessing histopathology images, their average workload has risen significantly, leading to unintended misdiagnosis [5]. The fusion of AI into medical imaging also enhances precision medicine. Outside of this, a machine learning tool can recognize types of cancer [6] and predict patient survival rates, resulting in healthcare providing better treatment [7]. In addition, AI can be a tool to predict potential risks for the future, such as the prediction of heart attacks [8].

Various studies that employ machine learning are also significant for sparking further applied or improved works. Weis et al. developed unsupervised segmentation via Linear Combination [9]. Beneficial works are segmentation in medical imaging [10,11] and studies related to body parts [12,13] that employed deep learning. Skin-lesion segmentation implemented using hybrid metaheuristics machine learner has also been studied [14]. Lv et al. highlighted speech and gesture recognition and natural language processing studies implemented using deep learning [15]. Face image identification has been used to detect fake faces via Xception, which is a variation of deep learning [16]. There has also been work on disease diagnosis implemented using Convolution Neural Network (CNN) [17,18]. Recognition of different types of skin lesions using Fast Random Forest (FRF) learner has been beneficial [19]. Massaro et al. performed comparative studies via Discrete Fourier Transform (DCT), K-Means clustering, and Long-Short Term Memory (LSTM) to determine assembled tires defects [20]. Mohimont et al. estimated crop yielding via machine learning and deep learning [21]. Massaro et al. identified water leaks in pipelines using infrared thermography (IRT), CNN-LSTM, and image filtering algorithm [22].

Similarly, AI-based research impacts significantly on recognizing internal gastric infection. Untreated *Helicobacter pylori* (*H. pylori*) gastric bacterial infection may lead to gastric cancer, among other complex morbidities. AG, which is mostly caused by *H. pylori* bacteria, induces chronic inflammation of gastric mucosa leading to the loss of glands in the stomach lining. Additionally, *H. pylori*-associated AG increases the risk of developing stomach cancer. Therefore, preliminary detection of AG is crucial.

To date, many meta-analyses have focused on the application of AI-based *H. pylori* diagnosis [23–25] and gastric disease [26,27]. Five studies [28–32] developed an AI algorithm based on Convolution Neural Network (CNN) to diagnose *H. pylori* infection. Studies [33,34] have employed AI to diagnose gastric disease and deep Recurrent Neural Network (RCNN) to diagnose AG progression to gastric cancer [35]. In addition, Support Vector Machine (SVM) algorithms have also been employed to recognize gastric disease [36,37] and *H. pylori* infection in the gut [38]. Yet another work with biomarkers features has used statistical analysis to detect AG [39].

This work acknowledged Pecere et al.'s [25] critique that many studies focus on selected populations that may experience spectrum bias or class imbalance. Pecere et al. [25] argued some studies have tendencies to employ CNN regardless of other available machine learners mentioned in Mohan et al. [24]. To date, the pooled accuracy, sensitivity, and specificity of AI in the diagnosis of gastrointestinal disease is 87.1% (95% CI 81.8–91.1), 86.3% (95% CI 80.4–90.6), and 87.1% (95% CI 80.5–91.7), respectively [25]. The recognition rates of the three measures indicated were between 86.3% and 87.1%, demonstrating an improvement in AI-based AG-related detection. This is still acceptable even when some classifiers, such CNN, do not satisfy the recognition rates in research connected to AG.

The essential issue is to find a better machine learner to improve the standard of diagnostic results. Hence, an accurate, fast, and reliable diagnosis is needed to treat

H. pylori infection early. This is important in bacteria transmission inhibition, which promotes a longer lifespan.

2. Related Works

Recent trends have a growing interest in incorporating artificial intelligence (AI), medical diagnosis, and medical imaging related to the area in gastric. Getting a correct diagnosis with the assistance of a machine learner leads to a positive response because it is consistent and less likely to have inter-reader variability. Similar to other medical conditions, early detection of *Helicobacter pylori* (*H. pylori*) infection in human gastric is essential to acquire treatment and suppress the progression of this bacteria. Note that *H. pylori* cause not only gastritis, but untreated infections may also turn into chronic gastritis that leads to gastro-duodenal ulcer, gastric neoplasia, and gastric cancer [24].

This work highlights AI-based studies to detect *H. pylori*-related AG and other types of conditions in the gut. For example, Takiyama et al. [40] and Wu et al. [41] both examined gastric cancer, particularly to identify the anatomic multi-location of cancer. Wu et al. found an accuracy of 90% for detecting gastric cancer location-wise, and Takiyama showed an Area Under Curve (AUC) of 0.99, a sensitivity of 96.9%, and a specificity of 98.5%, respectively. On the other hand, Kanesaka et al. [37] detected 66 abnormal conditions out of 126 images during training and 61 abnormalities out of 81 images during testing. Furthermore, the study employed a Support Vector Machine (SVM) to detect early gastric cancer with an accuracy of 96.3%, a sensitivity of 96.7%, and a specificity of 95%. This result is quite impressive compared to works that employed CNN that mostly report around 85% to 93.8% accuracy.

In 2018, Hirasawa et al. [42] identified gastric cancer using Single Shot MultiBox Detector (SSD) with a sensitivity of 92.2% via 13,584 abnormalities during training and 2.296 abnormalities during testing out of 51,558 images. Another work by Ishioka et al. [43] also detected gastric cancer via SSD with a sensitivity of 94.1%. However, unlike Li et al. [44], Horuichi et al. [45] and Zhu et al. [46] both managed to detect gastric cancer with an accuracy of 85.3% and 89.16%, respectively. Meanwhile, Li et al. scored 90.91% accuracy in identifying gastric cancer via CNN Inception-V3. The other two studies also employed CNN, GoogLeNet, and ResNet, respectively.

Subsequent works describe studies related to machine learning detecting *H. pylori* infection in the gastric antrum. For example, Shichijo et al. employed Deep CNN (DCNN) of GoogLeNet with 22 layers of the pre-trained model that learned generic image features of *H. pylori*-infected cases. The AUC was 0.89, while sensitivity, specificity, and accuracy were 81.9%, 83.4%, and 83.1%, respectively [28]. Otherwise, Shichijo et al. collected a huge range of data with 32,208 images; 735 were *H. pylori* positive and 1015 were negative [28]. Another work by Shichijo et al. in 2019 [31] recorded an 80% accuracy (465/582) of negative diagnoses, 84% (147/174) eradication, and 48% (44/91) positive diagnoses. Similarly, the author employed a deep pre-trained CNN learner to identify the occurrence of *H. pylori* in the gastric antrum.

Alternatively, Itoh et al. [29] and Dong-Hyun et al. [33] both employed GoogLeNet to detect *H. pylori* conditions and gastric lesions, respectively. Dong-Hyun et al. employed GoogLeNet based on the Inception module, fast and robust fuzzy C-means (FRFCM), and simple linear iterative clustering (SLIC) superpixel algorithms applied for image segmentation during preprocessing of those images. Their work scored 0.85 and 0.87 for normal and abnormal lesions based on Receiver Operating Characteristic (ROC) curves. Meanwhile, Itoh et al. [29] achieved an AUC of 0.956, a sensitivity of 86.7%, and a specificity of 86.7%. Furthermore, Guimarães et al. [34] had 100 normal and 100 abnormal images of pre-cancerous detection during training and 40 normal and 30 abnormal images during testing. Accordingly, he achieved an AUC of 0.981, an accuracy of 92.9%, a sensitivity of 100%, and a specificity of 87.5% via ImageNet. The result was impressive because image samples were acquired from generic databases and were not unique. However, there is a possibility that the databases captured fewer complex conditions of features

from atrophic gastritis (AG). In 2018, Nakashima et al. [30] employed GoogLeNet with the Caffe framework to detect *H. pylori* infection that causes AG acquired from 222 enrolled participants. The study showed 105 were *H. pylori*-positive with an AUC reading of 0.97. Aside from this, Zheng et al. [32] achieved an AUC of 0.66, an accuracy of 93.8%, a sensitivity of 91.6%, and a specificity of 98.6%, respectively, via ResNet-50. The study highlighted the identification of AG-infected patients with samples of 1959 patients: 1507 out of 1959 (77%) patients were deemed as training data, 847 (56%) out of 1507 cases were infected with *H. pylori*, and 11,729 gastric images were assigned for training using image augmentation. On the other hand, 452 (23%) out of 1959 patients were deemed as testing data, 310 (69%) out of 452 cases were infected with *H. pylori*, and 3755 gastric images were assigned for testing using image augmentation [32]. It is acknowledged that studies that employ only CNN Inception-V3 score 90.91% accuracy [44]. Similarly, the SVM can achieve high accuracy at 96.3% [37] in identifying gastric cancer.

Subsequently, Alquran et al. [47–49] employed ResNet and ShuffleNet Version 2, which are suitable for cervical cytology and cornea ulcer. Nevertheless, gastric cancer has different unique features compared to AG, which are incomparable. Improved recognition methods applied to the detection of *H. pylori*-related AG per se are essential to improve the standards of the diagnostic methods which are currently being used. In addition, further work to detect and upgrade *H. pylori*-related to AG is compelling. The current accuracy stands at 92.9% via ImageNet. As mentioned earlier, however, the samples may be generic and less descriptive since they are public data sets. Furthermore, there is always room for improvement to upgrade the accuracy of AG detection, which currently stands at 93.8% via ResNet-50. Accurate AG detection may promote faster treatment in the early stages of *H. pylori* infection, saving patients from experiencing chronic conditions.

3. Materials and Methods

This section elaborates on the feature fusion, feature selection, and classifier employed in the study. The materials and characteristics of atrophic gastritis and normal gastric are presented. The proposed approach, which includes these methods and deep learning specification, is also addressed.

3.1. Feature Fusion Canonical Correlation Analysis (CCA)

Canonical Correlation Analysis (CCA) is a multivariate-based correlation statistical method. CCA assesses the complex relationships among variable sets that originate from different modalities and is an improved technique for determining the traditional correlation between two sets of variables. Besides this, CCA focuses on a dependence relationship, aiming to model the correlation between variables and the correlation between the two datasets [50]. There are two main concepts to comprehend CCA, which are canonical variables and canonical correlation. Note that canonical variables are linear combinations of the variables in the data sets. Since CCA focuses on correlations between two data sets, users define pairs of canonical variables from the left side of the data set and a second canonical variable from the right side. In the case of a different number of variables in the datasets, it maintains many pairs of canonical variables by determining linear combinations for features with the most correlation in two datasets. On the other hand, Principal Component Analysis (PCA) highlights the determination of linear combinations for the most variance features in one data set. Meanwhile, Multivariate Multiple Regression (MMR) is similar to CCA, whereby the prior method tries to find linear combinations that enable the model of correlations between two data sets. This work uses CCA to extract essential features because its characteristics demonstrate correlated features from two huge datasets [51–53]. The idea behind CCA is described in Figure 1.

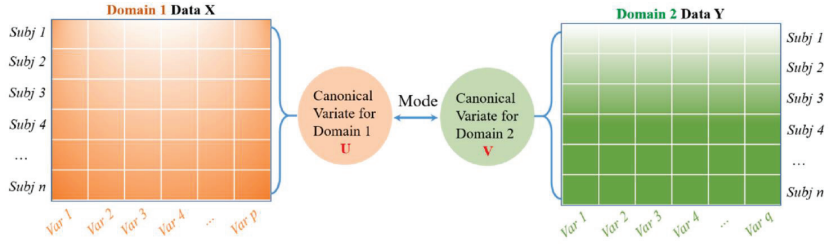


Figure 1. A general schematic for CCA to model the correlation between two feature datasets. Reprinted/adapted with permission frm Ref. [50]. Copyright from Li. et al., Mech. Syst. Signal Process; published by Elsevier Ltd., 2022.

3.2. ReliefF

Relief-F is an instance-based feature selection technique that assesses the feature ability to discriminate between samples rooted in different groups and the similarity between them. Relief-F is a filter-based feature selection method that computes the weighted value of each feature to determine the available essential features [54], as it carries the highest weight. If a feature value differs from its neighboring feature pair of a similar class, the weight decreases. In addition, Relief-F reduces the feature dimension by removing the negative-weighted value feature, which is suitable for a two-class scheme. Correspondingly, it is also useful for multi-class problems [55,56]. Hence, Relief-F was chosen in this work due to its ability to deal with noisy and missing data, a dependence feature that requires swift computation time and is fairly accurate.

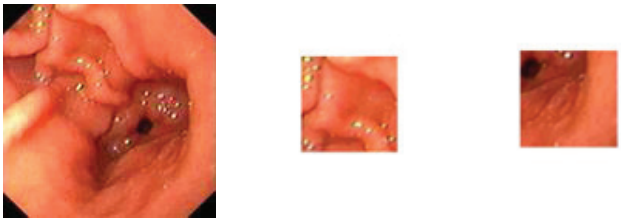
3.3. Generalized Additive Model (GAM)

A generalized additive model (GAM) is a generalized linear model whereby its response variable depends linearly on unknown smooth functions of some predictor variables, and interest focuses on inference about these smooth functions. Trevor Hastie and Robert Tibshirani developed GAMs to combine the characteristics of generalized linear models with additive models. They can be interpreted as the discriminative generalization of the Naive Bayes generative model. Additionally, GAM is a model that allows the linear model to learn nonlinear relationships. It assumes that instead of using simple weighted sums, it can use the sum of arbitrary functions. Besides, GAMs may overfit when there are too many explanatory variables or a small sample size. For cases where the sample size is too small, the overfitting problem can be solved by increasing the sampling [57]. This work employed GAM due to the ability of the learner to learn nonlinear relationships in the dataset.

3.4. Materials

The gastric images were acquired from the Endoscopy Unit, Hospital Universiti Sains Malaysia, Kubang Kerian, Kelantan. The images were captured from 20 patients where 12 were detected as having atrophic gastritis and 8 were classified as normal. The patients had undergone an endoscopy procedure performed by an endoscopy specialist. Each patient's hardcopy photo 4 by 6 inches in size was obtained from their medical report. The hardcopy photos were scanned in 300 dpi resolution and saved in softcopy form for the study. Thus, 12 images were categorized as atrophic gastritis (AG) and there were 8 normal gastric images. Due to the small-scale number of images available, each image was cropped into 20 regions of interest (ROI) of size 128 $\times$ 128 pixels. Therefore, the actual ROIs used in the classification were 244 images from AG and 160 images from normal gastric. The experiment was trained using 70% of the total data and tested on the remaining 30% of the available data. Thus, 282 mixed images of AG and normal gastric were employed for training. On the other hand, 122 mixed normal gastric and AG images were used for

testing. Figure 2 illustrates a sample of the normal gastric image, which was then cropped into 20 ROIs. Similarly, Figure 3 shows a sample of the AG image cropped into 20 ROIs.



(**A**) (**B**) (**C**)

Figure 2. Sample of the normal gastric image cropped into 20 ROIs; (**A**) normal gastric, (**B**) normal ROI_1, (**C**) normal ROI_20.

(**A**) (**B**) (**C**)

Figure 3. Sample of the atrophic gastric image cropped into 20 ROIs; (**A**) atrophic gastric, (**B**) atrophic ROI_1, (**C**) atrophic ROI_20.

3.5. Characteristics of Normal Gastric and Atrophic Gastric

The human stomach has a shield of mucus lining called the mucosa. The safe lining guards the stomach from the sturdy stomach acid that absorbs food. When the protective lining is damaged, the mucosa becomes inflamed, resulting in gastritis. It is indicative that *Helicobacter pylori* (*H. pylori*) is the most common bacteria that cause gastritis [58].

AG is a histopathological substance characterized by chronic inflammation of the gastric mucosa. The inflamed gastric mucosa experience loss of gastric glandular cells and is reinstated by intestinal-type epithelium, pyloric-type glands, and fibrous tissue. AG is frequently linked with *H. pylori* infection, unidentified environmental factors, and autoimmunity governed in the case of gastric glandular cells. Note that the bacteria disrupt the barrier of mucus that protects the human stomach lining from the acidic juices that help with digestion [58]. Hence, the infection will gradually destroy the cells in the stomach lining if untreated.

3.6. Proposed Method

This work focused on the detection of AG versus the normal gut. We highlighted machine learning improvement related to *H. pylori*-related AG infection in the gut. It was observed that previous studies employed GoogLeNet, pre-trained Deep Convolution Neural Networks (DCNN), ImageNet, and ResNet-50. Based on experimentation and analysis, this study used pre-trained ShuffleNet Version 1, CCA feature fusion, Relief-F feature selection method, and GAM classifier, as described in Figure 4.

The pre-trained ShuffleNet used the ReLu layer, which later produced 26,656 features and a global average pooling layer of 544. Subsequently, the high-dimensioned features were processed with feature fusion, known as CCA, resulting in 265 features. The features were further reduced to the 50 most relevant features using the ReliefF feature selection method. Finally, GAM was employed as a machine learner to perform binary classification to identify between atrophic and normal gastritis classes.

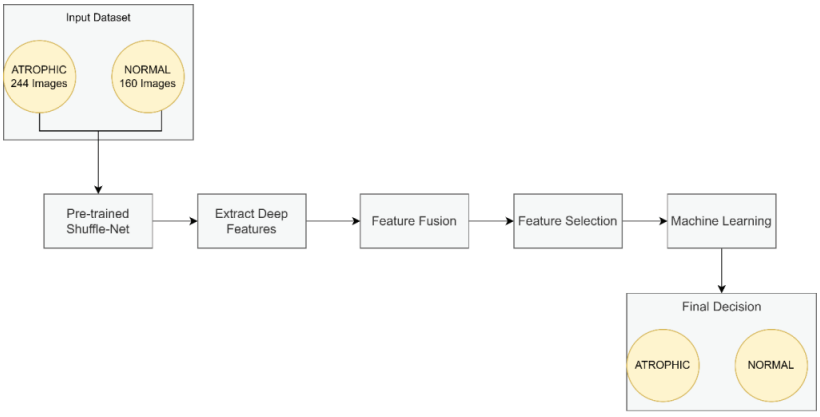


Figure 4. The proposed method for atrophic gastritis detection.

3.7. Specification of ShuffleNet

This work employed the pre-trained ShuffleNet model already implemented in MATLAB® Version 2021. The pre-trained ShuffleNet utilized the following hyperparameters setting as described in Table 1. It consisted of 171 layers, and employed the root means square propagation (RMSProp) optimizer, a mini-batch size of 64, sixty epochs, and a 0.0004 learning rate.

Table 1. Pre-trained ShuffleNet hyperparameters setting.

Hyperparameters Setting	
Optimizer	RMSprop
Number of Layers	171
Mini batch size	64
Maximum Epoch	60
Learning Rate	0.0004

The proposed ShuffleNet structure for AG and its summaries are presented in Figure 5 and Table 2. Table 2 describes the detailed structure of the model. All the images in the dataset were resized to 224 × 224 × 3 pixels to follow the standard image size suggested by ShuffleNet Version 1 network architecture. The model initiated with a group convolution-max pool stem and proceeded with three repeated stages before the finalized global pool-fully connected layer. Consequently, the block began with a 1 × 1 group convolution layer of filter size 112 × 112, kernel size 3 × 3, and stride 2 × 2. Then, the layer was processed via the ReLu layer and followed by an average pooling layer kernel size of 3 × 3 and stride 2 × 2. The initial group convolution did not have a shuffle layer. Thus, the output channel of the next stage was doubled into 56 × 56.

In the next stage, the filter size was 28 × 28 and stride 2 × 2 but not repeated. Subsequently, the average pooling layer featured a 28 × 28 filter size using stride 1 × 2 repeated three times. Finally, ReLu was the activated layer and was followed by batch normalization. Note that ShuffleNet was differentiated from other architectures via shuffle layers as this study had three channels due to colored images. First, the input tensor was reshaped to shuffle the channels from 224 × 224 × 3 channels to 224 × 224 × 3 groups × 3 channels divided by 3 groups. Then, the last two dimensions were permuted, and the input tensor was reshaped to the original shape.

Correspondingly, the next two stages had shuffle layers, in which the filter size in the group convolution layer remained 28 × 28 in Stage 3 with stride 2 × 2 but not repeated. The model applied the ReLu layer and batch normalization at this point. The average

pooling layer was of size 14 × 14 with stride 1 × 1, repeated seven times. Then, the channel layers were shuffled via permutation via 3 groups or 3 channels.

Stage 4 featured 7 × 7 maps in the grouped convolution layer with stride 2 × 2 and not repeated. After the layer was activated via ReLu and batch normalization, the average pooling size filter size remained the same: 7 × 7 but this time around with stride 1 × 1 and repeated three times.

The global average pool layer generated a 1 × 1 feature map with kernel size 7 × 7 that produced the best features, followed by a fully connected layer that applied a softmax layer to proceed with the classification layer with two neurons that depicted classes of normal and AG presence. This structure is described in Figure 5 and Table 2. The transfer learning strategy was performed in the last fully connected layer to be compatible with the number of intended classification classes.

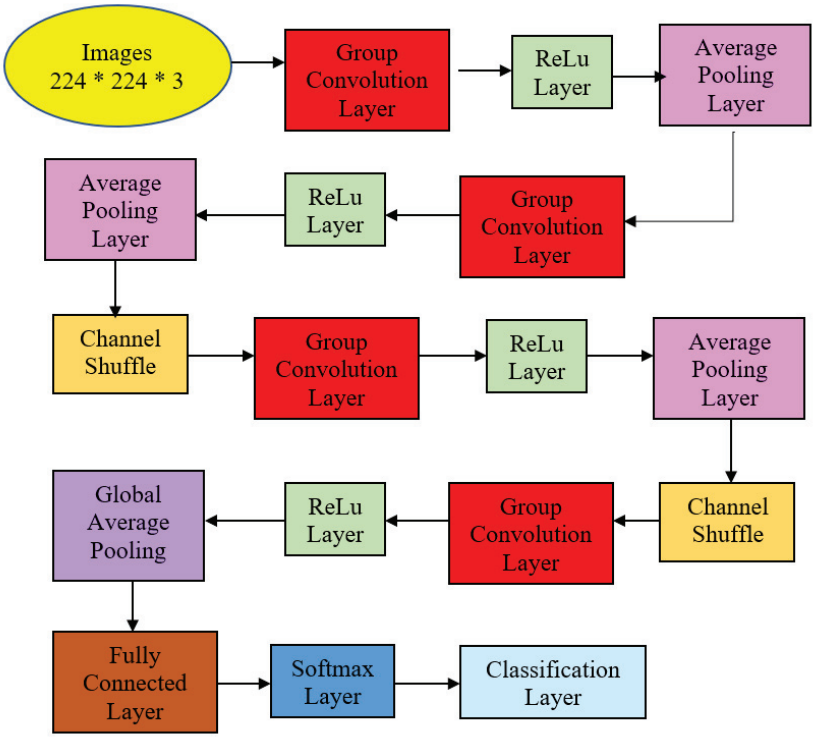


Figure 5. Atrophic Gastritis ShuffleNet structure.

Table 2. Structure summaries for Atrophic Gastritis ShuffleNet.

Layer	Information
Input Layer	Size: 224 × 224 × 3
Group Conv1	Number of Filters: 112 Kernel Size: 3 × 3 Stride: 2 × 2 Padding: 0
Activation Layer	ReLu
Pooling Layer	Type: Average Pooling Kernel Size: 3 × 3 Stride: 2 × 2 Padding: 0 Number of Groups: 3

Table 2. *Cont.*

Layer	Information
Group Convolution Layer	Number of Filters: 56 Kernel Size: 3 × 3 Padding: 0
Activation Layer	ReLu
Pooling Layer	Type: Average Pooling Kernel Size: 3 × 3 Stride: 1 × 1 Padding: 0
	Shuffle layer
Group Convolution Layer	Number of Filters: 28 Kernel Size: 3 × 3 Stride: 1 × 1 Padding: 0
Activation Layer	ReLu
	Shuffle layer
Group Convolution Layer	Number of Groups: 3 Number of Filters: 11 Kernel Size: 3 × 3 Stride: 1 × 1 Padding: 0
Activation Layer	ReLu
Fully Connected Layer	Type: Global Average Pooling 2 neurons
Softmax Layer	
Classification Layer	

4. Results and Discussion

Figure 6 shows the confusion matrix with results analyzed from the testing data executed via pre-trained ShuffleNet. Figure 6 demonstrates the results for the binary classes of atrophic gastritis (AG) and normal gut. A total of 52 out of 67 were correctly classified as AG, meaning 77.6% were correctly classified as AG. Whereas the remaining 15 samples were misclassified, meaning 22.4% were misclassified as normal gut. All 48 out of 48 normal gut images were correctly classified, meaning that 100% were classified correctly. The overall accuracy for the pre-trained ShuffleNet was 87.0% for binary classes of AG and normal classes. The misclassification rate for the pre-trained ShuffleNet was 13.0%.

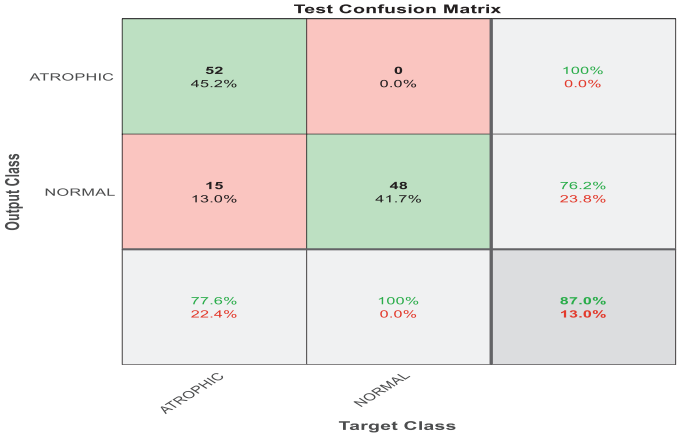


Figure 6. Confusion matrix with results analyzed from the testing data executed via pre-trained ShuffleNet.

The obtained results using deep learning structure were less prominent in distinguishing between atrophic and normal cases. Therefore, a new strategy was proposed to enhance the pre-trained classification result. The proposed method focused on the utilization of feature engineering techniques. The feature engineering enhancement focused on fusion between the ReLU and global average pooling layer. The resultant new features were passed to the feature selection algorithm to nominate the most relevant representative descriptors. Finally, these features were delivered to the generalized additive model (GAM) that distinguished between two classes: atrophic or normal. This hybrid model based on feature engineering clarified the prominent results. Figure 7 shows the confusion matrix with results analyzed from the testing data executed via the hybrid model.

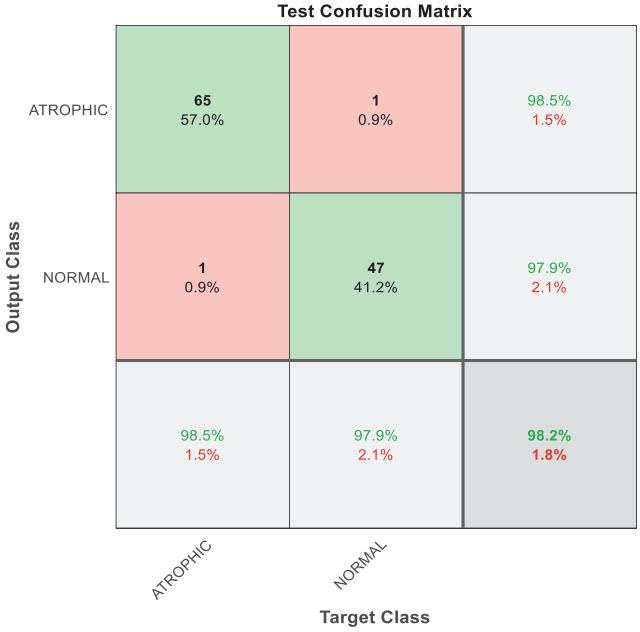


Figure 7. Confusion matrix with results analyzed from the testing data executed via hybrid model and classified by GAM.

As a result of the techniques that constitute the hybrid model, the classification result improved, with 65 out of 66 instances correctly detected as AG, constituting 98.5% accuracy; the remaining one sample was misclassified, meaning that 1.5% was misclassified as normal gut. Furthermore, 47 out of 48 cases of normal gastric images were accurately identified, i.e., 97.9%; the remaining one sample was misclassified, meaning 2.1% were misclassified as AG. The overall performance of the hybrid model was outstanding at 98.2% accuracy for binary classes of AG and normal classes, as depicted in Figure 7.

Even though the result from the pre-trained ShuffleNet was less prominent, it generated 26,656 significant features via the ReLu layer and 544 significant features from the global average pooling layer, as mentioned in Section 3.6. CCA successfully modeled the relationship between the two data sets of significant features from the ReLu layer and the global average pooling layer respectively. CCA focused on finding linear combinations that most accounted for the correlation of the two datasets. To some extent, ReliefF selected essential features by reducing the feature dimension of the negative-weighted value feature. The features may have consisted of non-linearity characteristics because they were extracted from huge features, thus GAM performed as a better learner. The strength of GAM is that it allows linear modeling to identify the nonlinear relationship among its features.

Table 3 describes the results based on precision and recall, also known as sensitivity, specificity, and accuracy matrices for the pre-trained ShuffleNet and the proposed method. Similarly, Table 4 shows the F1-score results for the mentioned models. Based on the results in Table 4, although the precision and specificity reading of the proposed method compared to the pre-trained network slightly dipped to 98.5% and 97.9% respectively, the recall or sensitivity, accuracy, and F1-score displayed significant improvement in the proposed method. Outside of this, the recall or sensitivity greatly increased from 77.6% in the pre-trained network to 98.5% in the proposed method. Subsequently, accuracy improved from 87.0% in the pre-trained network to 98.2% in the proposed method. Moreover, the F1-score also demonstrated a significant leap from 0.87 in the pre-trained model to 0.98 in the proposed method. These readings indicate that the proposed method can determine true positive cases of AG. Similarly, the proposed method can also predict the cases of atrophic and normal gastric correctly. On a scale from 0 to 1, 1 being the best, the proposed method successfully classified each observation into the correct atrophic and normal classes.

Table 3. Results based on precision, recall, specificity, and accuracy matrices for the pre-trained ShuffleNet and proposed method.

	Pre-Trained ShuffleNet (%)	Proposed Method (%)
Precision	100.0	98.5
Recall = Sensitivity	77.6	98.5
Specificity	100.0	97.9
Accuracy	87.0	98.2

Table 4. F1-score result for the pre-trained ShuffleNet and proposed method.

	Pre-Trained ShuffleNet	Proposed Method
FI-score	0.87	0.98

The proposed approach was compared with literature that worked on the same classification problem for endogastric diagnosis images. Table 5 illustrates their methods, type of infection, number of classes, and achievement. The comparison is based on the type of study, of either atrophic gastritis or non-atrophic gastritis. Similar studies of atrophic gastritis were compared based on performance in terms of specificity, sensitivity, and accuracy.

Table 5. Comparison of the proposed approach with literature on similar endogastric diagnosis classification problems.

Study	Method	Type	Classes	Accuracy
Shichijo S. et al. (2017) [28]	Deep CNN, 22 layers, pre-trained GoogLeNet via ImageNet	*H. pylori*	2	AUC = 0.89, Specificity = 81.9%, Sensitivity = 83.4%, Accuracy = 83.1%
Takiyama H. et al. (2018) [40]	GoogLeNet	Anatomical location of the upper digestive tract		AUC = 0.99, Specificity = 98.5%, Sensitivity = 96.9%, Accuracy = N/A
Itoh et al. (2018) [29]	GoogLeNet	*H. pylori*	2	AUC = 0.956, Specificity = 86.7%, Sensitivity = 86.7%, Accuracy = N/A
Hirasawa et al. (2018) [42]	SSD	Gastric cancer	2	AUC = N/A, Specificity = N/A, Sensitivity = 92.2%, Accuracy = N/A

Table 5. *Cont.*

Study	Method	Type	Classes	Accuracy
Kanesaka et al. (2018) [37]	SVM	Early gastric cancer	2	AUC = N/A, Specificity = 95%, Sensitivity = 96.7%, Accuracy = 96.3
Nakashima et al. (2018) [30]	GoogLeNet; Caffe	*H. pylori*	2	AUC = 0.97, Specificity = N/A, Sensitivity = NA, Accuracy = NA,
Shichijo S. et al. (2019) [31]	Pre-trained DCNN	*H. pylori*	2	Accuracy = 80% (negative), 84% (eradicated), 48% (positive)
Dong-hyun et al. (2019) [33]	GoogLeNet; Inception module	Gastric disease	2	ROC = 0.85 (normal) and 0.87 (abnormal)
Zheng et al. (2019) [32]	ResNet-50	*H. pylori*	2	AUC = 0.66, Specificity = 98.6%, Sensitivity = 91.6%, Accuracy = 93.8
Wu et al. (2019) [41]	VGG ResNet	26 locations of gastric cancer		AUC = N/A, Acc = 90, Sensitivity = N/A, Specificity = N/A
Ishioka et al. (2019) [43]	SSD	Gastric cancer	2	AUC = N/A, Specificity = N/A, Sensitivity = 94.1%, Accuracy = N/A
Horiuchi et al. (2020) [45]	GoogLeNet	Gastric cancer vs. gastritis	2	AUC = 0.85, Specificity = 71.0%, Sensitivity= 95.4%, Accuracy= 85.3%
Zhu et al. (2019) [46]	ResNet	Gastric cancer	2	AUC = 0.94, Specificity = 95.56%, Sensitivity= 76.47%, Accuracy= 89.16%
Guimarães P et al. (2020) [34]	ImageNET	*H. pylori*	2	AUC = 0.981, Specificity = 87.5%, Sensitivity = 100%, Accuracy = 92.9%
Li et al. (2020) [44]	Inception-V3	*H. pylori*	2	AUC = N/A, Specificity = 90.64%, Sensitivity = 91.18%, Accuracy = 90.9%
Our study	Pre-trained CNN: ShuffleNet version 1	*H. pylori*	2	Test: 98.2%; Atrophic sensitivity (98.5%), Normal sensitivity (97.9%)

Based on the comparison of our and the existing works in Table 5, some studies [37,40–42] were not comparable with our study because their work focussed on either gastric cancer or determining the location of infection. Furthermore, some studies [30,31,33] had insufficient information related to the results presented here and thus were incomparable to this work. It is evident from Table 5 that study [28] achieved specificity, sensitivity, and accuracy in the range of less than 83.5%. Meanwhile, study [44] achieved much a better result compared to [28]; however the specificity, sensitivity, and accuracy were much lower than 91.5%; thus this work was insignificant to this research, which achieved specificity, sensitivity, and accuracy all above 97%. References [28,44] achieved a less significant result compared to this work because they employed GoogleNet with Inception Module, which did not employ group convolution and had fixed convolution sizes for each layer, thus missing significant features. On the other hand, the result of this study is much better because it employed ShuffeNet, which performed group convolution and each output channel related to the input channels outside the group. As for study [34], even though their sensitivity reading was 100% compared to the this study's sensitivity reading of 98.5%, its specificity was 87.5% and its accuracy was 92.9%, both much lower than the readings in this study. Unlike this in this study, it is suggested that study [34],

which employed ImageNet, was actually a large-scale dataset rather than deep learning network itself. Study [32], which employed ResNet-50, achieved a slightly better specificity of 98.6% compared to that of this work with its specificity reading of 98.5%. Nevertheless, study [32]'s sensitivity reading of 91.6% and accuracy reading of 93.8% were both much lower than this study's readings of 98.5% and 98.2% respectively. Unlike in this study, study [32], which employed ResNet-50, suffered a degradation problem when the network became deeper. On the other hand, study [29], which employed GoogleNet, achieved a specificity of 98.5%, slightly better than the 97.9% reading in this study. Nevertheless, study [32]'s sensitivity and accuracy fell short at 96.9% and were not applicable respectively compared to the readings in this study. This study's sensitivity reading was at 98.5% and its accuracy was at 98.2%. It is evident this study's accuracy, at 98.2%, was higher than studies [28,32,34,44], which all had readings are lower than 94%.

Compared to existing works, this study had the best accuracy because it employed a hybrid model that included pre-trained ShuffleNet. The pre-trained ShuffleNet performed group convolution and each output channel was related to the input channels outside the group due to the shuffle mechanism. This mechanism strengthened the blocks of information representation, thus generating essential features. In the hybrid model, the essential features were channeled to perform a feature engineering technique besides learning with GAM. The feature engineering technique was supported by the Canonical Correlation Analysis (CCA) feature fusion approach and ReliefF feature selection method that successfully extracted relevant and essential features before the finalized classification method was executed. Indeed, CCA was supported in one of the stages whereby the method took a linear combinations approach that enabled the model correlation between two feature datasets. Furthermore, during the classification process, GAM successfully learned nonlinear relationships in the dataset and managed the small sample size problem by generating many regions of interest (ROIs).

5. Conclusions

To sum up, some previous studies have highlighted the detection of gastric cancer, yet studies to detect atrophic gastritis (AG) are equally important. This is because *Helicobacter pylori* (*H. pylori*)-related AG is the initial indicator of problematic gut condition. Furthermore, undetected and untreated AG may worsen into a chronic condition and eventually develop into gastric cancer.

This study proposed a hybrid model which included a pre-trained ShuffleNet that shuffled channels in a group-wise manner. The pre-trained ShuffleNet employed a cheap approach that later produced 26,656 features from the ReLu layer and 544 rich features from the global average pooling layer. Although accuracy at the pre-trained ShuffleNet stage was 87% and less prone to achieve an ideal model with an F1-score reading of 0.87, the rich and essential features were successfully identified. Extended approaches generated more relevant features from the deep features; namely feature fusion Canonical Correlation Analysis (CCA), which produced 265 features, and ReliefF feature selection, which generated 50 relevant features from the deep features. This was evident from the huge improvement to 98.2% accuracy and an almost ideal model based on an F1-score value of 0.98.

As mentioned earlier, the proposed hybrid model inclusive of pre-trained ShuffleNet achieved an accuracy of 98.2%, which was higher than similar studies that attempted to detect atrophic gastritis in binary classes of atrophic gastritis versus normal gastric condition. Unlike the method proposed in this study, some of the existing works employed a basic deep learning approach without an extended approach, which resulted in much lower accuracy and fewer chances to create an ideal model to detect atrophic gastritis or normal gastric condition.

Through endless efforts by researchers in medicine and artificial intelligence (AI) via the proposed hybrid model, the physician can detect the occurrence of AG as an initial step

to screen the *H. pylori* infection in the human stomach, thus reducing the possibility of a further chronic condition in the gastric.

Author Contributions: Conceptualization, Y.M.Y. and H.A.M.S.; Methodology, H.A. and M.A.; Writing—Y.M.Y. and H.A.; Writing—review and editing, H.A. and H.A.M.S.; Project administration, W.A.M.; Funding acquisition, M.S.L. and W.A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no specific external funding. However, thank you to Wan Azani Mustafa for the funding arrangement and sponsorship.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: This study would like to thank Lee Yeong Yeh, Aizal, and Nazri from the Endoscopy Unit, Hospital Universiti Sains Malaysia, Kubang Kerian, Kelantan, for the endoscopy images acquired during postgraduate work.

Conflicts of Interest: The authors declare no conflict of interest. The authors identify and declare no personal circumstances or interests that may be perceived as inappropriately influencing the representation or interpretation of the reported research results. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Goyal, H.; Mann, R.; Gandhi, Z.; Perisetti, A.; Ali, A.; Aman Ali, K.; Sharma, N.; Saligram, S.; Tharian, B.; Inamdar, S. Scope of Artificial Intelligence in Screening and Diagnosis of Colorectal Cancer. *J. Clin. Med.* **2020**, *9*, 3313. [CrossRef] [PubMed]
2. Kuo, R.Y.; Harrison, C.; Curran, T.A.; Jones, B.; Freethy, A.; Cussons, D.; Stewart, M.; Collins, G.S.; Furniss, D. Artificial Intelligence in Fracture Detection: A Systematic Review and Meta-Analysis. *Radiology* **2022**, *304*, 50–62. [CrossRef] [PubMed]
3. Segato, A.; Marzullo, A.; Calimeri, F.; De Momi, E. Artificial intelligence for brain diseases: A systematic review. *APL Bioeng.* **2020**, *4*, 041503. [CrossRef] [PubMed]
4. Bellini, V.; Valente, M.; Del Rio, P.; Bignami, E. Artificial intelligence in thoracic surgery: A narrative review. *J. Thorac.* **2021**, *13*, 6963–6975. [CrossRef]
5. Tizhoosh, H.R.; Pantanowitz, L. Artificial Intelligence and Digital Pathology: Challenges and Opportunities. *J. Pathol. Inform.* **2018**, *9*, 38. [CrossRef]
6. Savage, N. *Another Set of Eyes for Cancer Diagnostics*; Springer Nature: Cham, Switzerland, 2020; Volume 579, pp. S14–S16.
7. Ko, H.; Chung, H.; Kang, W.S.; Park, C.; Kim, D.W.; Kim, S.E.; Chung, C.R.; Ko, R.E.; Lee, H.; Seo, J.H.; et al. An Artificial Intelligence Model to Predict the Mortality of COVID-19 Patients at Hospital Admission Time Using Routine Blood Samples: Development and Validation of an Ensemble Model. *J. Med. Internet Res.* **2020**, *22*, e25442. [CrossRef]
8. Davenport, T.; Kalakota, R. The potential for artificial intelligence in healthcare. *Future Healthc. J.* **2019**, *6*, 94–98. [CrossRef]
9. Weis, C.-A.; Weihrauch, K.R.; Kriegsmann, K.; Kriegsmann, M. Unsupervised Segmentation in NSCLC: How to Map the Output of Unsupervised Segmentation to Meaningful Histological Labels by Linear Combination? *Appl. Sci.* **2022**, *12*, 3718. [CrossRef]
10. Park, S.; Kim, H.; Shim, E.; Hwang, B.-Y.; Kim, Y.; Lee, J.-W.; Seo, H. Deep Learning-Based Automatic Segmentation of Mandible and Maxilla in Multi-Center CT Images. *Appl. Sci.* **2022**, *12*, 1358. [CrossRef]
11. Wu, S.; Wu, Y.; Chang, H.; Su, F.T.; Liao, H.; Tseng, W.; Liao, C.; Lai, F.; Hsu, F.; Xiao, F. Deep Learning-Based Segmentation of Various Brain Lesions for Radiosurgery. *Appl. Sci.* **2021**, *11*, 9180. [CrossRef]
12. Asami, Y.; Yoshimura, T.; Manabe, K.; Yamada, T.; Sugimori, H. Development of Detection and Volumetric Methods for the Triceps of the Lower Leg Using Magnetic Resonance Images with Deep Learning. *Appl. Sci.* **2021**, *11*, 12006. [CrossRef]
13. Karhade, J.; Ghosh, S.K.; Gajbhiye, P.; Tripathy, R.K.; Rajendra Acharya, U. Multichannel Multiscale Two-Stage Convolutional Neural Network for the Detection and Localization of Myocardial Infarction Using Vectorcardiogram Signal. *Appl. Sci.* **2021**, *11*, 7920. [CrossRef]
14. Malik, S.; Akram, T.; Ashraf, I.; Rafiullah, M.; Ullah, M.; Tanveer, J. A Hybrid Preprocessor DE-ABC for Efficient Skin-Lesion Segmentation with Improved Contrast. *Diagnostics* **2022**, *12*, 2625. [CrossRef] [PubMed]
15. Lv, Z.; Poiesi, F.; Dong, Q.; Lloret, J.; Song, H. Deep Learning for Intelligent Human–Computer Interaction. *Appl. Sci.* **2022**, *12*, 11457. [CrossRef]
16. Mo, S.; Lu, P.; Liu, X. AI-Generated Face Image Identification with Different Color Space Channel Combinations. *Sensors* **2022**, *22*, 8228. [CrossRef] [PubMed]

17. Baazaoui, H.; Hubertus, S.; Maros, M.E.; Mohamed, S.A.; Förster, A.; Schad, L.R.; Wenz, H. Artificial Neural Network-Derived Cerebral Metabolic Rate of Oxygen for Differentiating Glioblastoma and Brain Metastasis in MRI: A Feasibility Study. *Appl. Sci.* **2021**, *11*, 9928. [CrossRef]
18. Taibouni, K.; Miere, A.; Samake, A.; Souied, E.; Petit, E.; Chenoune, Y. Choroidal Neovascularization Screening on OCTAngiography Choriocapillaris Images by Convolutional Neural Networks. *Appl. Sci.* **2021**, *11*, 9313. [CrossRef]
19. Cazzato, G.; Massaro, A.; Colagrande, A.; Lettini, T.; Cicco, S.; Parente, P.; Nacchiero, E.; Lospalluti, L.; Cascardi, E.; Giudice, G.; et al. Dermatopathology of Malignant Melanoma in the Era of Artificial Intelligence: A Single Institutional Experience. *Diagnostics* **2022**, *12*, 1972. [CrossRef]
20. Massaro, A.; Dipierro, G.; Cannella, E.; Galiano, A.M. Comparative Analysis among Discrete Fourier Transform, K-Means and Artificial Neural Networks Image Processing Techniques Oriented on Quality Control of Assembled Tires. *Information* **2020**, *11*, 257. [CrossRef]
21. Mohimont, L.; Alin, F.; Rondeau, M.; Gaveau, N.; Steffenel, L.A. Computer Vision and Deep Learning for Precision Viticulture. *Agronomy* **2022**, *12*, 2463. [CrossRef]
22. Massaro, A.; Panarese, A.; Selicato, S.; Galiano, A. CNN-LSTM Neural Network Applied for Thermal Infrared Underground Water Leakage. In Proceedings of the IEEE International Workshop on Metrology for Industry 4.0 and IOT (MetroInd4.0&IoT), Trento, Italy, 7–9 June 2022.
23. Bang, C.S.; Lee, J.J.; Baik, G.H. Artificial Intelligence for the Prediction of Helicobacter Pylori Infection in Endoscopic Images: Systematic Review and Meta-Analysis of Diagnostic Test Accuracy. *J. Med. Internet Res.* **2020**, *22*, e21983. [CrossRef]
24. Mohan, B.P.; Khan, S.R.; Kassab, L.L.; Ponnada, S.; Mohy-Ud-Din, N.; Chandan, S.; Dulai, P.S.; Kochhar, G.S. Convolutional neural networks in the computer-aided diagnosis of Helicobacter pylori infection and non-causal comparison to physician endoscopists: A systematic review with meta-analysis. *Ann. Gastroenterol.* **2021**, *34*, 20–25. [CrossRef] [PubMed]
25. Pecere, S.; Milluzzo, S.M.; Esposito, G.; Dilaghi, E.; Telese, A.; Eusebi, L.H. Applications of Artificial Intelligence for the Diagnosis of Gastrointestinal Diseases. *Diagnostics* **2021**, *11*, 1575. [CrossRef] [PubMed]
26. Mohan, B.P.; Khan, S.R.; Kassab, L.L.; Ponnada, S.; Dulai, P.S.; Kochhar, G.S. Accuracy of convolutional neural network-based artificial intelligence in diagnosis of gastrointestinal lesions based on endoscopic images: A systematic review and meta-analysis. *Endosc. Int. Open* **2020**, *8*, E1584–E1594. [CrossRef] [PubMed]
27. Cardos, A.I.; Maghiar, A.; Zaha, D.C.; Pop, O.; Fritea, L.; Miere, F.; Cavalu, S. Evolution of Diagnostic Methods for Helicobacter pylori Infections: From Traditional Tests to High Technology, Advanced Sensitivity and Discrimination Tools. *Diagnostics* **2022**, *12*, 508. [CrossRef]
28. Shichijo, S.; Nomura, S.; Aoyama, K.; Nishikawa, Y.; Miura, M.; Shinagawa, T.; Takiyamaah, H.; Tanimoto, T.; Ishihara, S.; Tada, T.; et al. Application of convolutional neural networks in the diagnosis of Helicobacter pylori infection based on endoscopic images. *EBioMedicine* **2017**, *25*, 106–111. [CrossRef]
29. Itoh, T.; Kawahira, H.; Nakashima, H.; Yata, N. Deep learning analyzes Helicobacter pylori infection by upper gastrointestinal endoscopy images. *Endosc. Int. Open* **2018**, *6*, E139–E144. [CrossRef]
30. Nakashima, H.; Kawahira, H.; Kawachi, H.; Sakaki, N. Artificial intelligence diagnosis of Helicobacter pylori infection using blue laser imaging-bright and linked color imaging: A single-center prospective study. *Ann. Gastroenterol.* **2018**, *31*, 462–468. [CrossRef]
31. Shichijo, S.; Endo, Y.; Aoyama, K.; Takeuchi, Y.; Ozawa, T.; Takiyama, H.; Matsuo, K.; Fujishiro, M.; Ishihara, S.; Ishihara, R.; et al. Application of convolutional neural networks for evaluating Helicobacter pylori infection status on the basis of endoscopic images. *Scand. J. Gastroenterol.* **2019**, *54*, 158–163. [CrossRef]
32. Zheng, W.; Zhang, X.; Kim, J.J.; Zhu, X.; Ye, G.; Ye, B.; Wang, J.; Luo, S.; Li, J.; Yu, T.; et al. High Accuracy of Convolutional Neural Network for Evaluation of Helicobacter pylori Infection Based on Endoscopic Images: Preliminary Experience. *Clin. Transl. Gastroenterol.* **2019**, *10*, e00109. [CrossRef]
33. Dong-hyun, K.; Hyun-Chin, C.; Hyun-chong, C. Gastric Lesion Classification Using Deep Learning Based on Fast and Robust Fuzzy C-Means and Simple Linear Iterative Clustering Superpixel Algorithms. *J. Electr. Eng. Technol.* **2019**, *14*, 2549–2556.
34. Guimarães, P.; Keller, A.; Fehlmann, T.; Lammert, F.; Casper, M. Deep-learning based detection of gastric precancerous conditions. *Gut* **2020**, *69*, 4–6. [CrossRef]
35. Kim, H.H.; Lim, Y.S.; Seo, S.-I.; Lee, K.J.; Kim, J.Y.; Shin, W.G. A Deep Recurrent Neural Network-Based Explainable Prediction Model for Progression from Atrophic Gastritis to Gastric Cancer. *Appl. Sci.* **2021**, *11*, 6194. [CrossRef]
36. Huang, C.-R.; Chung, P.-C.; Sheu, B.-S.; Kuo, H.-J.; Popper, M. Helicobacter pylori-related gastric histology classification using support-vector-machine-based feature selection. *IEEE Trans. Inf. Technol. Biomed.* **2008**, *12*, 523–531. [CrossRef]
37. Kanesaka, T.; Lee, T.-C.; Uedo, N.; Lin, K.P.; Chen, H.Z.; Lee, J.Y.; Wang, H.-P.; Chang, H.T. Computer-aided diagnosis for identifying and delineating early gastric cancers in magnifying narrow-band imaging. *Gastrointest Endosc.* **2018**, *87*, 1339–1344. [CrossRef] [PubMed]
38. Yasuda, T.; Hiroyasu, T.; Hiwa, S.; Okada, Y.; Hayashi, S.; Nakahata, Y.; Yasuda, Y.; Omatsu, T.; Obora, A.; Kojima, T.; et al. Potential of automatic diagnosis system with linked color imaging for diagnosis of Helicobacter pylori infection. *Dig. Endosc.* **2020**, *32*, 373–381. [CrossRef]

39. Chapelle, N.; Osmola, M.; Martin, J.; Blin, J.; Leroy, M.; Jirka, I.; Moussata, D.; Lamarque, D.; Olivier, R.; Tougeron, D.; et al. Serum Pepsinogens Combined with New Biomarkers Testing Using Chemiluminescent Enzyme Immunoassay for Non-Invasive Diagnosis of Atrophic Gastritis: A Prospective, Multicenter Study. *Diagnostics* **2022**, *12*, 695. [CrossRef] [PubMed]
40. Takiyama, H.; Ozawa, T.; Ishihara, S.; Fujishiro, M.; Shichijo, S.; Nomura, S.; Miura, M.; Tada, T. Automatic anatomical classification of esophagogastroduodenoscopy images using deep convolutional neural networks. *Sci. Rep.* **2018**, *14*, 7497. [CrossRef]
41. Wu, L.; Zhou, W.; Wan, X.; Zhang, J.; Shen, L.; Hu, S.; Ding, Q.; Mu, G.; Yin, A.; Yu, H. A deep neural network improves endoscopic detection of early gastric cancer without blind spots. *Endoscopy* **2019**, *51*, 522–531. [CrossRef]
42. Hirasawa, T.; Aoyama, K.; Tanimoto, T.; Ishihara, S.; Shichijo, S.; Ozawa, T.; Ohnishi, T.; Fujishiro, M.; Matsuo, K.; Tada, T.; et al. Application of artificial intelligence using a convolutional neural network for detecting gastric cancer in endoscopic image. *Gastric Cancer* **2018**, *21*, 653–660. [CrossRef]
43. Ishioka, M.; Hirasawa, T.; Tada, T. Detecting gastric cancer from video images using convolutional neural networks. *Dig. Endosc.* **2019**, *31*, e34–e35. [CrossRef] [PubMed]
44. Li, L.; Chen, Y.; Shen, Z.; Zhang, X.; Sang, J.; Ding, Y.; Yang, X.; Li, J.; Chen, M.; Yu, C.; et al. Convolutional neural network for the diagnosis of early gastric cancer based on magnifying narrow band imaging. *Gastric Cancer* **2020**, *23*, 126–132. [CrossRef]
45. Horiuchi, Y.; Aoyama, K.; Tokai, Y.; Hirasawa, T.; Yoshimizu, S.; Ishiyama, A.; Yoshio, T.; Tsuchida, T.; Fujisaki, J.; Tada, T.; et al. Convolutional neural network for differentiating gastric cancer from gastritis using magnified endoscopy with narrow band imaging. *Dig. Dis. Sci.* **2020**, *65*, 1355–1363. [CrossRef]
46. Zhu, Y.; Wang, Q.C.; Xu, M.D.; Zhang, Z.; Cheng, J.; Zhong, Y.-S.; Zhang, Y.-Q.; Chen, W.-F.; Yao, L.-Q.; Li, Q.L. Application of convolutional neural network in the diagnosis of the invasion depth of gastric cancer based on conventional endoscopy. *Gastrointest. Endosc.* **2019**, *89*, 806–815. [CrossRef] [PubMed]
47. Alquran, H.; Alsalatie, M.; Mustafa, W.A.; Abdi, R.A.; Ismail, A.R. Cervical Net: A Novel Cervical Cancer Classification Using Feature Fusion. *Bioengineering* **2022**, *9*, 578. [CrossRef] [PubMed]
48. Alquran, H.; Al-Issa, Y.; Alsalatie, M.; Mustafa, W.A.; Qasmieh, I.A.; Zyout, A. Intelligent Diagnosis and Classification of Keratitis. *Diagnostics* **2022**, *12*, 1344. [CrossRef] [PubMed]
49. Alquran, H.; Mustafa, W.A.; Qasmieh, I.A.; Yacob, Y.M.; Alsalatie, M.; Al-Issa, Y.; Alqudah, A.M. Cervical Cancer Classification Using Combined Machine Learning and Deep Learning Approach. *Comput. Mater. Contin.* **2022**, *72*, 5117–5134. [CrossRef]
50. Li, X.; Wang, Y.; Tang, B.; Qin, Y.; Zhang, G. Canonical correlation analysis of dimension reduced degradation feature space for machinery condition monitoring. *Mech. Syst. Signal Process.* **2023**, *182*, 109603. [CrossRef]
51. Hotelling, H. Relations between two sets of variates. *Biometrika* **1936**, *28*, 321–377. [CrossRef]
52. Anderson, T.W. *An Introduction to Multivariate Statistical Analysis*, 3rd ed.; John Willey and Sons: Hoboken, NJ, USA, 2003.
53. Brogin, J.; Faber, F.; Bueno, D. Enhanced use practices in SSVEP-based BCIs using an analytical approach of canonical correlation analysis. *Biomed. Signal Process.* **2020**, *55*, 101644. [CrossRef]
54. Ali, M.H.; Baiee, W.R. Choosing an Appropriate Feature Selection Method to Enhance Feed-Forward ANN. In Proceedings of the International Conference on Communication and Information Technology, ICICT, Basrah, Iraq, 23–24 December 2021; pp. 86–91.
55. Kononenko, I. Estimating attributes: Analysis and extensions of RELIEF. In *Machine Learning: ECML-94. ECML 1994*; Bergadano, F., De Raedt, L., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 1994; Volume 784. [CrossRef]
56. Zhang, Y. A Hybrid Convolutional Neural Network and Relief-F Algorithm for Fault Power Line Recognition in Internet of Things-Based Smart Grids. *Wirel. Commun. Mob. Comput.* **2022**, 4911553. [CrossRef]
57. Shen, Z.; Yang, Y.; Ai, L.; Yu, C.; Su, M. A hybrid CART-GAMs model to evaluate benthic macroinvertebrate habitat suitability in the Pearl River Estuary, China. *Ecol. Indic.* **2022**, *143*, 109368. [CrossRef]
58. Obleagă, C.V.; Vere, C.C.; Pătraşcu, A.M.; Moraru, E.; Crafciuc, A.V.; Foarfă, M.C.; Mogoantă, S.Ş.; Streba, C.T.; Bondari, S.; Paitici, Ş.; et al. Severe upper gastrointestinal bleeding determined by a gastric lymphoma associated with Helicobacter pylori-positive atrophic gastritis. *Rom. J. Morphol. Embryol.* **2017**, *58*, 611–617. [PubMed]

diagnostics

MDPI

Article

Feature Transformation for Efficient Blood Glucose Prediction in Type 1 Diabetes Mellitus Patients

Hatim Butt [1], Ikramullah Khosa [1,*] and Muhammad Aksam Iftikhar [2,*]

[1] Department of Electrical and Computer Engineering, Lahore Campus, COMSATS University Islamabad, Islamabad 54000, Pakistan
[2] Department of Computer Science, Lahore Campus, COMSATS University Islamabad, Islamabad 54000, Pakistan
* Correspondence: ikramullahkhosa@cuilahore.edu.pk (I.K.); aksamiftikhar@cuilahore.edu.pk (M.A.I.)

Abstract: Diabetes Mellitus, a metabolic disease, causes the body to lose control over blood glucose regulation. With recent advances in self-monitoring systems, a patient can access their personalized glycemic profile and may utilize it for efficient prediction of future blood glucose levels. An efficient diabetes management system demands the accurate estimation of blood glucose levels, which, apart from using an appropriate prediction algorithm, depends on discriminative data representation. In this research work, a transformation of event-based data into discriminative continuous features is proposed. Moreover, a multi-layered long short-term memory (LSTM)-based recurrent neural network is developed for the prediction of blood glucose levels in patients with type 1 diabetes. The proposed method is used to forecast the blood glucose level on a prediction horizon of 30 and 60 min. The results are evaluated for three patients using the Ohio T1DM dataset. The proposed scheme achieves the lowest RMSE score of 14.76 mg/dL and 25.48 mg/dL for prediction horizons of 30 min and 60 min, respectively. The suggested methodology can be utilized in closed-loop systems for precise insulin delivery to type 1 patients for better glycemic control.

Keywords: diabetes mellitus; blood glucose prediction; forecasting; long short-term memory

Citation: Butt, H.; Khosa, I.; Iftikhar, M.A. Feature Transformation for Efficient Blood Glucose Prediction in Type 1 Diabetes Mellitus Patients. *Diagnostics* **2023**, *13*, 340. https://doi.org/10.3390/diagnostics13030340

Academic Editors: Wan Azani Mustafa and Hiam Alquran

Received: 20 December 2022
Revised: 11 January 2023
Accepted: 15 January 2023
Published: 17 January 2023

1. Introduction

The ability of the human body to control imbalanced blood glucose (BG) levels is administered by the pancreas. When the glucose level in the bloodstream increases beyond the normal range, the pancreas senses this and instructs the β-cells to release insulin. As a result, excess sugar begins to be stored in fat cells, keeping the blood glucose in the normal range. In contrast, the low sugar level in the bloodstream is handled by α-cells, which release glucagon hormone. This hormone dictates the liver releasing glucose into the bloodstream, maintaining BG levels in the safe range. Diabetes mellitus (DM) is a medical condition in which the human body loses this closed-loop control partially or completely. In Type-1 diabetes, the human pancreas loses its ability to produce insulin to regulate elevated sugar levels.

In Type-2 diabetes, the patient becomes insulin resistant, or the insulin produced becomes ineffective at decreasing the high BG levels. In comparison to other diseases, diabetics have a higher prevalence of illness and mortality. Regardless of its present incidence and burden, i.e., 415 million adults, this disease is projected to affect 642 million by 2040 [1]. In 2015, it was estimated that there were 415 million people with diabetes aged 20–79 years, 5 million deaths attributable to diabetes, and a total global health expense due to diabetes of 673 billion US$. This figure was reported to be 760 billion dollars in a 2019 survey [2]. Since no established cure exists for diabetes, disease management is the only option. Efficient glycemic control requires constant monitoring of BG levels, sufficient physical activity, precise insulin delivery, and appropriate diet intake. In this regard, future estimation of blood glucose levels is the most important aspect for a patient.

A realistic assessment of BG levels can help patients avoid fatal glycemic conditions such as hypo-glycemia (BG < 70 mg/dL) or hyper-glycemia (BG > 180 mg/dL). The prediction of blood glucose levels is quite a complex task due to its non-linear nature and multiple confounding factors. Although a number of factors affect BG levels, the few important ones include previous BG values, dietary intakes, insulin infusion, and the physical activity of a patient [3].

BG prediction methodologies can be categorized into three types: Physiology-based (knowledge-based), data-driven (empirical-based), and hybrid. The physiological model requires comprehensive knowledge of individual underlying physiological mechanisms. It divides the person's BG metabolism into three different regulatory sectors: BG dynamics, meal absorption dynamics, and insulin dynamics [4]. Each one is modeled using a variety of mathematical (differential) equations and probabilistic frameworks. The physiology-based approach is primarily categorized into two: The lumped (semi-empirical) model and the comprehensive model. In contrast to the physiology-based approach, the data-driven strategy relies on the person's self-recorded historical data and needs little knowledge of the underlying physiological mechanism; thus, referred to as the "black box" approach. Neural networks (NNs) and autoregressive (AR) models are the most common examples of this type of approach. A data-driven approach can generally be divided into three models: A time series model, a machine learning model, and a hybrid model. Hybrid models, as the name suggests, are a combination of both physiological and data-driven models. The blood glucose prediction approaches are illustrated in Figure 1.

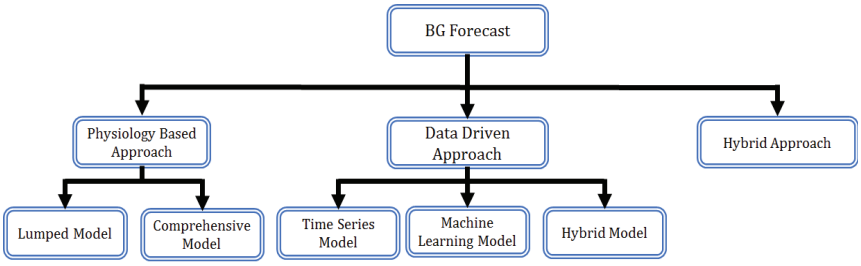


Figure 1. Blood glucose prediction approaches.

Historically, efforts have been made to employ statistical approaches for the prediction of blood glucose levels. A multivariate statistical modeling approach was employed using a latent variable for short-term blood glucose forecasting in T1DM patients [5]. Another study presented a comparative analysis of various machine learning techniques with classical time series forecasting methods such as auto-regression with Exogenous input (ARX) for glucose level prediction [6]. Event detection such as hypo/hyperglycemia has also been an interest of researchers for T1DM patients [7]. In addition to BG level forecasting, the study [8] presented the efficiency of detecting hypo/hyperglycemic events in Type 1 diabetic patients. In another study, the application of the XG-Boost algorithm was proposed for glycemia prediction [9]. The authors in the study [10] developed a model of multi-horizon glucose level prediction with alerts for the hypoglycemia condition.

Recently, machine learning and deep learning applications for blood glucose prediction in patients with Type 1 and Type 2 diabetes have gained a lot of attention. A recurrent neural network (RNN) was employed for blood glucose level forecasting with the provision of forecast certainty [11]. A personalized BG level forecasting model was proposed based on a convolutional neural network (CNN) by converting the regression task into a classification task [12]. Another study [13] proposed a personalized deep learning framework for the prediction of BG distribution over the prediction horizon of 30 and 60 min in T1DM patients. Authors in the study [14] worked on the development of a multitask learning [15] approach using a convolutional recurrent neural network MTCRNN for short-term personalized

blood glucose forecasting. Multitask learning allows for learning from the data of all available diabetic subjects who took part in the research.

The authors of another study [16] proposed the rectification of a sensor defect for CGM readings using the Kalman filter smoothing technique [17]. The prediction of blood glucose levels using a stacked Long-short term memory-based RNN model was presented. The proposed work on sensor fault rectification is prone to flaws since the target labels were modified in lieu of sensor noise rectification.

Deep learning approaches have recently played a significant role in increasing blood glucose forecasting accuracy. However, the need to furnish an optimized deep learning model, for multi-horizon prediction, is not over and requires more research. In addition to the learning model, the discriminative feature representation of sensor data is equally vital to establish an efficient and robust prediction model. Pre-processing of sensor data including normalization, interpolation, and filtering has been performed. However, the discriminative feature transformation of sensor data is rare in the literature.

In this research work, a deep learning technique based on multi-layer Long Short-Term Memory for blood glucose prediction of T1DM patients is presented. The pre-processing of sensor data is performed, including sampling consistency, interpolation to fill missing samples, and filtering. Moreover, to achieve lower forecasting errors, the event-based features are transformed into time-stamped samples. The Ohio T1DM dataset published by the University of Ohio for the BGLP challenge in 2018 is used. It contains eight weeks of data from six T1DM patients including a variety of physiological features along with event-based information such as meal and insulin injection. The contributions of this research work are as follows:

- Pre-processing of the T1DM dataset is performed including the incorporation of time-consistency in features as per the target values, interpolation for missing values, and filtering to achieve smoothing.
- Based on the relation with blood glucose levels, two event-based features, namely, meal and insulin, are transformed into continuous features, which led to improved accuracy.
- The LSTM and Bi-LSTM-based RNN models are developed and optimized to achieve minimum prediction error for blood glucose levels.
- The proposed models outperformed the state-of-the-art methods for the prediction horizons of 30 and 60 min.

The paper is organized as follows: Section 2 presents the dataset and the pre-processing. Feature transformation is explained in Section 3. The evaluation strategy is discussed in Section 4. The learning algorithm is presented in Section 5. Section 6 presents the results and discussion. Finally, the conclusion is added in Section 7.

2. Dataset and Pre-Processing

2.1. Dataset

The Ohio T1DM dataset [18] published by the University of Ohio is used in this research. It is a great source of blood glucose forecasting research due to the fact that it provides a diverse range of characteristics associated with Type 1 diabetes patients, which, according to the most recent literature on diabetes [3], affect blood glucose levels. The dataset was released in two phases: In 2018 and in 2020. Each release contains data from six T1DM patients who were on insulin therapy wearing a fitness tracker and Medtronic's insulin pump with continuous glucose monitoring (CGM). In addition, the self-reported life events from these contributors were also recorded via a smartphone application. The data recorded for each individual includes CGM sampled at 5 min intervals, finger stick blood glucose levels for self-monitoring, insulin doses (both basal and bolus), self-reported time of physical activity, work, stress, sleep, and illness, and physiological information from a fitness tracker such as Galvanic skin resistance (GSR), heart rate, step count, air temperature, and skin temperature. The dataset contains the sensors' data for a total period of eight weeks. Several data features, such as CGM and heart rate, are recorded at 5 min intervals, whereas others, such as meal and insulin, are recorded only at the time of eating a meal

or insulin intake, respectively. An illustrative overview of these features recorded from one of the contributors (ID 570) is illustrated in Figure 2. In each release of the Ohio T1DM dataset, the training data and testing data are already segregated by the provider. Table 1 summarizes the data of six patients in the Ohio T1DM dataset version 2018, including the number of training and testing samples predefined by the data providers.

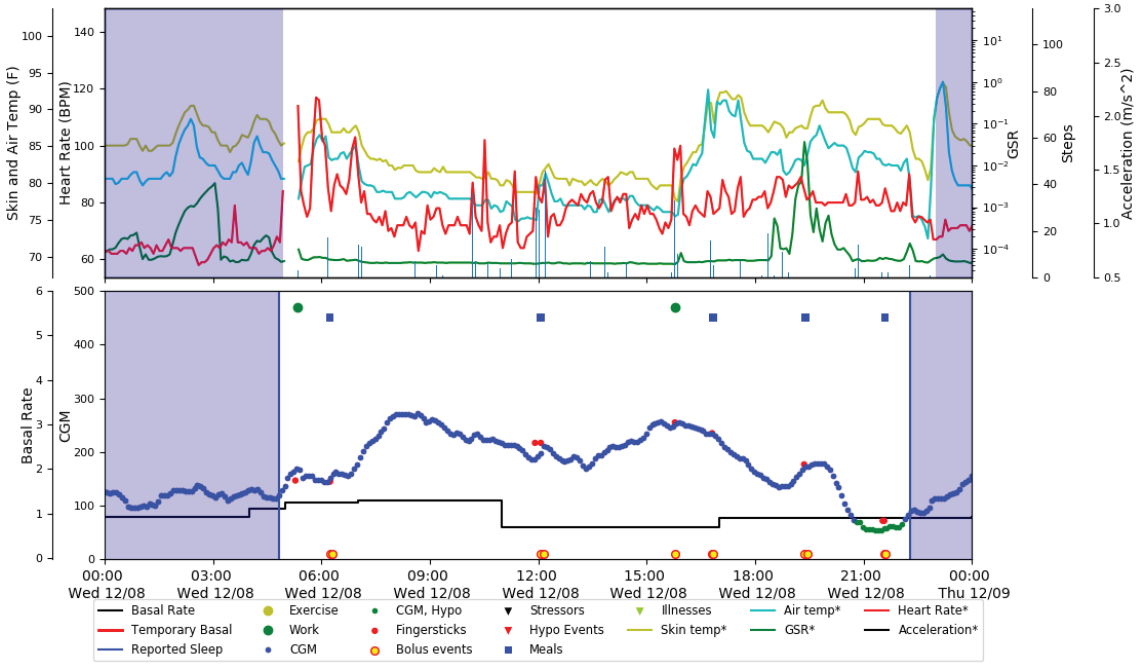


Figure 2. Feature visualization using data of T1DM patient ID 570.

Table 1. Ohio T1DM dataset contributor details.

ID	Gender	Pump Model	Sensor Band	Train Samples	Test Samples
559	female	530 G	Basis	10,796	2514
563	male	530 G	Basis	12,124	2570
570	male	530 G	Basis	10,982	2745
575	female	530 G	Basis	11,866	2590
588	female	530 G	Basis	12,640	2791
591	female	530 G	Basis	10,847	2760

2.2. Feature Vector

The dataset includes automatic and manual measuring of several parameters forming the feature vector for a patient. The type of feature is periodic and event-based; however, all of them are recorded with a time stamp. The details of recorded features are provided in Table 2.

Table 2. Feature details in Ohio T1DM dataset.

Ser #	Feature Name	Type	Source
1	glucose level	periodic	medtronic Sensor
2	basal insulin	event	self-Reported
3	bolus insulin	event	self-Reported
4	finger stick	event	self-Reported
5	meal	event	self-Reported
6	exercise	event	self-Reported
7	sleep	event	self-Reported
8	work	event	self-Reported
9	hypo Events	event	self-Reported
10	air temperature	periodic	basis Sensor
11	GSR	periodic	basis Sensor
12	heart Rate	periodic	basis Sensor
13	skin temperature	periodic	basis Sensor
14	sleep	periodic	basis Sensor
15	steps	periodic	basis Sensor

2.3. Pre-Processing

2.3.1. Time Coherence

The features, as shown in Table 2, are periodic and event-based in nature. Continuous glucose monitoring (CGM)—the vital information—is recorded at a sampling interval of 5 min. However, during sensor replacement, the time coherence breaks. Consequently, upcoming samples are time inconsistent. To avoid this, new samples are added to the data, ensuring a sampling time of 5 min. The net effect is the introduction of many new samples. For example, there exist 10,982 training samples of CGM for patient ID 570. After time coherence, 649 more samples were introduced. Next, since CGM is the most important feature, the rest of the periodic features were also time-aligned with the CGM time stamps.

2.3.2. Interpolation

The criterion for filling new CGM values was based on the forward-filling rule with a limit of 1, i.e., if there is a single new sample, it will be assigned the magnitude of the previous sample. In the case there were 2 or more consecutive new samples, interpolation was performed to fill the samples. Several interpolation techniques including linear, Akima cubic, spline, and shape-preserving methods were used. It was found that linear interpolation learns the best behavior of glucose levels, producing the best prediction results. Figure 3 shows the linear interpolation result of the CGM feature over a period of 4 days. The orange color in Figure 3 shows the interpolated samples, which were empty in the original data. The same interpolation strategy was adopted for the test data. However, for result evaluation, the interpolated samples in the test data were not considered for error calculation.

2.3.3. Median Filtering

Observing Figure 3, sharp changes in the CGM profile are visible, which can be regarded as sensor noise. To smooth the profile, a median filter is used replacing the extreme values with the median value as used in an earlier study [19]. Figure 4 shows the CGM profile smoothing using the median filter where the profiles of both the original data and the filtered data are shown. Smoothing the feature profile enables the predictor to learn better. The filtering scheme was only implemented on training data while the test data remained unchanged.

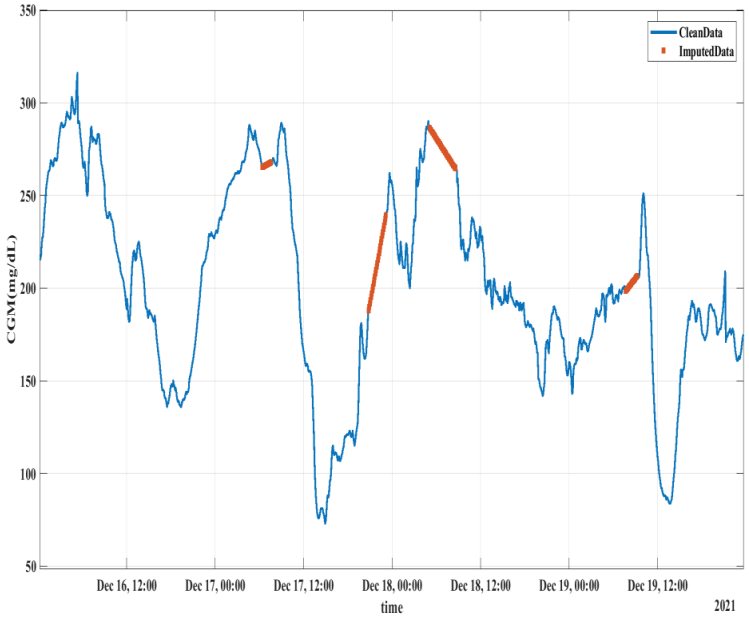


Figure 3. Missing CGM data imputation using linear interpolation. The orange lines represent the interpolated sample data, which were missing in the original dataset.

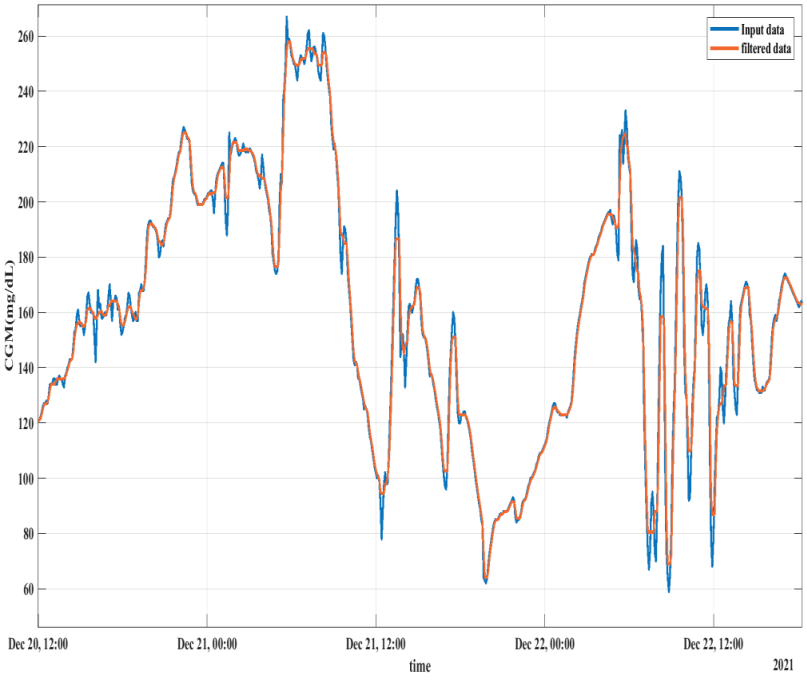


Figure 4. Data smoothing using median filtering.

3. Feature Transformation

The data include several periodic features such as CGM and some event-based features such as meal and sleep. Since all the periodic features are made time-coherent with the feature, it is beneficial to approximate the event-based features as time-coherent periodic values. The meal and insulin injection are two features recorded at random time stamps. Currently, such features can be treated as binary features where a value of 1 will be observed when a meal was eaten and 0 for the rest of the time. Such values are not only insignificant to the feature vector but also may cause an adverse effect on the learning of the algorithm. Therefore, we propose the transformation of such features into continuous-value observation. This transformation will improve the feature representation for meal and insulin information. To consider those features for blood glucose prediction, they need to be made time-coherent with CGM, which is performed and explained in the following sub-sections.

3.1. Carbs from Meal to Operative Carbs Transformation

One of the data features is the carb intake by patients in the form of a meal, which has a direct impact on blood glucose. The rate at which the carbs affect blood glucose levels depends on their type, as high glycemic index (GI) carbs instantly affect the blood glucose while low-glycemic-index carbs have a slower effect, but the effect remains over a longer period of time. Research [20] revealed that a normal meal starts raising the blood glucose level after 15 min of having the meal. The glucose level approaches its peak in 60 min and then decays at a certain rate such that after 3 h it becomes steady. Based on that, carbohydrate absorption can be converted into operative carbs, which provides a continuous-value feature. The primary objective of this conversion is to determine the effectiveness of carb intake at any certain time in the human body. The conversion model can be estimated by combining the rise and decay profile as shown in Figure 5.

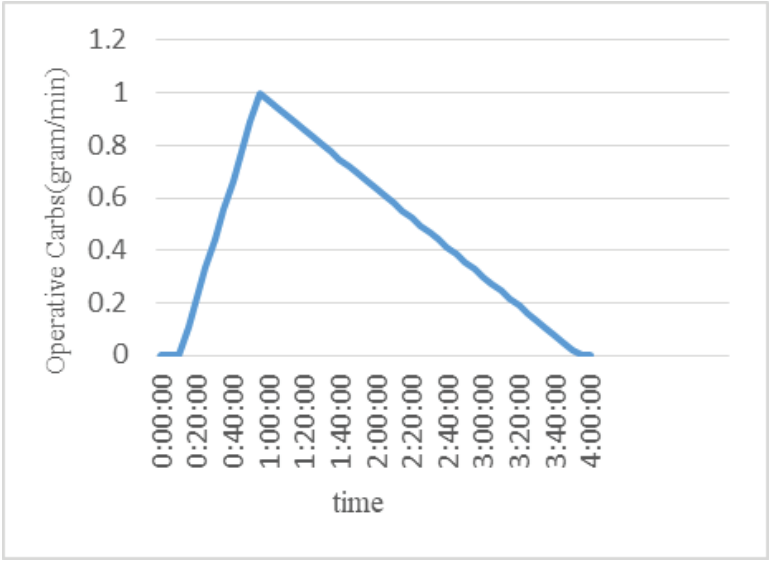


Figure 5. Carbohydrates to the operative carb conversion profile.

The operative carbohydrates at any time in the data are estimated using the following procedure:

i Based on the assumption of working at a 5 min reference time scale, the first three samples after taking the meal have zero operative carbs.
ii The operative carbs start rising at a rate of 0.11 (11.1%).

iii At the 12th sample or the 60th minute after having the meal, the value of operative carbs attains its maximum value, which is almost equal to the total amount of carbs.
iv After that, the operative carbs start decreasing at a rate of 0.028 (2.8%). It reaches zero after 3 h.

The mathematical piecewise function of the above-mentioned method can be obtained as follows:

$$C_{op}(t_s) = \begin{cases} 0, & 0 \leq t_{meal} \leq 2 \\ (t_s - t_{meal+2})/(t_s - t_{s-1}) * \alpha_{inc} * C_{meal}, & 3 \leq t_{meal} < 12 \\ \left(1 - (t_s - t_{peak})\right)/(t_s - t_{s-1}) * \alpha_{dec}) * C_{meal}, & 12 \leq t_{meal} < 48 \end{cases} \quad (1)$$

where:

t_s is the sampling time.
t_{meal} is the time when the meal is encountered.
C_{op} is the effective carbohydrates at any given time.
C_{meal} is the total amount of carbohydrates taken in a meal.
t_{peak} is the time when C_{op} reaches its maximum value $\rightarrow C_{op} \cong C_{meal}$.
α_{inc} is the increasing rate of the curve.
α_{dec} is the decreasing rate of the curve.

Figure 6 shows the meal carbs transformation to the operative carbs of patient ID 570.

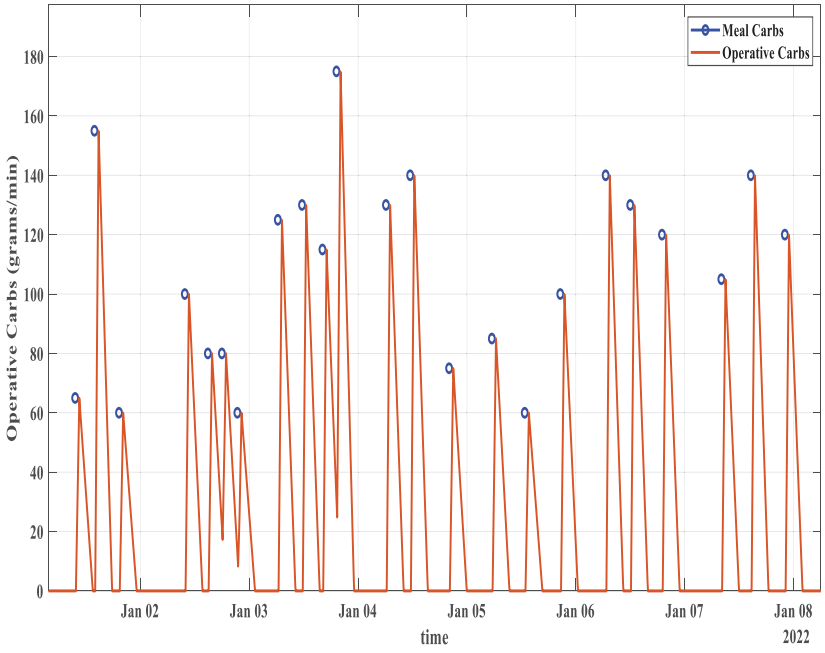


Figure 6. Meal into operative carbs transformation for data of patient ID 570. Blue circles represent the time of meal and the corresponding carbs whereas the brown curves show the transformation into operative carbs.

3.2. Bolus Insulin to Active Insulin Transformation

Patients with Type-1 diabetes rely on exogenous insulin delivery [21], which usually has two types, one is rapid-acting insulin named bolus and the second is slow-acting insulin termed basal insulin. Bolus insulin is normally taken before meals to compensate for the effect of a rise in blood glucose. Most insulin-pump users are familiar with the insulin

activity curve, as insulin's effect is time dependent. It takes a short while for insulin to have an effect on the blood glucose level: Normally, it peaks after 1 h and decays gradually over 6 h. The effect of insulin is characterized by the type and brand of insulin used, such as rapid-acting insulin—child, which has a peak time of 65 min, whereas rapid-acting insulin—adult has a peak time of 75 min. The insulin activity remaining or the percentage of active insulin after a bolus insulin injection can be modeled with an exponential decay curve [22] known as insulin on board (IOB). It is the inverse of the insulin activity curve for a given duration of insulin action (DIA).

The transformation of bolus insulin into active insulin is performed. With insulin infusion, the blood glucose is expected to drop based on several aspects such as the insulin sensitivity factor (ISF), amount of dose, and type of insulin. The effect of insulin could be visualized in three different ways, one of which is expected active insulin. We envision the effect of bolus insulin in terms of active insulin. The expected active insulin in the human body at any time is the product of the original insulin delivered and the percent of insulin activity remaining (IOB). Figure 7 shows the transformation of four units of bolus insulin into the respective active insulin.

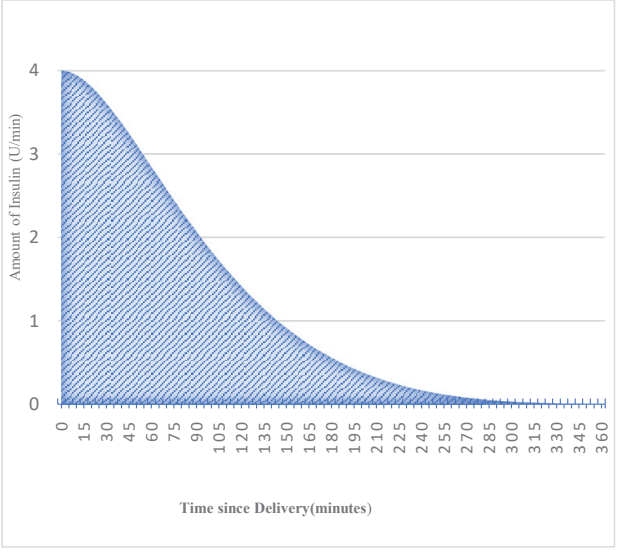


Figure 7. Bolus-to-active-insulin transformation curve.

The mathematical expression for active insulin can be described as follows:

$$\mathrm{IOB}(t_s) = 1 - S*(1-a) * \left\{ \left(\frac{{t_s}^2}{\tau * t_d * (1-a)} - \frac{t_s}{\tau} - 1 \right) * e^{-\frac{t_s}{\tau}} + 1 \right\} \quad (2)$$

where:

t_s is the sampling time.
t_d is the total duration of insulin activity.
τ is the time constant of exponential decay.
a is the Rise time factor.
S is the Auxiliary Scale factor.

$$\tau = t_p * \left(1 - \frac{t_p}{t_d}\right) / \left(1 - 2 * \frac{t_p}{t_d}\right) \quad (3)$$

$$a = 2 * \tau / t_d \quad (4)$$

$$S = 1/\left(1 - a + (1 + a) * e^{-\frac{t_d}{\tau}}\right) \quad (5)$$

Figure 8 shows the bolus insulin intake and the corresponding active insulin for patient ID 570.

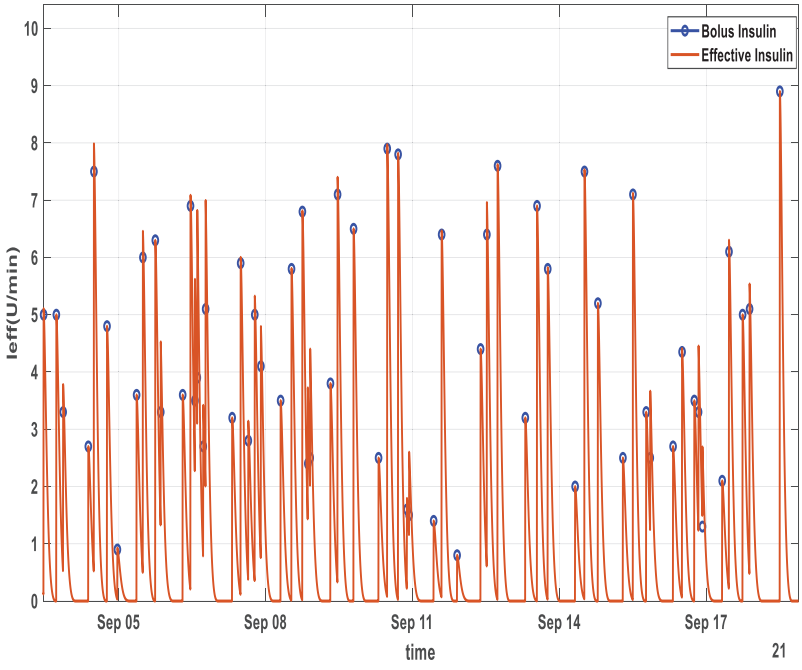


Figure 8. Bolus insulin transformation into active insulin for patient ID 570 over a period of 5 days. Blue circles represent the time and amount of bolus insulin, whereas the brown curves represent the transformation into active insulin over the next 6 h.

4. Evaluation Strategy

4.1. Evaluation Metric

The root mean squared error (RMSE) is used to evaluate the performance, as it has been widely used in existing studies. A prediction horizon (PH) of 30 min and 60 min is considered. If the prediction for the ith example made by the model against the actual target y_i is denoted as $\overline{y_i}$, then RMSE can be expressed as:

$$\text{RMSE} = \sqrt{1/n \sum_{i=1}^{n} (\overline{y_i} - y_i)^2} \quad (6)$$

where n is the total number of test samples. It is, once again, worth mentioning that within the test data, only samples that were recorded using the sensor were used to compute the error and not the interpolated samples.

4.2. Feature Configurations

The dataset contains plenty of features; to start with, a recent review study guides the suitable choice based on their historical employment for blood glucose prediction [3]. The adoption of different features in terms of percentage in 624 previous studies is shown in Figure 9.

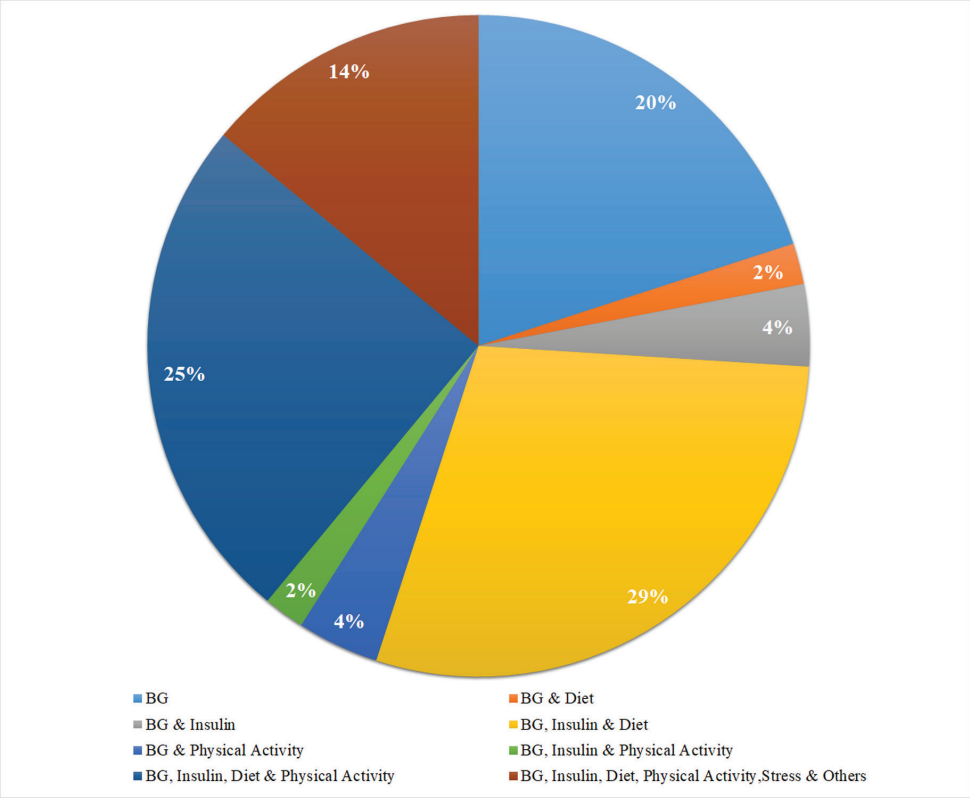


Figure 9. Input Feature set contribution in recent studies for diabetes prediction.

Based on widely used features, we opted for BG, diet, and insulin. Overall, three feature configurations are used in this work, as follows:

Configuration 1 (C-01): In this scheme, a univariate model is developed using CGM data only.
Configuration 2 (C-02): For this scheme, a combination of CGM and operative carbs is used as a feature set.
Configuration 3 (C-03): This is the combination of three features: CGM, operative carbs, and active insulin.

In addition to the mentioned features, other features such as basal insulin, heart rate, and physiological features were also used; however, the performance was inferior. The complete profiles of the three selected features for this study are shown in Figure 10.

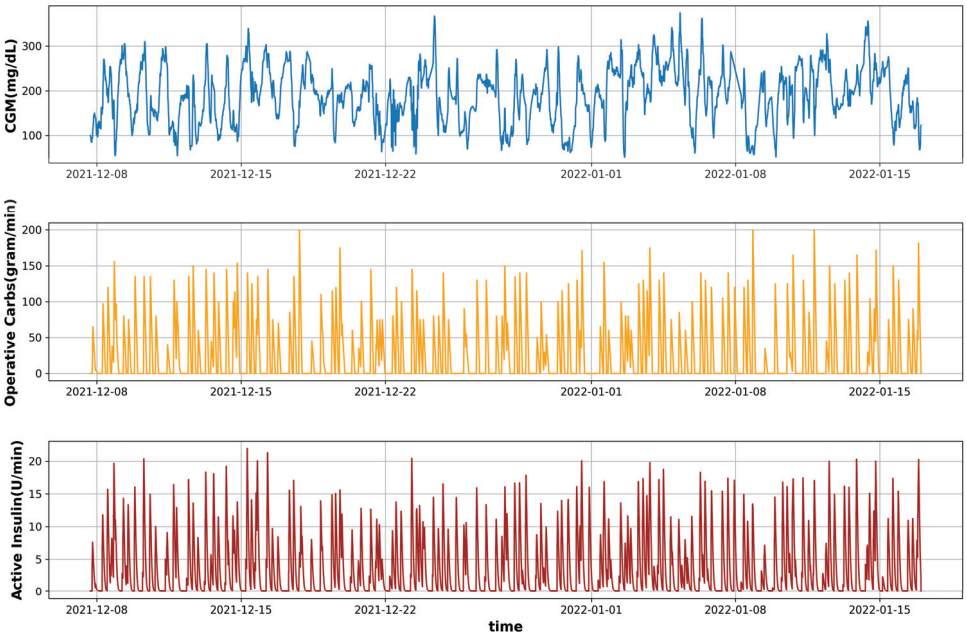


Figure 10. The profiles of CGM, Operative Carbs, and Active insulin in the dataset for patient ID 570.

5. Learning Model

Continuous blood glucose prediction is a multistep time-series forecasting task, and therefore a sequence-to-sequence learning algorithm is a suitable choice. The class of deep learning models for such a task belongs to recurrent neural networks (RNNs). The long short-term memory (LSTM) model is considered due to its ability to memorize long-term temporal dependencies [23]. In this section, the model architecture is presented first, and then the training and optimization details of the developed model are discussed.

5.1. Model Arhitecture

RNNs are a type of neural network that has the capability to learn sequential data. This is because of the provision of feedback paths known as the hidden state h. The hidden state can be computed using the following expression:

$$h(t) = f_c(Ux(t) + Wh(t-1) + b) \tag{7}$$

where $h(t)$ is the current hidden state, $h(t-1)$ is the previous hidden state, f_c is a non-linear activation function such as tanh or ReLU (Rectified Linear Unit), $x(t)$ is the input vector, b is the bias term, and U and W are the weight matrices defined between input-to-hidden and hidden-to-hidden connections, respectively. The standard RNN computes the output through cyclic flow across different hidden layers. It faces the problem of a vanishing gradient due to the short-term memory constraint and thus minimizes the learning. The updated RNN architecture, known as long short-term memory (LSTM), resolves this problem by making use of a memory cell and gates to regulate the flow of information [24].

A two-layer LSTM, i.e., Bi-LSTM, model is developed for this research work, comprising two layers stacked on top of each other. The first hidden layer in LSTM 1 contains 128 neurons followed by a leaky ReLU layer and then a dropout layer to prevent overfitting. The second LSTM layer with 64 neurons is introduced followed by a dropout layer and a dense layer, respectively. The dense layer is the deeply connected output layer of this

network, which receives input from all previous hidden layers. The number of units in the dense layer depends on the prediction horizon (PH) for which the model is trained, which are 6 for 30 min PH and 12 for 60 min PH. The developed Bi-LSTM structure for 30 min blood glucose level forecasting is shown in Figure 11. The figure is generated using the software '**Netron'**, which is a utility to visualize the trained deep learning models. The model expects data to have the following shape: [*examples, sample per example, number of features*], which the software produced as [? × 6 ×1]. Figure 11 depicts the proposed model architecture that is trained using a single feature with six lag samples (timestamps) to predict the output.

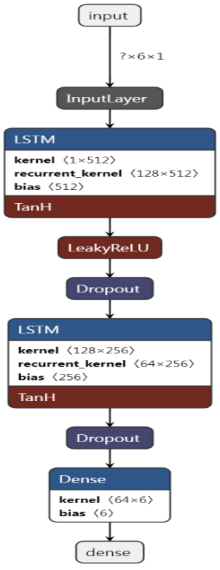


Figure 11. Bi-LSTM model architecture for blood glucose forecasting at 30 min prediction horizon with 6 samples per example and single feature. The size of input data is '? × 6 × 1′ where '?' represents the number of examples fed to the model.

To understand the input data volume, we consider an example of patient ID 570 with a univariate (one input feature: Glucose level) model, which has the following three-dimensional training and test set shapes:

- Total Number of training samples = 10,982.
- Total Number of training samples after resampling and interpolation = 11,611.
- Number of training samples (80%) = 9288 [1548, 6, 1].
- Number of training examples = 1548 (9288/6).
- Number of test samples (20%): 2322 [387, 6, 1].
- Number of test examples 387 (2322/6).

The training set has 1548 samples with one feature and six lag values. In other words, 9288 timestamps in the training data are converted into 1548 training examples with 6 samples (timestamps) per example. Similarly, there are 387 test examples with 6 samples per test example. The output of the model is 6 samples for a prediction horizon of 30 min as shown in Figure 11.

Similarly, '? × 12 × 2′ represents the data input into the model for 60- min prediction (12 samples per example) using 2 features. Likewise, the output of the model for the 60 min prediction horizon will be 12 samples.

In the end, the dense layer produces a vector of 6 values, which represents the prediction for a 30 min horizon. For example:

Input→Output:
[[152. 155. 156. 158. 161. 164.]]=====>[[168. 170. 171. 173. 174. 175.]]

The model takes six previous samples of single features as the input and predicts the six-step-ahead values of the target variable.

Moreover, the vanilla model with a single LSTM layer is also used separately.

5.2. Model Training and Optimization

The Ohio T1DM dataset includes pre-split files of each contributor as the training data and test data. In this research, the test data are used to finally evaluate the results, while the training data are subdivided for training and validation using the time-series cross-validator [25] provided by the Scikit-learn library of python. It splits the data based on the rolling origin, where each successive training set is a superset of those that come before them. The training data were split into training and validation sets. The five splits were made where the validation data size remains constant, but the training data size increases. For instance, the number of training samples visualized in Figure 12 is 1936, 3871, 5806, 7741, and 9676 in splits 1, 2, 3, 4, and 5, respectively. The number of validation samples remains at 1935 in each split; however, the validation samples themselves change every time. The data-splitting strategy is illustrated in Figure 12. The data for each split are visualized in Figure 13. After normalizing the training and validation data using a standard scalar, the trained model is evaluated using validation data, and the mean score of five splits is recorded. While training, the number of input samples was also altered using the sliding window mechanism with a stride of 1 and a variable window size of half an hour to 6 h. The optimum input sample size was observed as 6, i.e., the last half hour, based on the best mean prediction score.

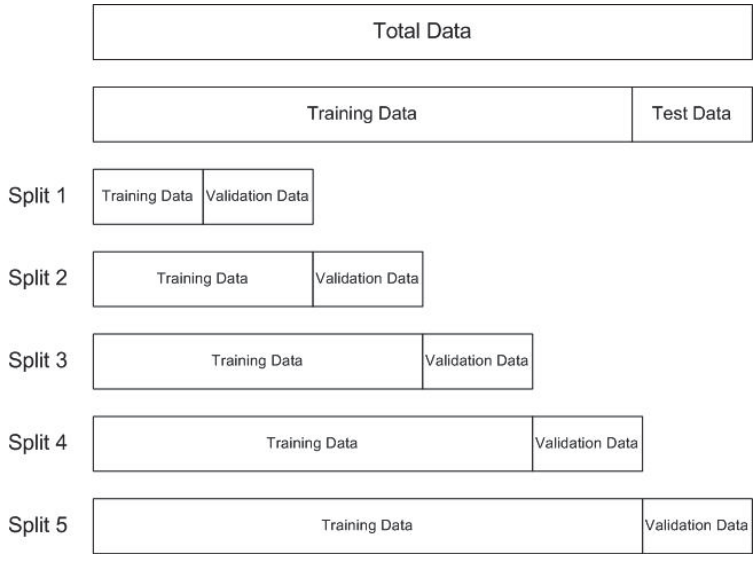


Figure 12. Data splits used for training the model.

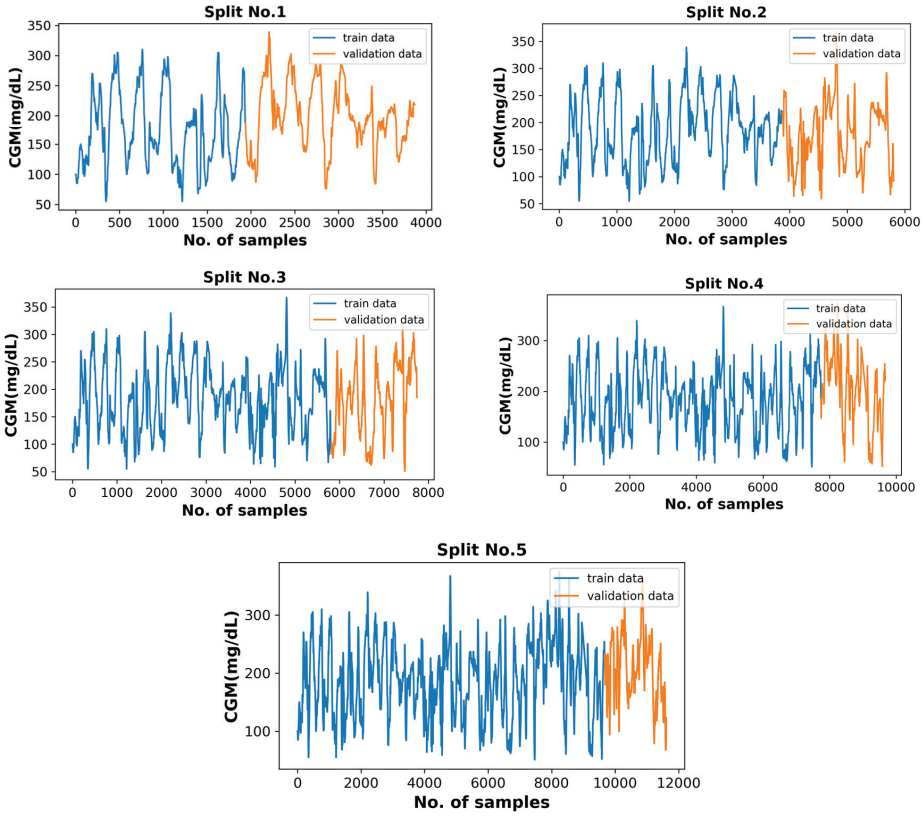


Figure 13. The five combinations of training–validation splits for CGM data of patient ID 570.

As discussed earlier, both vanilla- and Bi-LSTM RNN models are employed; the performance is evaluated based on the blood glucose prediction. The blood glucose regulation system is nonlinear in nature; therefore, to best approximate the input–output relationship, several hyper parameters were adjusted to achieve the optimum results, such as the number of neurons in each LSTM layer, the batch size, the learning rate, and the number of epochs. Different batch sizes including 32, 64, and 128 were used in the experiments where 128 produced the best validation results. Likewise, different combinations of the number of neurons in the LSTM layers were tested and resulted in 128 and 64 neurons in the first and second LSTM layers, respectively. The Adam optimizer was used with an optimized learning rate of 0.001 to train the model. Once all the model parameters are optimized, the final model is evaluated using the walk-forward validation scheme [26] on the test data.

6. Results and Discussion

The model results are recorded using the test data provided in the dataset. Each of the three input feature configurations are used to produce the prediction results. The root mean square error is computed to evaluate the model performance. The continuous glucose monitoring (CGM) prediction results of patient ID 570 for 30-min and 60-min prediction horizon are shown in Figures 14 and 15 respectively. Figure 16 shows the complete test data forecasting results for patient ID 570.

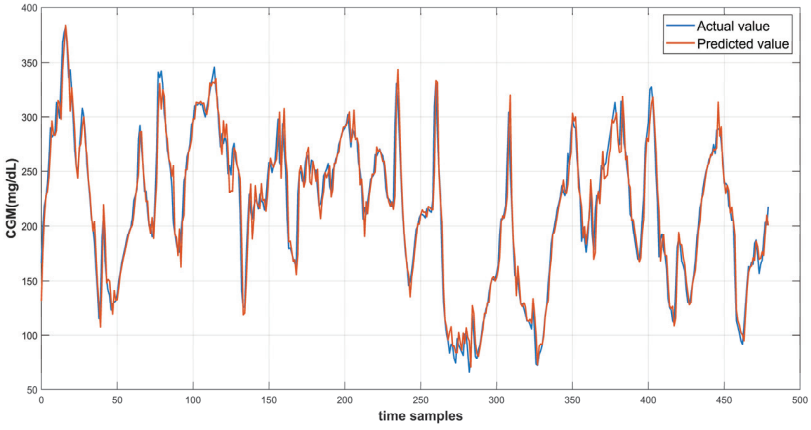


Figure 14. Prediction curve for patient ID 570 at PH of 30 min.

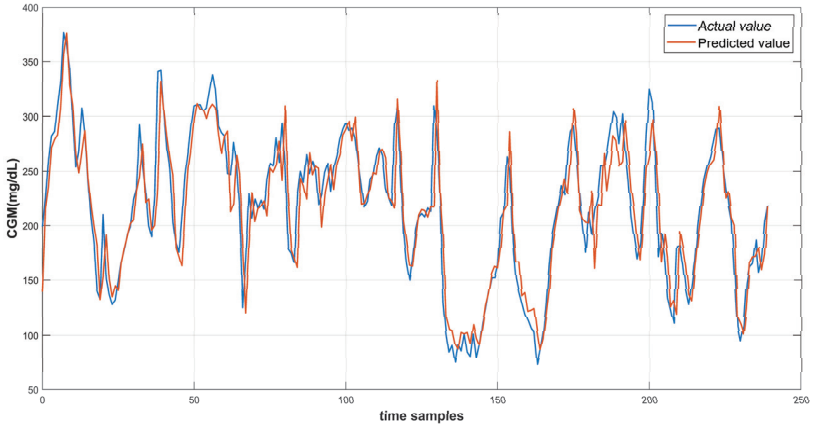


Figure 15. Prediction curve for patient ID 570 at PH of 60 min.

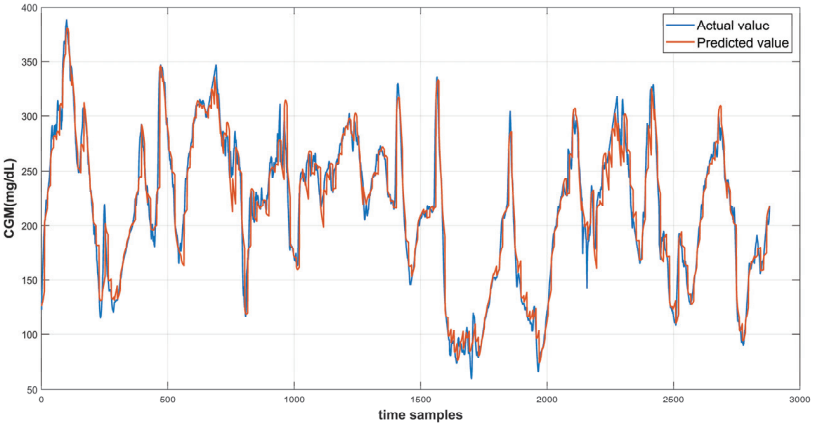


Figure 16. Prediction curve for complete test data of Patient ID 570.

Table 3 shows the model's prediction error for each of the three feature configurations over 30-min as well as 60-min PH for patient ID 570. The performance of the model with Bi-LSTM shows better results than vanilla LSTM due to its capability of memorizing the long-term temporal dependencies.

Table 3. Prediction error using different input feature configurations at 30 min and 60 min PHs for patient 570.

Configuration	RMSE @ PH = 30 min		RMSE @ PH = 60 min	
	Vanilla-LSTM	Bi-LSTM	Vanilla-LSTM	Bi-LSTM
C-01	15.43	15.22	26.41	26.10
C-02	15.67	15.12	26.12	25.48
C-03	15.48	14.76	26.18	25.65

The model with an input configuration of C-03 = [CGM, Carbs(Cop), Insulin(Ieff)] produced the best results for the PH of 30 min while configuration C-02 = [CGM, Carbs(Cop)] outperformed for the 60 min PH. Based on these results for patient ID 570, the blood glucose prediction results of other patients were recorded by employing a, similar feature configuration pattern, i.e., configuration C-03 for PH of 30 min and C-02 for 60 min. Table 4 presents the error results produced by the trained model for patients ID 588 and 563. The Bi-LSTM model outperformed the vanilla-LSTM model. A comparison of the proposed model with existing studies for the 30 min prediction horizon is presented in Table 5. It can be observed that the proposed methodology outperformed the state-of-the-art tool except for patient ID 563 where the margin was almost negligible. Moreover, the mean RMSE across the data of three contributors was computed, and the proposed methodology produced the minimum error.

Table 4. Prediction error for patient ID 588 and 563.

Patient ID	Configuration	RMSE @ PH = 30 min		RMSE @ PH = 60 min	
		Vanilla-LSTM	Bi-LSTM	Vanilla-LSTM	Bi-LSTM
588	C-02	×	×	30.44	30.17
	C-03	18.29	17.55	×	×
563	C-02	×	×	29.98	29.11
	C-03	18.58	18.14	×	×

Table 5. Comparison of prediction RMSE with other studies at prediction horizon of 30 min.

Patient ID	Bi-LSTM (Prposed Work)	[11]	[4]	[5]	[9]
570	14.76	15.94	16.40	18.26	15.73
588	17.55	17.71	19.20	21.69	17.66
563	18.14	18.12	19.00	20.17	18.51
Mean RMSE	16.81	17.25	18.20	20.04	17.30

Among existing studies, for this dataset, several performed blood glucose prediction using the prediction horizon of 60 min. Their results are compared with the proposed model results for the three patients as shown in Table 6. The proposed model produced less prediction error than the state of the art for the prediction horizon of 60 min.

Table 6. Comparison of prediction RMSE with other studies at prediction horizon of 60 min.

Patient ID	Bi-LSTM (Proposed Work)	[18]	[4] (Martinsson, et al., 2018)
570	25.48	25.74	28.6
588	30.17	30.45	33.1
563	29.11	30.42	29.9
Mean RMSE	28.25	28.87	30.53

Although the achieved error margin is relatively small with the state of the art, considering the highly non-linear nature of the blood glucose profile, the slight improvement in forecasting becomes significant. The existing studies used features such as insulin and meals as event-based features. However, the proposed methodology incorporates the transformation of event-based features into continuous values, which led to improving the prediction performance.

7. Conclusions

Efficient blood glucose forecasting is critical for continuous monitoring in diabetic patients. This research proposed an efficient method for accurate continuous glucose monitoring. The data of three contributors in the Ohio T1DM (2018) dataset were used for the performance evaluation of the proposed algorithm. Preprocessing of data was performed including interpolation and filtering. Two important event-based input features were transformed into continuous features that are inherent to the critical relationship with the blood glucose level. Recurrent-neural-network-based LSTM models were trained for blood glucose forecasting with prediction horizons of 30 and 60 min. The proposed models outperformed the state-of-the-art tools, achieving an RMSE as low as 14.76 mg/dL and 25.48 mg/dL for prediction horizons of 30 min and 60 min, respectively. This research will help practitioners and patients in closed-loop systems such as artificial pancreas/automated insulin-delivery systems. It will assist people with diabetes in predicting and managing their blood glucose levels, potentially reducing the risk of complications such as hypoglycemia (low blood sugar) or hyperglycemia (high blood sugar). It was observed that our model generated larger errors approximating sharp transitions in the CGM. Moreover, interpolated data may not reflect the true nature of the blood glucose profile. Therefore, efficient approximation of missing samples and the introduction of discriminative features for improved performance are possible future work directions.

Author Contributions: Conceptualization, H.B. and I.K.; methodology, H.B., I.K., and M.A.I.; validation, H.B. and I.K.; formal analysis, I.K. and M.A.I.; software, H.B. and I.K.; supervision, I.K.; writing—original draft, H.B. and I.K.; writing—review and editing, H.B, I.K and M.A.I. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data is publically available from the following link: https://pubmed.ncbi.nlm.nih.gov/33584164/.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Atlas, I.D. Global Estimates for the Prevalence of Diabetes for 2015 and 2040. *Diabetes Res. Clin. Pract.* **2017**, *128*, 40–50.
2. Elflein, J. *Number of Adults with Diabetes in the U.S. as of 2019 (in Millions)*; Statista: Hamburg, Germany, 2022.
3. Woldaregay, A.Z.; Årsand, E.; Walderhaug, S.; Albers, D.; Mamykina, L.; Botsis, T.; Hartvigsen, G. Data-Driven Modeling and Prediction of Blood Glucose Dynamics: Machine Learning Applications in Type 1 Diabetes. *Artif. Intell. Med.* **2019**, *98*, 109–134. [CrossRef] [PubMed]

4. Oviedo, S.; Vehí, J.; Calm, R.; Armengol, J. A Review of Personalized Blood Glucose Prediction Strategies for T1DM Patients. *Int. J. Numer. Methods Biomed. Eng.* **2017**, *33*, e2833. [CrossRef] [PubMed]
5. Sun, X.; Rashid, M.; Sevil, M.; Hobbs, N.; Brandt, R.; Askari, M.R.; Shahidehpour, A.; Cinar, A. Prediction of Blood Glucose Levels for People with Type 1 Diabetes Using Latent-Variable-Based Model. *CEUR Workshop Proc.* **2020**, *2675*, 115–119.
6. Xie, J.; Wang, Q. Benchmark Machine Learning Approaches with Classical Time Series Approaches on the Blood Glucose Level Prediction Challenge. *CEUR Workshop Proc.* **2018**, *2148*, 97–102.
7. Zecchin, C.; Facchinetti, A.; Sparacino, G.; Cobelli, C. Jump Neural Network for Online Short-Time Prediction of Blood Glucose from Continuous Monitoring Sensors and Meal Information. *Comput. Methods Programs Biomed.* **2014**, *113*, 144–152. [CrossRef]
8. McShinsky, R.; Marshall, B. Comparison of Forecasting Algorithms for Type 1 Diabetic Glucose Prediction on 30 and 60-Minute Prediction Horizons. *CEUR Workshop Proc.* **2020**, *2675*, 12–18.
9. Midroni, C.; Leimbigler, P.J.; Baruah, G.; Kolla, M.; Whitehead, A.J.; Fossat, Y. Predicting Glycemia in Type 1 Diabetes Patients: Experiments with XGBoost. *CEUR Workshop Proc.* **2018**, *2148*, 79–84.
10. Dave, D.; Erraguntla, M.; Lawley, M.; DeSalvo, D.; Haridas, B.; McKay, S.; Koh, C. Improved Low-Glucose Predictive Alerts Based on Sustained Hypoglycemia: Model Development and Validation Study. *JMIR Diabetes* **2021**, *6*, e26909. [CrossRef]
11. Martinsson, J.; Schliep, A.; Eliasson, B.; Meijner, C.; Persson, S.; Mogren, O. Automatic Blood Glucose Prediction with Confidence Using Recurrent Neural Networks. *CEUR Workshop Proc.* **2018**, *2148*, 64–68.
12. Zhu, T.; Li, K.; Herrero, P.; Chen, J.; Georgiou, P. A Deep Learning Algorithm for Personalized Blood Glucose Prediction. *CEUR Workshop Proc.* **2018**, *2148*, 64–78.
13. Li, K.; Liu, C.; Zhu, T.; Herrero, P.; Georgiou, P. GluNet: A Deep Learning Framework for Accurate Glucose Forecasting. *IEEE J. Biomed. Health Inform.* **2020**, *24*, 414–423. [CrossRef]
14. Daniels, J.; Herrero, P.; Georgiou, P. Personalised Glucose Prediction via Deep Multitask Networks. *CEUR Workshop Proc.* **2020**, *2675*, 110–114.
15. Caruana, R. Multitask Learning. *Mach. Learn.* **1997**, *28*, 41–75. [CrossRef]
16. Rabby, M.F.; Tu, Y.; Hossen, M.I.; Lee, I.; Maida, A.S.; Hei, X. Stacked LSTM Based Deep Recurrent Neural Network with Kalman Smoothing for Blood Glucose Prediction. *BMC Med. Inform. Decis. Mak.* **2021**, *21*, 101. [CrossRef]
17. Staal, O.M.; Salid, S.; Fougner, A.; Stavdahl, O. Kalman Smoothing for Objective and Automatic Preprocessing of Glucose Data. *IEEE J. Biomed. Health Inform.* **2019**, *23*, 218–226. [CrossRef]
18. Marling, C.; Bunescu, R. The OhioT1DM Dataset for Blood Glucose Level Prediction: Update 2020. *CEUR Workshop Proc.* **2020**, *2675*, 71.
19. Chen, J.; Li, K.; Herrero, P.; Zhu, T.; Georgiou, P. Dilated Recurrent Neural Network for Short-Time Prediction of Glucose Concentration. *CEUR Workshop Proc.* **2018**, *2148*, 69–73.
20. Kraegen, E.W.; Chisholm, D.J.; McNamara, M.E. Timing of Insulin Delivery with Meals. *Horm. Metab. Res.* **1981**, *13*, 365–367. [CrossRef]
21. Boiroux, D.; Finan, D.A.; Jørgensen, J.B.; Poulsen, N.K.; Madsen, H. Optimal Insulin Administration for People with Type 1 Diabetes. *IFAC Proc. Vol.* **2010**, *43*, 248–253. [CrossRef]
22. LoopDoc. Glucose Prediction. Available online: https://loopkit.github.io/loopdocs/operation/algorithm/prediction/ (accessed on 3 February 2022).
23. Manaswi, N.K. RNN and LSTM. In *Deep Learning with Applications Using Python*; Apress: Berkeley, CA, USA, 2018. [CrossRef]
24. Hochreiter, S. Recurrent Neural Net Learning and Vanishing Gradient. *Int. J. Uncertainity Fuzziness Knowl. Based Syst.* **1998**, *6*, 8.
25. TimeSeriesSplit. Sklearn.Model_selection.TimeSeriesSplit—Scikit-Learn 1.2.0 Documentation. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection (accessed on 19 December 2022).
26. Brownlee, J. *Introduction to Time Series Forecasting with Python: How to Prepare Data and Develop Models to Predict the Future*; Machine Learning Mastery: Vermont, Australia, 2017; Available online: https://books.google.com.pk/books/about/Introduction_to_Time_Series_Forecasting.html?id=-AiqDwAAQBAJ&redir_esc=y (accessed on 19 December 2022).

diagnostics

MDPI

Article

White Blood Cells Classification Using Entropy-Controlled Deep Features Optimization

Riaz Ahmad [1,2], Muhammad Awais [3,*], Nabeela Kausar [1] and Tallha Akram [3]

[1] Department of Computer Science, Iqra University, Islamabad 44800, Pakistan
[2] Department of Computer Science, COMSATS University Islamabad, Wah Campus, Wah Cantt 47010, Pakistan
[3] Department of Electrical & Computer Engineering, COMSATS University Islamabad, Wah Campus, Wah Cantt 47010, Pakistan
* Correspondence: muhammadawais@ciitwah.edu.pk

Abstract: White blood cells (WBCs) constitute an essential part of the human immune system. The correct identification of WBC subtypes is critical in the diagnosis of leukemia, a kind of blood cancer defined by the aberrant proliferation of malignant leukocytes in the bone marrow. The traditional approach of classifying WBCs, which involves the visual analysis of blood smear images, is labor-intensive and error-prone. Modern approaches based on deep convolutional neural networks provide significant results for this type of image categorization, but have high processing and implementation costs owing to very large feature sets. This paper presents an improved hybrid approach for efficient WBC subtype classification. First, optimum deep features are extracted from enhanced and segmented WBC images using transfer learning on pre-trained deep neural networks, i.e., DenseNet201 and Darknet53. The serially fused feature vector is then filtered using an entropy-controlled marine predator algorithm (ECMPA). This nature-inspired meta-heuristic optimization algorithm selects the most dominant features while discarding the weak ones. The reduced feature vector is classified with multiple baseline classifiers with various kernel settings. The proposed methodology is validated on a public dataset of 5000 synthetic images that correspond to five different subtypes of WBCs. The system achieves an overall average accuracy of 99.9% with more than 95% reduction in the size of the feature vector. The feature selection algorithm also demonstrates better convergence performance as compared to classical meta-heuristic algorithms. The proposed method also demonstrates a comparable performance with several existing works on WBC classification.

Keywords: deep learning; nature-inspired feature selection; leukemia; CNN; white blood cell; classification; medical imaging

Citation: Ahmad, R.; Awais, M.; Kausar, N.; Akram, T. White Blood Cells Classification Using Entropy-Controlled Deep Features Optimization. *Diagnostics* **2023**, *13*, 352. https://doi.org/10.3390/diagnostics13030352

Academic Editors: Wan Azani Mustafa and Hiam Alquran

Received: 23 November 2022
Revised: 13 January 2023
Accepted: 13 January 2023
Published: 18 January 2023

1. Introduction

Blood is a crucial fluid in the human body that is essential for life. Human blood is made up of plasma and blood cells. Plasma is the yellowish liquid component of blood that is largely water and accounts for 55% of blood volume [1]. The blood also includes proteins, carbohydrates, minerals, hormones, carbon dioxide, and blood cells. Red blood cells (RBCs), white blood cells (WBCs), and platelets (thrombocytes) are the three different types of cellular components found in the blood, each distinguished by their color, texture, and appearance. RBCs, also known as erythrocytes, carry hemoglobin, an iron-containing protein that aids in the delivery of oxygen from the lungs to the tissues. WBCs, also known as leukocytes, are an essential component of the human immune system, assisting the body in fighting infectious diseases and foreign substances [2,3]. Figure 1 demonstrates a classification of WBCs on the basis of their structure. WBCs are primarily of two types, i.e., granulocytes and agranulocytes. The granulocytes have their origin in the bone marrow and are present within the cytoplasm in the form of granules of protein. There are three types of granulocyte cells, namely basophils, eosinophils, and neutrophils. Agranulocytes,

which are defined as cells without granules in their cytoplasm, are further divided into two types, i.e., lymphocytes and monocytes [4]. Each type of cell has a unique role in the immune system of the body. For example, neutrophils act as scavengers that surround and destroy bacteria and fungi present in the body. Eosinophils play a role in the general immune and inflammatory responses of the body. An increased level of basophils results in a blood disorder after an allergic reaction. Monocytes fight against infections, remove dead or damaged tissues, and kill cancerous cells; lymphocytes combat bacteria, viruses, and other cells that pose a threat to the body's ability to function [5]. A detailed analysis of WBCs is very important to assess the overall condition of the human immune system. In particular, WBC analysis is crucial in the diagnosis of leukemia, a type of blood cancer that occurs due to the excessive production of malignant WBCs in the bone marrow. Leukemia diagnosis is performed by one of three main clinical tests, i.e., physical test, complete blood count (CBC) test, and bone marrow test. The first step of CBC is to determine different types of WBCs from the blood samples. This task is mainly performed by hematologists through the visual examination of microscopic images of blood smears. This manual method is labor-intensive, time-consuming, and prone to inaccuracy due to judgment errors influenced by several external factors.

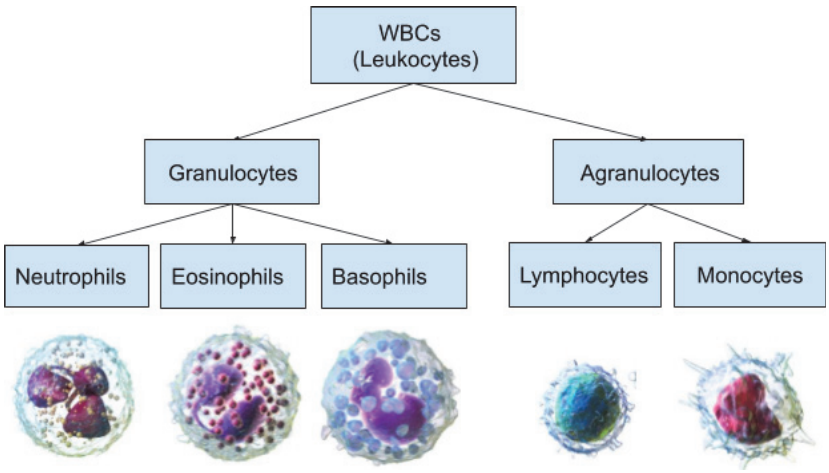


Figure 1. White blood cell subtypes.

With the recent advancement in digital image processing technology, the automated classification of WBCs using computer vision techniques has attracted significant research interest. However, due to morphological overlap between different subclasses and their structural irregularities, the machine learning-based classification and localization of WBCs is challenging. Deep learning with convolutional neural networks (CNNs) is the most promising method for classification and detection tasks in the field of contemporary medical imaging [6,7]. Despite the fact that CNNs perform best on large datasets, training them takes a lot of data and computational power. The dataset is frequently small and may not be sufficient to train a CNN from scratch. In such a case, transfer learning is frequently used to maximize the effectiveness of CNNs while also decreasing the computational costs [8]. In this approach, the CNN is initially pre-trained on a large dataset consisting of a diverse range of classes and then applied to a specific task [9]. There are various pre-trained neural networks that have won international contests, including VGGNet [10], Resnet [11], Darknet [12], Densenet [13], Mobilenet [14], Inception [15], Xception, [16] etc. Through their capacity for self-learning, these models are able to extract a rich set of features from images that contain substantial semantic information. This helps to achieve a significant level of accuracy for a variety of image classification scenarios. In modern deep learning applications, feature selection is a crucial step which reduces the difficulty

of model learning by removing irrelevant or redundant features. The present research is focused on achieving a high level of accuracy with a smaller feature set to reduce the computation costs and memory requirements of expert systems.

The existing works on WBC classification are broadly classified into two categories, i.e., (a) classical methods and (b) deep learning methods. The classical methods consist of approaches which propose efficient preprocessing techniques to extract strong features from WBC images and classify them using baseline classifiers. Some remarkable works in this domain are discussed as follows. In [17], the authors proposed a method which selects the eigenvectors from color images of blood cells based on the minimization of similarities. The Bayesian classifier is then used to classify the eigen cells on the basis of density and color information. In [18], Fuzzy C-means clustering is applied to separate the nucleus and cytoplasm of leukocytes. Then, various geometric, color, and statistical properties are extracted and classified by support vector machines (SVMs). In [19], an image segmentation method is proposed based on mean-shift clustering and boundary removal rules with a gradient vector flow. An ensemble of features is extracted from the segmented nucleus and cytoplasm, which is then classified using a random forest algorithm. In [20], the authors tested the performance of six different machine learning algorithms on 35 different geometric and statistical features. The multinomial logistic regression algorithm outperformed other methods. A stepwise linear discriminant analysis method is proposed in [21], which extracts specific features from blood structure images and classifies them using reversion values such as partial F values. In [22], the authors presented a WBC cancer detection method which combines various morphological, clustering, and image pre-processing steps with random forest classifier. The suggested method uses a decision tree learning method, which uses predictors at each node to make better decisions, in order to categorize various types of cancer.

The second category of works is based on deep learning approaches for WBC classification. The works in this category primarily employ transfer learning of a pretrained deep neural network for feature extraction or classification. Some important works are discussed as follows. In [23], the authors proposed a deep learning method that uses the DenseNet121 [13] model to classify WBC subtypes. The model is optimized with the preprocessing techniques of normalization and data augmentation applied to a Kaggle dataset. The work in [24] first applies a thresholding-based segmentation on the WBC images. Feature extraction from segmented images is performed using VGG16 CNN [10] model learning. The extracted feature vectors are classified using the K-nearest neighbor (KNN) algorithm. In [25], the authors investigated generative adversarial networks (GANs) for data augmentation and employed the DenseNet169 [13] network for WBC classification. In [26], the authors applied Gaussian and median filtering before training the images using multiple deep neural networks. The authors in [27] applied a you-only-look-once (YOLO) algorithm for the detection of blood cells from a smear images. In [28], two techniques are proposed for blood cell identification, namely single-shot multibox detector and an incrementally improved version of YOLO.

Although modern approaches based on transfer learning on deep CNN models achieve a decent level of accuracy for a variety of classification tasks, they all share the use of a large number of features extracted from deep neural networks. This suffers from high computational cost and memory requirements for practical deployment. In most biomedical scenarios, many of these deep features are redundant or contain zeros. Effective dimensionality reduction, or choosing only powerful, discriminant features, increases classifier accuracy while decreasing computational time and expense. WBC classification using deep feature selection is an emerging research area. Few works have reported population-based meta-heuristics for deep feature selection. The authors of [29] have proposed a leukemia detection system in which various features, such as color, texture, shape, and hybrid features, are first extracted from WBC images and then an optimization algorithm inspired by social spiders is used to select the most useful features. In [30], a leukemia detection approach is proposed which combines deep feature extraction using VGGNet

and a statistically enhanced salp swarm algorithm for feature selection. Furthermore, the classification of reduced feature vectors was performed using a baseline classifier. The work in [31] proposes a self-designed neural network named W-Net to classify five subtypes of WBCs. The authors also generated a synthetic WBC dataset using a generative adversarial network (GAN).

In this study, we have proposed a hybrid approach for WBCs classification. The proposed approach first creates an ensemble of deep features extracted by applying transfer learning of multiple deep CNNs on WBC images and then performs feature selection using an entropy-controlled nature-inspired algorithm. The main contributions of this work can be summarized in the following steps.

1. Using a synthetic, real-world, large-scale dataset of five WBC types, transfer learning is performed using two deep CNNs, namely Darknet53 and Densenet201, followed by their feature fusion;
2. For feature selection, a nature-inspired meta-heuristic named entropy-controlled marine predators algorithm (ECMPA) is proposed. The proposed algorithm effectively selects only the most dominant features;
3. The reduced feature set is classified using various baseline classifiers with multiple kernel settings;
4. The proposed feature selection algorithm demonstrates a high accuracy with significant reduction in feature size. The algorithm also achieves a better convergence rate as compared to classical population-based selection methods.

The main focus of our manuscript is to present a novel method of deep-feature selection using an entropy-controlled population-based algorithm and show its effectiveness in the domain of WBCs classification for leukemia detection. Since the definition of appropriate image features is a very difficult task due to the morphological similarity of images and subject variability, WBC classification is a pertinent design case for such an approach. The rest of this paper is organized as follows. Section 2 discusses all steps of the proposed WBC classification pipeline, Section 3 presents the results and analysis, and Section 4 concludes the paper.

2. Materials and Methods

This section provides a description of all steps of the proposed WBC classification system which are discussed in the following subsections.

2.1. Dataset Description

This work uses the public dataset in [31], which was generated synthetically from a real-world dataset [32] of five sub-types of WBC images, namely neutrophil, eosinophil, basophil, lymphocyte, and monocyte. The synthetic dataset was generated with the help of a deep convolutional generative adversarial network (DCGAN) trained on the original real-world dataset [32] of blood smear images, captured by a Sysmex DI-69 machine and provided by the Catholic University of Korea. The synthetic dataset is composed of 5000 images each of size ($128 \times 128 \times 3$), with 1000 images belonging for each class. Figure 2 shows samples belonging to all classes of the dataset of [32] used in this work.

2.2. WBCs Classification Pipeline

Figure 3 shows the proposed WBCs classification pipeline, whose main computation steps are discussed as follows.

2.2.1. Preprocessing

Image contrast enhancement is a fundamental pre-processing step of many digital image-processing applications. In this work, the input image enhancement is performed with the help of color histogram equalization. The classical method of histogram equalization is applied to grayscale images and performs as redistribution of their intensity. In case of color images, performing histogram equalization on R, G, and B components

independently will not necessarily enhance the image. Color histogram equalization can be achieved by converting a color image into a HSV/HSI image and enhancing the intensity while preserving hue and saturation components. The main steps of color histogram equalization are as follows.

- Convert the RGB image into HSI image;
- Obtain the intensity matrix from the HSI image matrix;
- Perform histogram equalization on the intensity matrix;
- Replace the intensity matrix of the HSI image with the histogram-equalized intensity matrix;
- Convert HSI image back to RGB image.

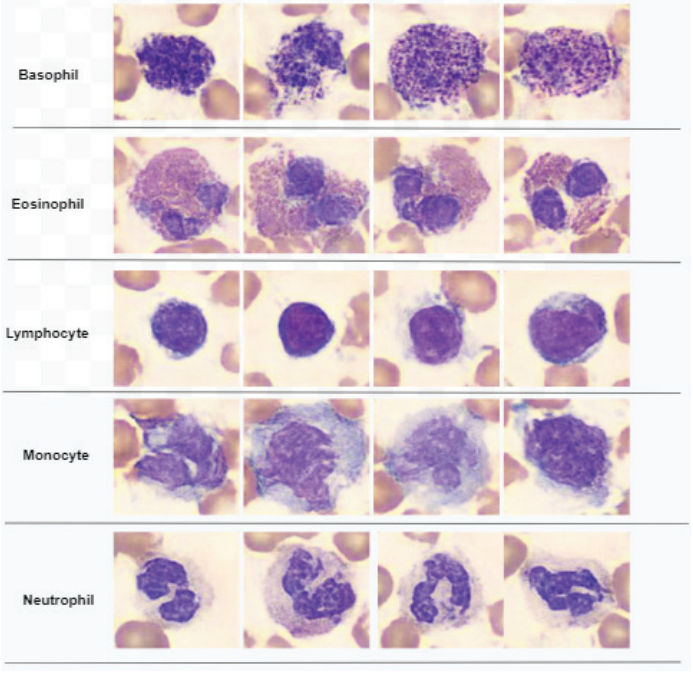


Figure 2. Samplesof WBC images of dataset used in this work.

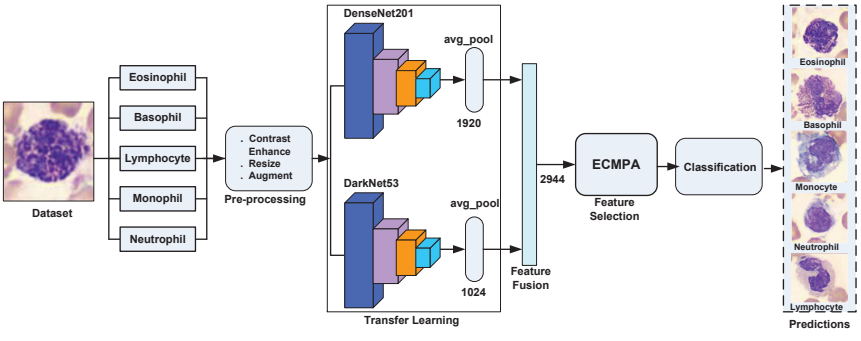


Figure 3. Pipeline of proposed WBCs classification system.

2.2.2. Feature Extraction Using Transfer Learning

The pre-processed image dataset is now subjected to the feature extraction using the transfer learning step. In this work, we performed transfer learning on two well-known deep CNNs, namely DarkNet53 and DenseNet201, which are discussed as follows.

DarkNet53 is a convolutional neural network proposed as a feature extractor in YOLO3 image detection workflow [12]. It is pretrained on more than a million images from ImageNet database [33]. The pretrained network is able to classify up to 1000 categories of image objects. Details about the various layers in the DarkNet CNN architecture are shown in Table 1. The network has an input layer with a size of $256 \times 256 \times 3$ and is primarily made up of convolution layers with sizes of 1×1 and 3×3, totaling 53 layers, including the final fully connected layer but excluding the residual layer. Each convolutional layer is composed of a Conv2d layer followed by a batch normalization (BN) [34] and LeakyReLU [11] layer. The residual layer is added to solve the gradient disappearance or gradient explosion problems in the network [12]. In Darknet53, a significant reduction in parameters is achieved as compared to its previous version, i.e., Darknet19.

Table 1. DarkNet53 layer architecture.

Layer Type	Filters	Filter Size	Stride Size	Repeat	Output Size
Input	-	-	-	-	224×256
Convolutional	32	3×3	1	1	256×256
Convolutional	64	3×3	2	1	128×128
Convolutional	32	1×1	1		
Convolutional	64	3×3	1	1	
Residual					128×128
Convolutional	128	3×3	2	1	64×64
Convolutional	64	1×1	1		
Convolutional	128	3×3	1	2	
Residual					64×64
Convolutional	256	3×3	2	1	32×32
Convolutional	128	1×1	1		
Convolutional	256	3×3	1	8	
Residual					32×32
Convolutional	512	3×3	2	1	16×16
Convolutional	256	1×1	1		
Convolutional	512	3×3	1	8	
Residual					16×16
Convolutional	1024	3×3	2	1	8×8
Convolutional	512	1×1	1		
Convolutional	1024	3×3	1	4	
Residual					8×8
GlobalAvgPool					
Fully Connected			1000		
Softmax					

In order to perform transfer learning of Darknet53, the last learnable layer of Darknet53, i.e., "Conv5", is replaced with a new fully connected layer with the number of outputs equal to the number of classes in our WBCs dataset (5 classes). A new softmax layer is created and replaced with the original softmax layer of the network. Similarly, the classification layer of the network is replaced with a new classification layer without class labels. To perform the network training, first the dataset images are resized to $256 \times 256 \times 3$ using the nearest neighbor interpolation method, followed by image augmentation which performs

random rotation of images in the range of $[0, 360]$ degrees and scaling with a factor in the range of $[0.5, 1]$. The deep features are extracted from the "GlobalAvgPool" layer. The DarkNet53 returns a deep feature vector of size 1024 per image.

DenseNet201. This deep convolutional neural network is 201 layers deep [13]. It is also pre-trained on Imagenet [33] dataset. DenseNet is designed to overcome the vanishing gradient problem in high-level neural networks. In DenseNet, each layer receives new inputs from all preceding levels and passes on its own feature maps to all following layers. Concatenation is utilized. Each layer receives "collective knowledge" from all preceding levels. This results in a thinner and compact network that achieves a high computational efficiency and memory saving. Table 2 shows the layer details of DenseNet201.

Table 2. DenseNet201 layer architecture.

Layer Type	Composition	Repeat	OutSize
Input	–	–	224×224
Convolution	Conv(7×7), stride 2		112×112
MaxPool	(3×3), stride 2		56×56
Dense Block 1	Conv(1×1) Conv(3×3)	6	56×56
Transition Layer 1	Conv(1×1) Avg Pool(2×2), Stride 2	1	56×56 28×28
Dense Block 2	Conv(1×1) Conv(3×3)	12	28×28
Transition Layer 2	Conv(1×1) Avg Pool(2×2), Stride 2	1	28×28 14×14
Dense Block 3	Conv(1×1) Conv(3×3)	48	14×14
Transition Layer 3	Conv(1×1) Avg Pool(2×2), Stride 2	1	14×14 7×7
Dense Block 4	Conv(1×1) Conv(3×3)	32	7×7
Classification Layer	7×7 Global Avg. Pool 1000D fully Connected, softmax		1×1

In order to perform transfer learning using DenseNet201, the last learnable layer of the network, i.e., "fc1000" is replaced with a new fully connected layer with 5 classes of the WBCs dataset used in this work. A new softmax layer is created and replaced with the original softmax layer of the network. Similarly, the classification layer of the network is replaced with a new classification layer without class labels. The dataset images are first resized to 224×224 and augmented in a way similar to DarkNet53. From the trained network, the deep features are extracted from the global average pooling layer. DenseNet201 returns a deep feature vector of size 1920 per image. There was no layer freezing carried out during the training process. As a result, the highest possible numbers of trainable parameters, i.e., 18.1 million for DenseNet201 and 41.6 million for DarkNet53, were considered.

2.2.3. Feature Fusion

The deep features extracted from both the Darknet53 and Densenet201 networks discussed above are concatenated together to form a fused feature vector. Let X and Y be the feature vectors of Darknet53 and Densenet201, respectively, and the fused feature vector Z is given as

$$Z = [XY]$$

The total size of fused feature vector Z is 2944 features per image.

2.2.4. Feature Selection Using Marine Predators Algorithm

Feature selection is an important step which significantly alleviates the curse of dimensionality by reducing the size of the feature vector, selecting only relevant features. The classical feature selection methods based on search algorithms such as complete and sequential search suffer from high computational cost. Population-based meta-heuristic algorithms have been demonstrated as an effective way to solve complex optimization problems [35,36].

The marine predators algorithm (MPA) is a meta-heuristic algorithm that draws inspiration from nature and models the foraging behavior of marine predators (MPs) to find their prey [37]. In the aquatic environment, both the prey and the predator are looking for food at the same time. Position updates for the predator and prey are based on Brownian or Lévy movement, depending on the relative velocities of the two. The goal of MPA, like other swarm optimization techniques, is to choose the best solution (elite) from the population of predators. The MP with the strongest foraging capacity is called elite predator. The MPA is based on the observation that MPs move in Lévy patterns when there are few prey items present and in Brownian patterns when there are many prey items present. Additionally, predators alter their behavior and move to areas with different prey concentrations in the presence of environmental effects. As a result, there are three phases to the position updates in MPA optimization based on the relative velocities of predator and prey: low velocity ratio, unit velocity ratio, and high velocity ratio. The velocity ratio vs. is defined as the ratio between the velocity of prey and predator.

1. In low velocity ratio $(v <= 0.1)$, the most suitable movement strategy for the MP is Lévy, whereas the prey moves in Brownian or Lévy movement;
2. In unit velocity ratio $(v <= 1)$, if the prey moves in Lévy, the most suitable movement for MP is Brownian;
3. In high velocity ratio $(v > 1)$, the best strategy for a predator is not moving at all. In this case, either prey is moving Brownian or Lévy.

Standard MPA Methodology. The standard MPA is an iterative, population-based optimization algorithm. The first step is to generate an initial population of solutions. The population matrix of size $n \times d$ is generated as follows:

$$P = \begin{bmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,d} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,d} \\ \vdots & \vdots & \vdots & \vdots \\ X_{n,1} & X_{n,2} & \cdots & X_{n,d} \end{bmatrix} \quad (1)$$

where n is the size of population, i.e., number of search agents (each predator and prey are searching for food and can be considered as a search agent), and d is the dimension (no. of variables) of each agent. Each variable of initial solution is uniformly distributed over the search space computed as

$$X_i = l_b + rand \times (u_b - l_b) \quad (2)$$

where l_b denotes the lower bound, u_b denotes the upper bound, and $rand$ is a uniformly distributed random number from the interval $[0, 1]$. Based on the concept of survival of the fittest, the top predators have the best foraging capabilities. Therefore, the fittest solution is nominated as the best predator and used to construct a matrix called *Elite*. In an iteration I, the *Elite* matrix is constructed as

$$Elite = \begin{bmatrix} X^I_{1,1} & X^I_{1,2} & \cdots & X^I_{1,d} \\ X^I_{2,1} & X^I_{2,2} & \cdots & X^I_{2,d} \\ \vdots & \vdots & \vdots & \vdots \\ X^I_{n,1} & X^I_{n,2} & \cdots & X^I_{n,d} \end{bmatrix} = \begin{bmatrix} \overline{X} \\ \overline{X} \\ \vdots \\ \overline{X} \end{bmatrix} \quad (3)$$

where $\overline{X}$ is the top predator vector which is replicated n times to construct the *Elite* matrix. At the end of each iteration, the *Elite* matrix will be updated if the fittest predator of a population is replaced by another better predator. Another matrix named *Prey* is generated with the same dimensions as *Elite*. The Prey matrix is computed as

$$Prey = \begin{bmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,d} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,d} \\ \vdots & \vdots & \vdots & \vdots \\ X_{n,1} & X_{n,2} & \cdots & X_{n,d} \end{bmatrix} \tag{4}$$

where $X_{i,j}$ denotes the j-th dimension of i-th prey. During first iteration, the Prey matrix is equal to randomly generated population matrix P. In all subsequent iterations, the Prey is updated and its values are used to compute the *Elite* matrix. The update of the *Prey* matrix is carried separately in three phases of MPA optimization. These phases include:

Phase 1: This phase corresponds to the high velocity ratio and happens in the first $(\frac{1}{3})^{rd}$ of maximum iterations of algorithm where exploration is more significant. The update rule of this phase is given as:

$$\overline{Stepsize_i} = \overline{R_B} \bigotimes (\overline{Elite_i} - \overline{R_B} \bigotimes \overline{Prey_i}),\ \forall i = 1, \cdots, n \tag{5}$$

$$\overline{Prey_i} = \overline{Prey_i} + P.\overline{R} \bigotimes \overline{Stepsize_i},\ \forall i = 1, \cdots, n \tag{6}$$

where $\overline{Prey_i}$ is a vector of Prey matrix, $\overline{R_B}$ and $\overline{R}$ are vectors of dimensions d containing random numbers from Normal and Uniform distribution, respectively, P is constant value equal to 0.5, and $\bigotimes$ shows element-wise multiplication.

Phase 2: This phase corresponds to unit velocity ratio when predator and prey are moving at the same pace. This is the phase which occurs for intermediate $(\frac{1}{3})^{rd}$ of iterations, where exploration and exploitation matters. The update rules for this phase are given as follows:

$$\overline{Stepsize_i} = \overline{R_L} \bigotimes (\overline{Elite_i} - \overline{R_L} \bigotimes \overline{Prey_i}),\ \forall i = 1, \cdots, \frac{n}{2} \tag{7}$$

$$\overline{Prey_i} = \overline{Prey_i} + P\overline{R} \bigotimes \overline{Stepsize_i},\ \forall i = 1 \cdots, \frac{n}{2} \tag{8}$$

$$\overline{Stepsize_i} = \overline{R_B} \bigotimes (\overline{Elite_i} - \overline{R_B} \bigotimes \overline{Prey_i}),\ \forall i = \frac{n}{2} + 1, \cdots, n \tag{9}$$

$$\overline{Prey_i} = \overline{Elite_i} + P.CF \bigotimes \overline{Stepsize_i},\ \forall i = \frac{n}{2} + 1 \cdots, n \tag{10}$$

$$\tag{11}$$

where $\overline{R_L}$ is vector of size d containing random numbers based on Lévy distribution, $CF = (1 - \frac{I}{I_{Max}})^{(2\frac{I}{I_{Max}})}$ is an adaptive parameter to control the step size for predator movement, I is the current iteration, and I_{Max} is the maximum number of iterations.

Phase 3: This phase corresponds to low velocity ratio when predator is moving faster than prey. This scenario happens in the last $(\frac{1}{3})^{rd}$ iterations of the optimization phase where exploitation matters. The update rules for this phase are given as follows:

$$\overline{Stepsize_i} = \overline{R_L} \bigotimes (\overline{Elite_i} - \overline{Prey_i}),\ \forall i = 1, \cdots, n \tag{12}$$

$$\overline{Prey_i} = \overline{Elite_i} + P.CF \bigotimes \overline{Stepsize_i},\ \forall i = 1 \cdots, n \tag{13}$$

The next step is to model the behavioral change in MPs as a result of environmental effects. These effects are known as fish aggregating devices (FADs). The FADs are known as local optima; therefore, the prey and predators must perform longer jumps during simulation to avoid stagnation in local optima. The update of *Prey* matrix considering the FAD effect is mathematically represented as follows:

$$\overline{Prey_i} = \begin{cases} \overline{Prey_i} + CF[\overline{X_{min}} + \overline{R} \otimes (\overline{X_{max}} - \overline{X_{min}})] \otimes \overline{U} & r \leq FADs \\ \overline{Prey_i} + [FADs(1-r) + r](\overline{Prey_{r1}} - \overline{Prey_{r2}}) & r > FADs) \end{cases} \quad (14)$$

where $FADs = 0.2$ is the probability of occurrence of FAD effects, $\overline{U}$ is a randomly generated binary vector, r is the uniform number in $[0, 1]$, X_{max} and X_{min} are the vectors containing lower and upper bounds of dimensions, respectively, and $r1$, $r2$ are the random indices of the *Prey* matrix.

After the *Prey* matrix is updated using Equations (6)–(13), and incorporating the FAD effects of Equation (14), this matrix is evaluated for fitness function. The fitness of each solution of current iteration is compared to its equivalent solution in prior iteration. If the current solution is more fitted, it replaces the previous one. In the next iteration, the best solution of *Prey* is used to generate the *Elite* matrix and update the *Prey* matrix using Equations (6)–(13).

2.2.5. Entropy-Controlled MPA for Feature Selection

In this work, we proposed a multi-level feature selection algorithm named entropy-controlled marine predators algorithm (ECMPA). The proposed technique is based on two stages of feature selection, the first of which corresponds to feature reduction based on the entropy of the fused feature vector, followed by additional feature reduction based on the MPA. The main computational steps of ECMPA are discussed in Algorithm 1.

Notations: In Algorithm 1, matrices and vectors are represented as double struck characters (e.g., $\mathbb{F}$) and scalars are represented as normal letters.

The algorithm receives as inputs the fused feature matrix $\mathbb{F}$, the label vector $\mathbb{L}$, the entropy-controlled feature reduction parameter e_c, the maximum number of iterations I_{Max}, and the population size N. Other constant parameters specific to MPA are upper bound u_b, lower bound l_b, threshold t, constant P, Levy coefficient β, and fish aggregating devices effect F_{ADS}. The matrix $\mathbb{F}$ is of dimensions $n_t \times D$, where n_t is the number of training images and D is the number of features extracted from the feature fusion step. The first level of feature extraction, i.e., entropy-based, is performed by steps 4–7. Step 4 computes the entropy of each column of $\mathbb{F}$ and returns a vector $\mathbb{E}$ of size $1 \times D$. Step 5 sorts $\mathbb{E}$ in descending order, thus returning the sorted vector $\mathbb{E}_2$ along with indices stored in $\mathbb{I}$. Step 6 extracts the indices of the first e_c percentage of features with maximum entropy. Step 7 extracts features in these indices from $\mathbb{F}$ and stores them in $\mathbb{F}_2$. The task for generating initial population matrix $\mathbb{P}$ using Equation (1) is performed in Steps 11–15. Step 13 corresponds to Equation (2), where $rand()$ computes a random number from uniform distribution from interval $[0, 1]$. In Steps 19–26, the fitness function of each individual of $\mathbb{P}$ is computed using *CostFunction* and stored in fitness vector *Fit*. The best fitness is stored in fit_g and best individual is stored in $\mathbb{x}_{gb}$. The *Elite* matrix $\mathbb{E}$ is computed in Step 28, by performing the *Repeat* function which concatenates N copies of $\mathbb{x}_{gb}$ along the first dimension (i.e., column-wise). Steps 31–37 perform update of *Prey* matrix $\mathbb{P}$ according to Phase 1 (Equations (5) and (6)). In Step 32, the function $randn(1, D)$ returns a vector of size $1 \times D$, containing random numbers from Normal distribution. The Phase 2 of MPA according to update rules of Equations (7)–(11) is performed by Steps 39–57. In Step 43, $Levy(\beta, D)$ generates a vector of size D containing random numbers from Lévy Distribution. Steps 59-66 perform the $\mathbb{P}$ matrix update according to Equations (12) and (13) of Phase 3. In Steps 68–83, the FAD effects are added to the *Prey* according to update rules of Equation (14). In each iteration, Steps 85–92 correspond to the memory-saving step where the updated *Prey* matrix P is evaluated for *CostFunction* and best individual of $\mathbb{X}_{gb}$ is obtained. Steps 95–97 correspond to the output step where $\mathbb{S}_F$, i.e., a binary vector of size D is obtained. The indices of non-zero entries of $\mathbb{S}_F$ correspond to the indices of selected features of the fused feature vector.

In Steps 100–115 the execution of *CostFunction* is demonstrated, which performs the computation of fitness value of each individual solution. The function accepts as inputs the entropy-reduced feature vector $\mathbb{F}_2$, the label vector $\mathbb{L}$, and a binary vector $\mathbb{a}$ computed

by comparing the entries of ith solution of $\mathbb{P}$ with the threshold t. The Step 104 extracts all the features of $\mathbb{F}_2$ whose indices correspond to non-zero values of $\mathbb{a}$, and then the function *partition* splits the feature matrix $\mathbb{F}_2$ and label vector $\mathbb{L}$ into training feature set $\mathbb{F}_{train}$, testing feature set $\mathbb{F}_{test}$, training label set $\mathbb{L}_{train}$, and testing label set $\mathbb{L}_{test}$ with a splitting ratio h_o. In the subsequent steps of the function, model training and prediction is performed using KNN classifier and fitness value (cost) is computed using the classification error metric. An individual with a smaller *cost* value is the fitter individual.

Algorithm 1 ECMPA for feature selection.

Inputs: $\mathbb{F}, \mathbb{L}, e_c, I_{Max}, N$
Parameters: $l_b = 0, u_b = 1, t = 0.5, \beta = 1.5, P = 0.5, F_{ADS} = 0.2$
Level1: Entropy Based Feature Selection
$\mathbb{E} \leftarrow entropy(\mathbb{F})$
$[\mathbb{I}, \mathbb{E}_2] \leftarrow sort(\mathbb{E})$
$\mathbb{I}_2 \leftarrow select(\mathbb{I}, e_c)$
$\mathbb{F}_2 \leftarrow \mathbb{F}(:, \mathbb{I}_2)$
Level2: MPA
$D \leftarrow \text{size}(\mathbb{F}_2, 2)$
Generate Initial Population
for $i = 1 : N$ **do**
  for $j = 1 : D$ **do**
    $\mathbb{P}(i, j) \leftarrow l_b + rand()(u_b - l_b)$
  end for
end for
while $I < I_{Max}$ **do**
  $fit_g \leftarrow \infty$
  Compute the fitness and Elite Solution
  for $I = 1 : N$ **do**
    $\mathbb{a} \leftarrow (\mathbb{P}(i, :) > t)$
    $Fit(i) \leftarrow CostFunction(\mathbb{F}_2, \mathbb{L}, \mathbb{a})$
    if $Fit(i) < fit_g$ **then**
      $fit_g \leftarrow Fit(i)$
      $\mathbb{x}_{gb} \leftarrow \mathbb{P}(i, :)$
    end if
  end for
  Generate Elite Matrix
  $\mathbb{E} \leftarrow Repeat(\mathbb{x}_{gb}, N, 1)$
  Phase 1
  if $I < \frac{I_{Max}}{3}$ **then**
    for $i = 1 : N$ **do**
      $\mathbb{R}_B \leftarrow randn(1, D)$
      for $d = 1 : D$ **do**
        $step \leftarrow \mathbb{R}_B(d).(\mathbb{E}(i, d) - \mathbb{R}_B(d).\mathbb{P}(i, d))$
        $\mathbb{P}(i, d) = \mathbb{P}(i, d) + P.rand().step$
      end for
    end for
    Phase2
  else if $\frac{I_{Max}}{3} < I < \frac{2}{3} I_{Max}$ **then**
    $C_F = (1 - \frac{I}{I_{Max}})^{\frac{2I}{I_{Max}}}$
    for $i = 1 : N$ **do**
      if $i \leq \frac{N}{2}$ **then**
        $\mathbb{R}_L \leftarrow Levy(\beta, D)$
        $\mathbb{R}_B \leftarrow randn(1, D)$
        for $d = 1 : D$ **do**
          $step \leftarrow \mathbb{R}_L(d).(\mathbb{E}_i(i, d) - \mathbb{R}_L(d).\mathbb{P}(i, d))$
          $\mathbb{P}(i, d) = \mathbb{P}(i, d) + P.rand().step$
        end for
      else
        $\mathbb{R}_B \leftarrow randn(1, D)$
        for $d = 1 : D$ **do**
          $step \leftarrow \mathbb{R}_B(d).(\mathbb{R}_B(d).\mathbb{E}_i(i, d) - \mathbb{P}(i, d))$
          $\mathbb{P}(i, d) \leftarrow \mathbb{E}(i, d) + P.C_F.step$
        end for
      end if
    end for
    Phase3
  else if $I > \frac{2}{3} I_{Max}$ **then**
    for $i = 1 : N$ **do**
      $\mathbb{R}_L \leftarrow Levy(\beta, D)$
      for $d = 1 : D$ **do**
        $step \leftarrow \mathbb{R}_L(d).(\mathbb{R}_L(d).\mathbb{E}_i(i, d) - \mathbb{P}(i, d))$
        $\mathbb{P}(i, d) \leftarrow \mathbb{E}(i, d) + P.C_F.step$
      end for
    end for
  end if
  FADs Effect
  $r \leftarrow randn()$
  if $r <= F_{ADS}$ **then**
    for $i = 1 : N$ **do**
      $: U \leftarrow (randn(1, D) < F_{ADS})$
      for $d = 1 : D$ **do**
        $\mathbb{P}(i, d) \leftarrow \mathbb{P}(i, d) + C_F(l_b + rand().(u_b - l_b).U(d)$
      end for
    end for
  else
    for $i = 1 : N$ **do**
      for $d = 1 : D$ **do**
        $\mathbb{P}_1 = \mathbb{P}(randn(), :), \mathbb{P}_2 = \mathbb{P}(randn(), :),$
        $\mathbb{P}(i, d) \leftarrow \mathbb{P}(i, d) + [F_{ADS}(1 - r) + r](\mathbb{P}_1 - \mathbb{P}_2)$
      end for
    end for
  end if
  Memory Saving
  for $i = 1 : N$ **do**
    $\mathbb{a} \leftarrow (\mathbb{P}(i, :) > t)$
    $Fit(i) \leftarrow CostFunction(\mathbb{F}_2, \mathbb{L}, \mathbb{a})$
    if $Fit(i) < fit_g$ **then**
      $fit_g \leftarrow Fit(i)$
      $\mathbb{x}_{gb} \leftarrow \mathbb{P}(i, :)$
    end if
  end for
end while
Compute the Index of Best Features
$\mathbb{I} \leftarrow 1 : D$
$\mathbb{S}_F \leftarrow \mathbb{I}((\mathbb{X}_{gb} > t) == 1)$
OUTPUT: $\mathbb{S}_F$

Function: *CostFunction*
Inputs:$\mathbb{F}_2, \mathbb{L}, \mathbb{a}$
Parameters: $\alpha_1 = 0.99, \alpha_2 = 0.01, k = 5, h_o = 0.2$
if (sum($\mathbb{a} == 1$) $== 0$ **then**
  $cost = 1$
else
  $\mathbb{F}_2 \leftarrow \mathbb{F}_2(:, \mathbb{a} == 1)$
  $\mathbb{F}_{train}, \mathbb{L}_{train}, \mathbb{F}_{test}, \mathbb{L}_{test} \leftarrow partition(\mathbb{F}_2, \mathbb{L}, h_o)$
  $\mathbb{a}_2 \leftarrow (\mathbb{a} == 1)$
  $Model \leftarrow trainKNN(\mathbb{F}_{train}, \mathbb{L}_{train}, k)$
  $\mathbb{L}_{pred} \leftarrow predict(Model, \mathbb{F}_{test})$
  $acc \leftarrow sum(\mathbb{L}_{pred} == \mathbb{L}_{test})/length(\mathbb{L}_{test})$
  $err \leftarrow 1 - acc$
  $f_s \leftarrow sum(\mathbb{a} == 1)$
  $f_t \leftarrow length(\mathbb{a})$
  $cost \leftarrow \alpha_1 \times err + \alpha_2 \times (\frac{f_s}{f_t})$
end if
OUTPUT: *cost*

2.2.6. Classification

The indices of the reduced feature set extracted from the ECMPA step discussed above are used to perform training and validation using baseline classifiers with multiple feature

settings. In this work, we used KNN [38] and SVM [39] classifiers with multiple kernel settings.

3. Results and Discussion

The proposed system was implemented in Matlab 2021 on Windows 10 64-bit using a Core i5, 2.5 GHz CPU, and 8 GB of RAM. The dataset of 5000 thousand images was split into training and testing with an 80% splitting ratio in order to perform transfer learning using DenseNet201 and DarkNet53 deep models. Table 3 demonstrates the main model training parameters for deep transfer learning. For both networks, a significant level of training and validation accuracy was achieved with five epochs. Figure 4 shows the accuracy and loss function plots for DenseNet201. In order to extract the indices of the most dominant features, the fused feature vector of size 2944 features was then subjected to feature reduction using ECMPA. The reduced set of features is then used to train the KNN and SVM classifiers with multiple kernel settings. In order to perform the classification task, testing images are applied to the trained deep models and a fused feature vector is obtained. The reduced feature vector is generated by using the indices obtained by the ECMPA. This is then classified using the trained KNN and SVM classifiers. Figure 5 demonstrates a set of reduced features extracted from the ECMPA step. Figure 6 demonstrates the results of the proposed WBCs classification system with various kernels of SVM and KNN classifiers. The SVM classifier achieves a 99.9% accuracy with a reduced feature set consisting of 70 strong features. The confusion matrix of SVM with the quadratic kernel is also demonstrated. The high value of true positive rate (TPR) and low value of false negative rate (FNR) are achieved for all image classes.

Table 3. Model training parameters for transfer learning of DenseNet201 and DarkNet53 models.

Property	Value	Property	Value
Kernel	sdgm	Initial Learning Rate	1×10^{-4}
Execution Environment	Auto	MiniBatch Size	20
MaxEpochs	5	Validation Frequency	30
Dropout rate	0.1	Stride Size	1

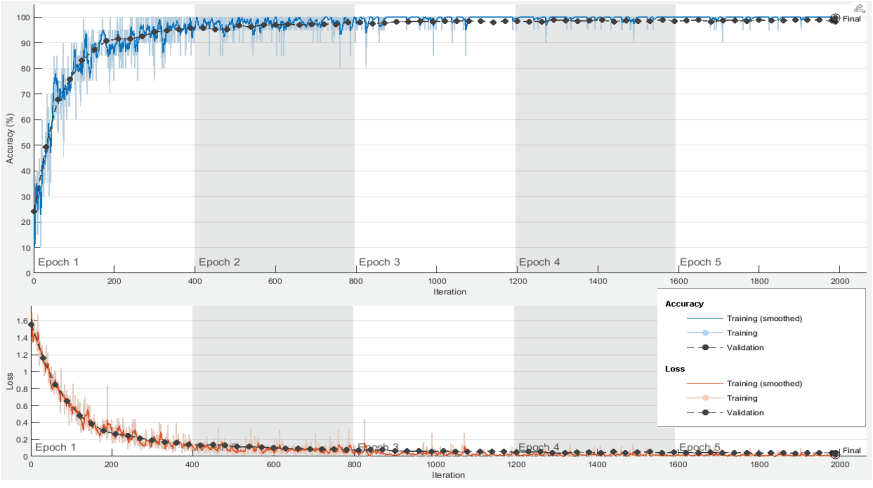


Figure 4. Training accuracy and loss function plots for DenseNet201 network.

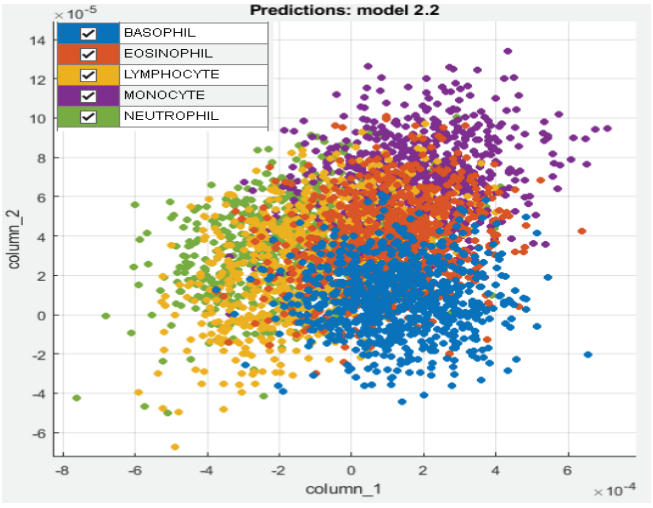


Figure 5. Extracted deep features from proposed ECMPA.

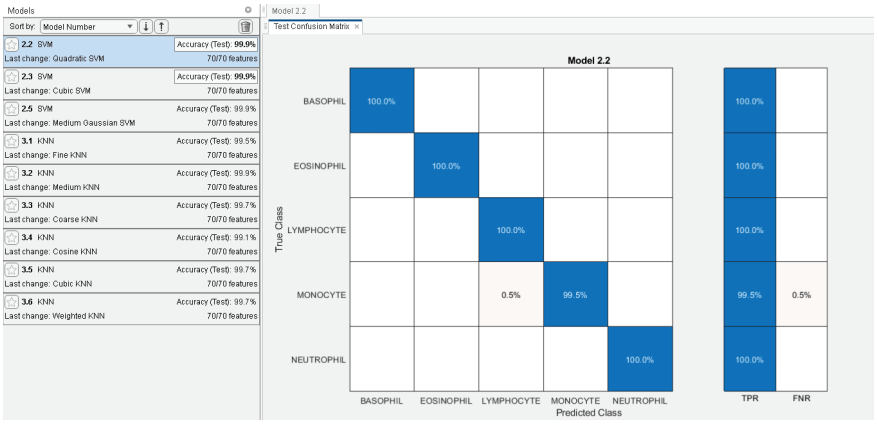


Figure 6. Classification results of proposed WBCs classification system. Left: Test accuracy achieved by SVM and KNN classifiers with several kernels. Right: Confusion matrix of SVM with quadratic kernel.

In Figure 7, the convergence of the proposed ECMPA is compared with a classical population-based meta-heuristic algorithm, i.e., genetic algorithm (GA). The graph demonstrates that ECMPA achieves a better value of cost function with a smaller number of iterations.

Table 4 shows an accuracy comparison of the proposed approach with some recent works on WBC classification using deep learning networks that use similar datasets. The proposed method shows a comparable or even better accuracy performance with a smaller number of features as compared to other works. This demonstrates the validity of the proposed approach.

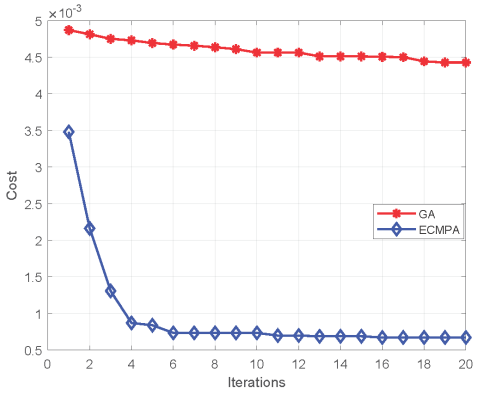


Figure 7. Convergence performance of ECMPA and GA.

Table 4. Performance Comparison of proposed method with some existing works. ×: Not done, N.A: Information not available.

Work	Deep Learning Model	Feature Selection	Feature Vector Size	Classifier	Accuracy %
[40]	GoogleNet, ResNet-50	Maximal Information Coefficient, Ridge Regression Model	755	Quadratic Discriminant Analysis	97.95
[41]	AlexNet	×	1000	CNN	98.4
[42]	PatternNet fused ensemble of CNNs	×	N.A	CNN	99.90
[43]	ResNet and Inception	Hierarchical Approach	N.A	ResNet and Inception	99.84
This Work	**DenseNet201 and DartkNet53**	**ECMPA**	**76**	**SVM and KNN**	**99.6**

Statistical Significance

Obtaining a certain level of confidence in the proposed strategy is the main goal of this statistical investigation. We use the analysis of variance (ANOVA) [44] to compare the means of several distributions in order to determine whether the results are statistically significant. We consider classification accuracy as a performance characteristic for our proposed framework. In order to implement ANOVA, we performed a series of tests to validate the assumption of normality using the Shapiro–Wilk test [45], and homogeneity of variance using the Bartlett's test [44]. In these testing procedures, we used 1% level of significance (i.e., $\alpha = 0.01$). The means of classification accuracy values for the selected classifiers, i.e., SVM, KNN, and NNN, as μ_1, μ_2, and μ_3, respectively. For each of the tests mentioned above, the null hypotheses are considered true if the computed Shapiro–Wilk p-values are less than or equal to α; otherwise, the alternative hypotheses are affirmed as true. The p-values of SVM, KNN, and NNN classifiers are obtained as $p_1 = 0.784$, $p_2 = 0.7124$, and $p_3 = 0.7031$, respectively. The Chi-squared probability from the Bartlett's test is $p_{ch} = 0.85$. From these p-values, we fail to renounce the null hypotheses and confidently claim that our accuracy data are normally distributed with homogeneous variances.

Table 5 presents the statistical results obtained from the ANOVA test including the sum of squared deviation (SS), degree of freedom (df), F-statistics, mean squared error

(MSE), and p-value. The obtained p-value is 0.705, which is greater than α and leads to the conclusion that the means of three classifiers are identical.

Table 5. Statistical test results based on ANOVA using accuracy metric.

V-Source	SS	df	MSE	F-Statistics	p-Value
Between	7.0812×10^{-5}	2	3.6333×10^{-5}	0.37	0.705
Within	5.9123×10^{-4}	6	9.8222×10^{-5}	-	-
Total	6.6232×10^{-4}	8	-	-	-

Figure 8 shows the confidence interval plots of accuracy values of the three selected classifiers. In the figure, red bars present the average accuracy, whereas the black bars present the 99% confidence limits of each classifier. In addition, the blue bars show lower and upper quantile points obtained by performing the above-mentioned statistical tests. From the figure, we can observe that the SVM classifier achieves a higher average accuracy with relatively smaller confidence interval size as compared to other classifiers. The quantile points of each classifier lie within their respective confidence limits. The higher p-values resulting from these quantile points lead to the acceptance of null hypotheses, which means significant differences in the accuracy distribution of the classifiers.

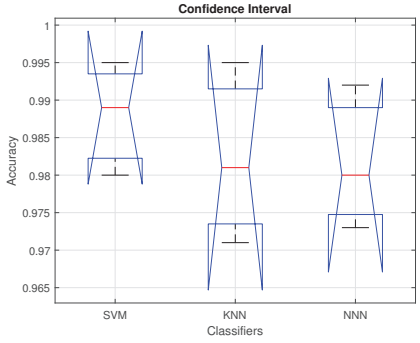


Figure 8. Confidence interval for selected classifiers.

4. Conclusions

WBCs classification is a vital step in the correct diagnosis of Leukemia. The existing manual methods of WBC classification are labor-intensive and error-prone. Automated WBC classification using computer vision techniques is an emerging paradigm. Modern approaches using deep neural networks achieve a significant level of accuracy for a variety of tasks. However, these neural networks suffer from exorbitant computational complexity, processing power, and memory requirement owing to very large feature sets. Therefore, an efficient feature reduction is essential to make deep neural networks feasible for real-time biomedical applications. This work proposes a complete WBCs classification pipeline that performs transfer learning using deep neural networks followed by an efficient feature reduction algorithm. The proposed feature reduction method is validated using several baseline classifiers with multiple kernel settings. An accuracy of 99.9% is achieved with a feature reduction of 95%, which demonstrates the feasibility of the proposed WBCs classification method. While the proposed approach has been applied to an augmented clean dataset containing only WBC subtype images, the ECMPA feature selection algorithm can be applied in any blood cell classification setup with little tuning of parameters. In the future, we plan to extend this work to a more challenging dataset for clinical-grade classification of other cell entities such as platelets and red blood cells, among others. The proposed algorithm can also be tested on the bench mark datasets for other diseases such as skin lesion and brain tumors, among others. In order to address the "*curse of dimensionality*",

other similar bio-inspired meta-heuristics can be investigated to obtain a trade-off between classification accuracy and computational complexity.

Author Contributions: Conceptualization, R.A. and M.A.; methodology, R.A. and M.A.; software, R.A., M.A. and T.A.; validation, M.A., R.A. and T.A.; resources, N.K.; writing—original draft preparation, R.A. and M.A.; writing—review and editing, M.A., T.A. and R.A. visualization, N.K.; supervision, N.K.; project administration, N.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research recieved no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data for this work shall be available upon request.

Conflicts of Interest: The authors declare no conflict of interest for this work.

Abbreviations

The following abbreviations are used in this manuscript:

WBC	White blood cell
CNN	Convolutional neural network
DNN	Deep neural network
SVMs	Support vector machines
MPA	Marine predators algorithm
ECMPA	Entropy-controlled marine predators algorithm
KNN	K-nearest neighbors
TPR	True positive rate
FNR	False negative rate
GAN	Generative adversarial network
GVF	Gradient vector flow

References

1. Kuan, D.H.; Wu, C.C.; Su, W.Y.; Huang, N.T. A microfluidic device for simultaneous extraction of plasma, red blood cells, and on-chip white blood cell trapping. *Sci. Rep.* **2018**, *8*, 15345. [CrossRef]
2. Farag, M.R.; Alagawany, M. Erythrocytes as a biological model for screening of xenobiotics toxicity. *Chem. Biol. Interact.* **2018**, *279*, 73–83. [CrossRef] [PubMed]
3. Rezatofighi, S.H.; Soltanian-Zadeh, H. Automatic recognition of five types of white blood cells in peripheral blood. *Comput. Med. Imaging Graph.* **2011**, *35*, 333–343. [CrossRef]
4. Weatherspoon, D. What to Know about White Blood Cells. Available online: https://www.medicalnewstoday.com/articles/327446#types-and-function (accessed on 22 November 2022).
5. Mathur, A.; Tripathi, A.S.; Kuse, M. Scalable system for classification of white blood cells from Leishman stained blood stain images. *J. Pathol. Inform.* **2013**, *4*, 15. [CrossRef]
6. Khan, M.A.; Sharif, M.; Akram, T.; Damaševičius, R.; Maskeliūnas, R. Skin Lesion Segmentation and Multiclass Classification Using Deep Learning Features and Improved Moth Flame Optimization. *Diagnostics* **2021**, *11*, 811. [CrossRef] [PubMed]
7. Bengio, Y.; Courville, A.; Vincent, P. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1798–1828. [CrossRef]
8. Donahue, J.; Jia, Y.; Vinyals, O.; Hoffman, J.; Zhang, N.; Tzeng, E.; Darrell, T. Decaf: A deep convolutional activation feature for generic visual recognition. In Proceedings of the International Conference on Machine Learning, PMLR, Bejing, China, 21–26 June 2014; pp. 647–655.
9. Nguyen, L.D.; Lin, D.; Lin, Z.; Cao, J. Deep CNNs for microscopic image classification by exploiting transfer learning and feature concatenation. In Proceedings of the 2018 IEEE International Symposium on Circuits and Systems (ISCAS), Florence, Italy, 27–30 May 2018; pp. 1–5.
10. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
11. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
12. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
13. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.

14. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
15. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
16. Shin, H.C.; Roth, H.R.; Gao, M.; Lu, L.; Xu, Z.; Nogues, I.; Yao, J.; Mollura, D.; Summers, R.M. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Trans. Med. Imaging* **2016**, *35*, 1285–1298. [CrossRef]
17. Sanei, S.; Lee, T.K. T: Cell recognition based on pca and bayesian classification. In Proceedings of the 4th International Symposium, ICA 2003, Nara, Japan, 1–4 April 2003.
18. Sarrafzadeh, O.; Rabbani, H.; Talebi, A.; Banaem, H.U. Selection of the best features for leukocytes classification in blood smear microscopic images. In Proceedings of the Medical Imaging 2014: Digital Pathology, San Diego, CA, USA, 15–20 February 2014; Volume 9041, pp. 159–166.
19. Ko, B.; Gim, J.; Nam, J. Cell image classification based on ensemble features and random forest. *Electron. Lett.* **2011**, *47*, 638–639. [CrossRef]
20. Abdullah, E.; Turan, M.K. Classifying white blood cells using machine learning algorithms. *Int. J. Eng. Res. Dev.* **2019**, *11*, 141–152.
21. Alruwaili, M. An intelligent medical imaging approach for various blood structure classifications. *Complexity* **2021**, *2021*, 5573300. [CrossRef]
22. Nithyaa, A.N.; Kumar, R.P.; Gokul, M.; Aananthi, C.G. Matlab Based Potent Algorithm for Wbc Cancer Detection and Classification. *Biomed. Pharmacol. J.* **2021**, *14*, 2277–2284. [CrossRef]
23. Sharma, S.; Gupta, S.; Gupta, D.; Juneja, S.; Gupta, P.; Dhiman, G.; Kautish, S. Deep learning model for the automatic classification of white blood cells. *Comput. Intell. Neurosci.* **2022**, *2022*, 7384131. [CrossRef]
24. Baby, D.; Devaraj, S.J.; Anishin Raj, M.M. Leukocyte classification based on transfer learning of VGG16 features by K-nearest neighbor classifier. In Proceedings of the IEEE 2021 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 13–14 May 2021; pp. 252–256.
25. Almezhghwi, K.; Serte, S. Improved classification of white blood cells with the generative adversarial network and deep convolutional neural network. *Comput. Intell. Neurosci.* **2020**, *2020*, 6490479. [CrossRef]
26. Yildirim, M.; Çinar, A. Classification of White Blood Cells by Deep Learning Methods for Diagnosing Disease. *Rev. d'Intell. Artif.* **2019**, *33*, 335–340. [CrossRef]
27. Alam, M.M.; Islam, M.T. Machine learning approach of automatic identification and counting of blood cells. *Healthc. Technol. Lett.* **2019**, *6*, 103–108. [CrossRef]
28. Wang, Q.; Bi, S.; Sun, M.; Wang, Y.; Wang, D.; Yang, S. Deep learning approach to peripheral leukocyte recognition. *PLoS ONE* **2019**, *14*, e0218808. [CrossRef]
29. Sahlol, A.T.; Abdeldaim, A.M.; Hassanien, A.E. Automatic acute lymphoblastic leukemia classification model using social spider optimization algorithm. *Soft Comput.* **2019**, *23*, 6345–6360. [CrossRef]
30. Sahlol, A.T.; Kollmannsberger, P.; Ewees, A.A. Efficient classification of white blood cell leukemia with improved swarm optimization of deep features. *Sci. Rep.* **2020**, *10*, 2536. [CrossRef]
31. Jung, C.; Abuhamad, M.; Mohaisen, D.; Han, K.; Nyang, D. WBC image classification and generative models based on convolutional neural network. *BMC Med. Imaging* **2022**, *22*, 94 . [CrossRef] [PubMed]
32. The Catholic University of Korea Institutional Review Board. 2019. Available online: https://bit.ly/2YrlQPl (accessed on 22 November 2022).
33. ImageNet. Available online: http://www.image-net.org (accessed on 22 November 2022).
34. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on Machine Learning, PMLR, Lille, France, 6–11 July 2015; pp. 448–456.
35. Awais, M.; Ahmed, A.; Naeem, M.; Iqbal, M.; Qadri, N.; Anpalagan, A. Multiple line outages identification: A customized quantum inspired approach. *Electr. Power Syst. Res.* **2016**, *134*, 47–55. [CrossRef]
36. Ahmed, A.; Khan, Q.; Naeem, M.; Iqbal, M.; Anpalagan, A.; Awais, M. An insight to the performance of estimation of distribution algorithm for multiple line outage identification. *Swarm Evol. Comput.* **2018**, *39*, 114–122. [CrossRef]
37. Faramarzi, A.; Heidarinejad, M.; Mirjalili, S.; Gandomi, A.H. Marine Predators Algorithm: A nature-inspired metaheuristic. *Expert Syst. Appl.* **2020**, *152*, 113377. [CrossRef]
38. Kafeel, A.; Aziz, S.; Awais, M.; Khan, M.A.; Afaq, K.; Idris, S.A.; Alshazly, H.; Mostafa, S.M. An Expert System for Rotating Machine Fault Detection Using Vibration Signal Analysis. *Sensors* **2021**, *21*, 7587. [CrossRef] [PubMed]
39. Khan, S.A.; Nazir, M.; Khan, M.A.; Saba, T.; Javed, K.; Rehman, A.; Akram, T.; Awais, M. Lungs nodule detection framework from computed tomography images using support vector machine. *Microsc. Res. Tech.* **2019**, *82*, 1256–1266. [CrossRef]
40. Toğaçar, M.; Ergen, B.; Cömert, Z. Classification of white blood cells using deep features obtained from Convolutional Neural Network models based on the combination of feature selection methods. *Appl. Soft Comput.* **2020**, *97*, 106810. [CrossRef]
41. Hegde, R.B.; Prasad, K.; Hebbar, H.; Singh, B.M.K. Feature extraction using traditional image processing and convolutional neural network methods to classify white blood cells: A study. *Australas. Phys. Eng. Sci. Med.* **2019**, *42*, 627–638. [CrossRef]

42. Wang, J.L.; Li, A.Y.; Huang, M.; Ibrahim, A.K.; Zhuang, H.; Ali, A.M. Classification of white blood cells with patternnet-fused ensemble of convolutional neural networks (pecnn). In Proceedings of the 2018 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), Louisville, KY, USA, 6–8 December 2018; pp. 325–330.
43. Habibzadeh, M.; Jannesari, M.; Rezaei, Z.; Baharvand, H.; Totonchi, M. Automatic white blood cell classification using pre-trained deep learning models: Resnet and inception. In Proceedings of the Tenth International Conference on Machine Vision (ICMV 2017), Vienna, Austria, 13–15 November 2017; Volume 10696, pp. 274–281.
44. Akram, T.; Laurent, B.; Naqvi, S.R.; Alex, M.M.; Muhammad, N. A deep heterogeneous feature fusion approach for automatic land-use classification. *Inf. Sci.* **2018**, *467*, 199–218. [CrossRef]
45. Akram, T.; Naqvi, S.R.; Haider, S.A.; Kamran, M.; Qamar, A. A novel framework for approximation of magneto-resistance curves of a superconducting film using GMDH-type neural networks. *Superlattices Microstruct.* **2020**, *145*, 106635. [CrossRef]

diagnostics

MDPI

Review

Accuracy Analysis of Deep Learning Methods in Breast Cancer Classification: A Structured Review

Marina Yusoff [1,2,*], Toto Haryanto [3], Heru Suhartanto [4], Wan Azani Mustafa [5,*], Jasni Mohamad Zain [1,2] and Kusmardi Kusmardi [6,7]

[1] Institute for Big Data Analytics and Artificial Intelligence (IBDAAI), Kompleks Al-Khawarizmi, Universiti Teknologi MARA (UiTM), Shah Alam 40450, Selangor, Malaysia
[2] College of Computing, Informatic and Media, Kompleks Al-Khawarizmi, Universiti Teknologi MARA (UiTM), Shah Alam 40450, Selangor, Malaysia
[3] Department of Computer Science, IPB University, Bogor 16680, Indonesia
[4] Faculty of Computer Science, Universitas Indonesia, Depok 16424, Indonesia
[5] Faculty of Electrical Engineering Technology, Universiti Malaysia Perlis, UniCITI Alam Campus, Sungai Chuchuh, Padang Besar 02100, Perlis, Malaysia
[6] Department of Anatomical Pathology, Faculty of Medicine, Universitas Indonesia/Cipto Mangunkusumo Hospital, Jakarta 10430, Indonesia
[7] Human Cancer Research Cluster, Indonesia Medical Education and Research Institute, Universitas Indonesia, Jakarta 10430, Indonesia
* Correspondence: marina998@uitm.edu.my (M.Y.); wanazani@unimap.edu.my (W.A.M.)

Abstract: Breast cancer is diagnosed using histopathological imaging. This task is extremely time-consuming due to high image complexity and volume. However, it is important to facilitate the early detection of breast cancer for medical intervention. Deep learning (DL) has become popular in medical imaging solutions and has demonstrated various levels of performance in diagnosing cancerous images. Nonetheless, achieving high precision while minimizing overfitting remains a significant challenge for classification solutions. The handling of imbalanced data and incorrect labeling is a further concern. Additional methods, such as pre-processing, ensemble, and normalization techniques, have been established to enhance image characteristics. These methods could influence classification solutions and be used to overcome overfitting and data balancing issues. Hence, developing a more sophisticated DL variant could improve classification accuracy while reducing overfitting. Technological advancements in DL have fueled automated breast cancer diagnosis growth in recent years. This paper reviewed studies on the capability of DL to classify histopathological breast cancer images, as the objective of this study was to systematically review and analyze current research on the classification of histopathological images. Additionally, literature from the Scopus and Web of Science (WOS) indexes was reviewed. This study assessed recent approaches for histopathological breast cancer image classification in DL applications for papers published up until November 2022. The findings of this study suggest that DL methods, especially convolution neural networks and their hybrids, are the most cutting-edge approaches currently in use. To find a new technique, it is necessary first to survey the landscape of existing DL approaches and their hybrid methods to conduct comparisons and case studies.

Keywords: breast cancer; classification; histopathological image; review

Citation: Yusoff, M.; Haryanto, T.; Suhartanto, H.; Mustafa, W.A.; Zain, J.M.; Kusmardi, K. Accuracy Analysis of Deep Learning Methods in Breast Cancer Classification: A Structured Review. *Diagnostics* **2023**, *13*, 683. https://doi.org/10.3390/diagnostics13040683

Academic Editor: Sameer Antani

Received: 3 January 2023
Revised: 6 February 2023
Accepted: 7 February 2023
Published: 11 February 2023

1. Introduction

Cancer, which occurs when cells in the body grow unnaturally, is one of the main causes of human death. It has become a huge problem that threatens the safety and well-being of people all over the world. Breast cancer is one of the most widely known types of disease affecting women. These days, breast cancer is the deadliest cancer that can strike women, making it the leading cause of death overall. Due to its high prevalence and broad dissemination, breast cancer is dangerous and mostly depends on pathological

diagnosis [1]. Accurate breast cancer diagnoses can improve cancer survival rates. Imaging modalities are being developed to promptly detect this disease. Cancer diagnosis relies on medical images, which are used for screening, diagnostic, and adjunctive evaluation to give field experts more confidence in their initial decision. Mammography, histopathology, MRI, ultrasonography, thermography, and PET/CT are imaging modalities. Each modality has its own advantages and disadvantages [2]. For instance, a mammogram is suitable for early detection. However, it is expensive and has low specificity.

Histopathological images serve as a valuable guide for the pathological analysis of this disease, as they can be used to describe tumor characteristics in all subgroups of interval cancers. A precise prognosis is dependent on the accurate classification of these images. It has been shown that the histopathological tumor characteristics of missed interval cancers are worse than those of true interval cancers in terms of prognosis [3]. For this reason, the number of interval cancers in a screening program can be reduced by improving a radiologist's perception and interpretation by establishing a systematic collection of features and implementing structured reviews. Much research on breast cancer has been conducted to identify benign and malignant categories, with histopathological imaging often used in diagnosis. However, the high complexity of histopathological images and their associated heavy workload makes this work extremely time-consuming [3–5]. Additionally, diagnosis is tough, routine, and repetitive, as the clinical diagnosis of breast cancer is a complex problem. Additionally, the subjectivity of each pathologist may influence results, further complicating matters.

Histopathological breast cancer image recognition is a difficult task [6,7]. Traditional manual feature extraction and image classification usually necessitate in-depth knowledge, and more professional researchers are needed to extract and calculate high-quality image features. This operation typically takes a long time and produces poor classification results. It is critical in this field to develop an accurate, automated method for analyzing histopathological images because breast cancer diagnosis heavily relies on histopathological image classification [8]. Furthermore, previous research has shown that histopathological images have important clinical implications in the early stage of breast cancer screening [9,10]. The accurate classification of histopathological images could be used to develop automated and precise breast cancer classification systems for early detection [9]. Therefore, the proper screening of histopathological images is critical for the earlier detection and diagnosis of breast cancer.

Much work on histopathological image classification has been reported. Decades ago, traditional machine learning (ML) approaches, such as artificial neural networks (ANNs), support vector machines (SVMs), and decision trees, began to be used in breast cancer image classification. Deep learning (DL) has been introduced to overcome issues of accuracy in ML models. DL has snowballed and been successfully applied to many applications [11]. DL allows for massive amounts of data learning [12–14]. More importantly, DL has improved sound and image recognition [15–17]. DL comprises several methods, such as convolution neural networks (CNNs), recurrent neural networks (RNNs), long short-term memory (LSTM), gated recurrent unit (GRU), and bidirectional LSTM [15–17]. The CNN is a popular DL method that directly learns from input without feature extraction. The CNN is famous for being used to improve regularized MLP networks. CNN layers optimize parameters for meaningful output and reduce model complexity. Visual geometry group (VGG), AlexNet, Xception, Inception, ResNet, and other CNN variants can be used in different application domains depending on their learning capabilities [18,19]. Another popular DL method is the RNN. The CNN and RNN architectures differ. RNN works with feedback results [15,20]. Time series data mostly require an RNN where feedback loops provide memory for optimal performance in context-sensitive data [21], such as sentence completion.

As a result, this study examined recent technological advances in histopathological breast cancer image classification using DL methods. Consequently, this research considered DL methods to evaluate the performance of current classification solutions. Researchers have proposed DL techniques to determine the best method for classifying

histopathological breast cancer images, and these techniques were reviewed in this paper. The findings of this study are expected to assist researcher in choosing a suitable DL method. Correct diagnoses are required for complete care in a limited amount of time in cases of breast cancer, as the classification of benign and malignant cancers can save lives. DL performance depends on an image's type, size, and features. Many previous studies embedded augmentation operations or strategies into DL methods to improve classification accuracy for histopathological images, with several methods employed to enhance image features.

While DL methods have shown promise, there is a need for a recent review of their efficacy in analyzing histopathological breast cancer images solutions. This work investigated DL and integration with other feature extraction, normalization, and optimization methods in histopathological image classification. This study employed a research ideology review to determine the best way to classify breast cancer based on its histopathology. The review considers what is known now and what methodological contributions have been made to classification solutions for specific problems. The rest of the paper is organized as follows. Section 2 describes the review methods. Section 3 shows the results and findings. Section 4 concludes the discussion.

2. Review Method

The use of contemporary technology in medical data analysis has evolved along with technology, particularly in image processing, classification, and segmentation, as well as in cancer research. The most well-liked machine learning method for medical image diagnosis, DL, is being used by an increasing number of researchers. The medical community agrees that the future of DL in disease prediction is promising [22]. Additionally, scientists have applied various classification techniques to cancer image data to categorize breast cancer. Convolutional neural networks (CNNs) are widely used in image classification [23,24]. In addition, many researchers have tried to apply pre-processing, feature extraction, ensemble learning, and classifier techniques to automate the detection of breast cancer cells.

High-performance evaluation values in terms of accuracy, specificity, sensitivity, precision, and F-measure can be obtained using several methods, and DL methods can be used to compensate for traditional methods of breast cancer diagnosis, allowing for the early detection of the disease. This paper provides a comprehensive review based on advanced searching related to the classification of histopathological breast cancer images using DL and its hybrids. Advanced evaluation is one of the most critical discussions at the moment. Thus, a systematic flow method was used in this work. A protocol or plan with clearly stated criteria before a review is referred to as a structured review [25], which is a method for strategically identifying patterns, trends, and critical evaluations of the literature on research subjects [26].

Following the analysis and integration of the results, further readings in the literature were utilized to develop future research directions on the applications of DL to histopathological classification. The review technique comprised four steps for choosing numerous relevant papers for this study. This study used the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) methodology [27]; it is a framework created to illustrate the flow of information during the various stages of a systematic review, as shown in Figure 1. The first step in writing this comprehensive literature review was identifying research items that may be relevant to the research question. The total number of searched papers was then screened. Finally, the eligibility of each paper based on its abstract was evaluated. Overall, the scientific literature was reviewed and summarized to identify, select, and evaluate breast cancer classification techniques. Subsequently, additional research directions to address the raised concerns were recommended. In this study, the best practice method was used to conduct the comprehensive literature review, and the publication rules provided essential information to help the researchers evaluate the accuracy of the review. Furthermore, an investigation for the systematic analysis of the various studies

considered within this review. The WOS and Scopus databases were used to examine the studied methodologies.

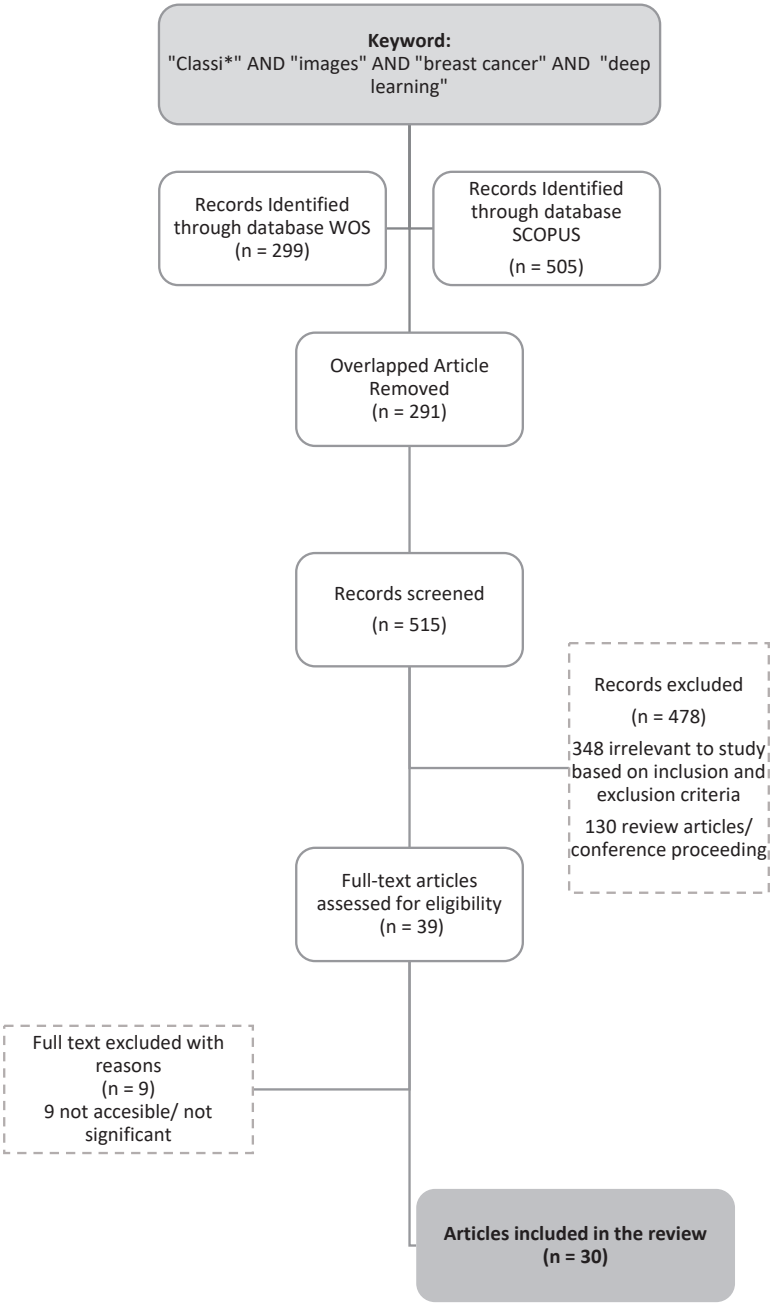


Figure 1. The PRISMA flow diagram of the entire procedure for selecting reviewed articles.

2.1. Preliminary Identification

Articles in the literature were identified to select studies on the utilization of DL in histopathological classification. Keywords such as "classi*" AND "image*" AND "breast cancer" AND "deep learning" were applied. The year was restricted to 2022 to consider all related recent studies. The literature search was conducted by using the Scopus and WOS databases. The initial search yielded 505 articles from Scopus and 299 articles from the WOS, as demonstrated in Table 1.

Table 1. Search strings from the Scopus and WOS databases.

Search strings from the Scopus database	TITLE-ABS-KEY "classi*" AND "images" AND "breast cancer" AND "deep learning" (LIMIT-TO PUBYEAR, 2022)	Results = 503 Articles
Search strings from the WOS (Web of Science) database	TS = "classi*" AND "images" AND "breast cancer" AND "deep learning" AB = "classi*" AND "images" AND "breast cancer" AND "deep learning" Refined by: DOCUMENT TYPES: (ARTICLE) AND PUBLICATION YEAR: 2022	Results = 299 Articles

2.2. Screening

Screening is used to examine relevant research items for content that matches predefined research question(s). Here, machine learning-classified cervical cancer cells were employed to select research items in the screening phase. This step removed duplicate papers from the searched list. The first screening eliminated 291 publications, and the second step comprised the examination of 515 papers based on this study's exclusion and inclusion criteria (see Table 2). The selection criteria for the search were based on those of Alias et al. [28]. Research papers were the first criterion because they provide practical advice, and they include reviews, meta-synthesis, meta-analyses, books, book series, chapters, and conference proceedings not included in the most recent studies. Publications written in English released in 2022 were analyzed and considered for coverage. Note that 478 articles were removed because of their premature results and because they did not discuss the DL method in histopathological classification. Some articles were also incomplete, and some of the full articles were not accessible, had broken links, and overlapped.

Table 2. Second-stage examination criteria.

Criterion	Inclusion	Exclusion
Language	English	Non-English
Published Year	2022	<2022
Source type	Journal (only research articles)	Conference proceeding
Document Type	Article	Letter, review, conference, and note
Research Area	Computer Science and Engineering	Besides Computer Science and Engineering

2.3. Eligibility

After all inclusion and exclusion criteria were met, the final review sample was generated. Based on the research objectives, the inclusion criteria included articles that helped us find recent solutions for challenges in histopathological breast cancer imaging, DL methodology, and types of classification analysis. Consequently, 9 publications were excluded since their titles and abstracts were not significantly related to the study's purpose based on empirical data. Finally, 30 papers were made available for evaluation (see Figure 1).

A further inclusion criterion was that the studies had to be in the field of computer science and engineering. This helped narrow this review to DL theories and methods in histopathological breast cancer classification. Furthermore, tools and predictive factors aided the extraction, comparison, and synthesis processes. The exclusion criteria excluded articles that focused on different contexts (such as brain cancer, traditional machine learning, and healthcare) and studies that were not specifically about histopathological breast cancer images. Figure 2 shows examples of four magnification classes of histopathological images from the BreakHis dataset. Then, the authors of this study generated themes based on evidence from this analysis. Analyses, points of view, questions, and other data interpretation ideas were recorded in a log. Finally, the authors looked for theme design inconsistencies in the results; they discussed any conceptual differences, and themes were tweaked to ensure consistency. To validate the issues and ensure subtheme clarity, importance, and appropriateness during the expert review, Dr. Kusmardi from Cipto Mangunkusumo Hospital, Jakarta, and the Faculty of Medicine, Universitas Indonesia, Indonesia, was chosen as a pathology expert.

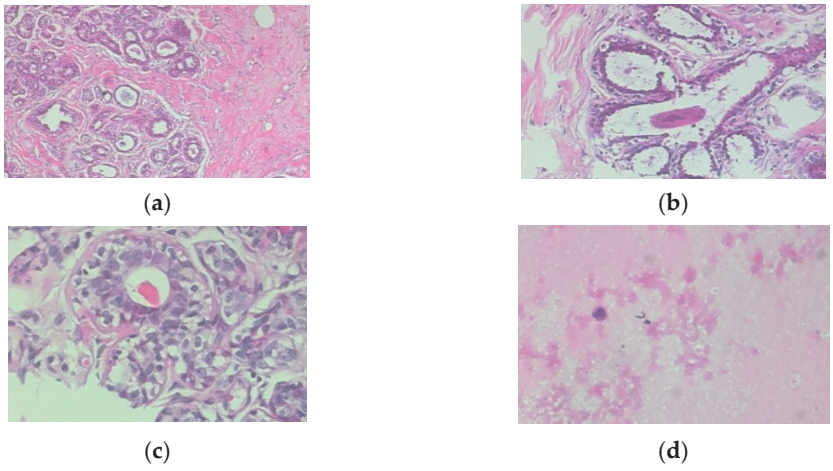


Figure 2. (**a**) Histopathological breast cancer images of the BreakHis dataset: (**a**) 40×; (**b**) 100×; (**c**) 200×; (**d**) 400×.

3. Results and Findings

As the prevalence of breast cancer has grown, image-based histopathological analysis has become widely used in pathological research and disease diagnosis. However, pathologist errors in cell identification have been identified as a significant problem. As a result, a comparative evaluation of the proposed model was performed to demonstrate the utility of feature selection and class imbalance. In addition, how well the classifier performed in terms of accuracy, sensitivity, precision, F-measure, and specificity was assessed. This study evaluated the use of a cell classification algorithm associated with imaging in breast cancer screening to improve the effectiveness and accuracy of the early clinical diagnosis of breast cancer, and it was found that researchers have developed several methods to fix the problems of the proposed classification methods. One popular technique for categorizing cancer cells is the use of CNNs. Note that 30 articles were ultimately selected for inclusion and analysis based on extensive searching.

Classification of Histopathological Images on Deep Learning Approach

Many researchers have contributed to the field of histopathological image classification, and a summary of methods and outcomes facilitated comparisons between studies. Table 3 summarizes recent DL research on categorizing histopathological breast cancer images.

Table 3. Summary of recent deep learning research.

Reference	Dataset	Class	Methodology	Results	Advantages/Shortcomings
[5]	BreakHis	Multi-class	Data augmentation and handcrafted feature extraction (FE) techniques (Hu moment, Haralick textures, color histograms, and deep neural networks (DNNs)). Hybrid DL method: CNN + feature extraction.	Accuracy: 40×: 97.87%; 100×: 97.60%; 200×: 96.10%; 400×: 96.84%. Average accuracy: 97.10%	The proposed handcrafted feature extraction and DNN showed the best performance.
[29]			Dual residual block with a multi-scale dual residual recurrent network (MTRRE-Net74). DL Method: RNN.	Accuracy: 40×: 97.12%; 100×: 95.2%; 200×: 96.8%; 400×: 97.81%.	Better accuracy at four magnification levels than previously described models.
[30]			NetDense residual dual-shuffle attention network (DRDA-Net) inspired by the bottleneck unit of the ShuffleNet architecture. DL Method: CNN.	Accuracy: 40×: 95.72%; 100×: 94.41%; 200×: 97.43%; 400×: 98.1%/	The proposed model showed acceptable accuracy. Densely connected blocks addressed the overfitting and vanishing gradient problems.
[3]			DenseNet201 and VGG16 architecture models as an ensemble model used to extract global features. DEEP Pachi extracts spatial information on the region of interest. Hybrid DL method: CNN + pre-processing + augmentation + ensemble.	BreakHis Malignant Accuracy: 99%. Benign Accuracy: 100%.	A significant result.
[31]	Wide-scale data		DeepML framework to achieve multi-class classification. DL Method: CNN.	Average accuracy: 98% (90–10% train–test split) and 89% (80–20% train–test).	Acceptable accuracy.
[32]	BreakHis	Binary	DL are DenseNet_201, MobileNet_V2, and Inception_V3. Ensemble or boosting methods are AdaBoost (ADB), Gradient Boosting Machine (GBM), LightGBM (LGBM), and XGBoost (XGB) with Decision Tree (DT) as a base learner. Hybrid DL method: CNN + ensemble.	XGB + Inception_V3 showed the best accuracy of 92.52%.	Feature extraction and boosting ensembles were shown to be a good combination for image classification.
[10]			Pre-processed by multiple scaling decompositions to prevent overfitting due to the large dimension. It improved the DenseNet-201-MSD network model. DL Method: CNN + preprocessing.	Accuracy: 40×: 99.4%; 100×: 98.8%; 200×: 98.2; 400×: 99.4%.	The mode demonstrated very good results and can be used with other image data.

Table 3. *Cont.*

Reference	Dataset	Class	Methodology	Results	Advantages/Shortcomings
[33]	BreakHis	Binary	LightXception is based on cutting off layers at the bottom of the Xception network, which reduces the number of convolution filter channels. LightXception only has about 35% of the parameters of the original. Hybrid DL method: CNN + filter.	At 100× magnification: Xception Accuracy: 97.31%; Xception Recall: 98.67; Xception Precision: 99.26%; LightXception Accuracy: 97.42%; LightXception Recall: 97.42%; LightXception Precision: 97.42%.	Acceptable classification solution; the model needs improvement.
[34]			Support vector machine (SVM), CNN, and CNN with transfer learning (TL). Hybrid DL method: CNN + TL.	Accuracy: SVM: 92%; CNN: 94%; CNN + TL: 97%.	Improved results for CNN + transfer learning compared with a single method.
[35]			DenseNet201 with the SVM RBF classifier. Hybrid DL method: CNN + classifier.	At 200× magnification: Accuracy of 95.39% and precision of 95.43%.	Acceptable accuracy could have been due to the use of traditional machine learning as a classifier.
[23]			The attention model features multi-scale channel recalibration and msSE-ResNet convolutional neural network (msSE-ResNet34). Hybrid DL method: CNN + FE.	Accuracy: 88.87%.	Accuracy was low, although the approach was embedded with FE.
[36]			CNN and color channel with attention module (CWA-Net). Hybrid DL method: CNN + FE.	At 400× magnification: Private dataset accuracy: 95%; BreakHis dataset accuracy: 96%	Acceptable accuracy.
[37]			DenseNet as a backbone model and transfer learning (DenTnet). Hybrid DL method: CNN + TL.	Accuracy: 99.28%.	Good generalization ability and computational speed.
[38]			"Deep-Hist" with pre-trained and Stochastic Gradient Descent (SGD). Type: CNN + optimization.	Accuracy: 92.46%.	The proposed model required accuracy improvements.
[39]			Pre-trained Xception model with VGG16 and enhanced with logistic regression. Uses real-time data augmentation (AUG). Hybrid DL method: CNN + TL + AUG.	Xception + VGG16: Precision of 78.67% and recall of 0.76. F-score of 0.75, and AUC of 0.86. Xception + VGG16 + logistic regression: precision of 82.45%, recall of 0.82. F-score of 0.82, and AUC of 0.90.	The proposed model outperformed a conventional CNN. Augmentation could reduce the problem of overfitting.

Table 3. *Cont.*

Reference	Dataset	Class	Methodology	Results	Advantages/Shortcomings
[22]	Breaches'	Binary	Xception and deeplabv3+. DL method: CNN.	Binary accuracy: 95%. Malignant accuracy: 99%.	The proposed framework demonstrated remarkable performance on a malignant dataset.
[40]			Inception-ResNet-v2 and Categorical Boosting (CatBoost), Extreme Gradient Boosting (XGBoost), and Light Gradient Boosting Machine (LightGBM). DL method: CNN.	Accuracy: 40×: 96.82%; 100×: 95.84%; 200×: 97.01%; 400×: 96.15%; Average: 96.46%.	The proposed Inception-ResNet-v2 and Categorical Boosting (CatBoost) models outperformed other methods.
[41]			Two custom deep architectures, CSAResnet and DAMCNN, integrated with channel and spatial attention. Type: CNN + ensemble.	Accuracy: 99.55%.	The proposed model demonstrated outstanding accuracy, showing that the employed ensemble method could be successfully used on the studied type of images.
[42]	Invasive breast carcinoma		CNN with Resnet50 and Xception. Type: CNN.	Xception was better than Restnet50, with an accuracy of 88% and a sensitivity of 95%.	Model performance was still low for Xception. The model could be improved by embedding ensemble and classifier programs.
[43]	BACH, UC, and BreakHis		Deconv-Transformer (DecT) with color jitter data augmentation. Hybrid DL method: CNN + AUG.	BreakHis dataset accuracy: 93.02%. BACH dataset accuracy: 79.06%. UC dataset accuracy: 81.36%.	Performance was low even though augmentation was added. A hyperparameter tuning and pre-processing method would improve performance.
[44]			Embeds attention mechanism and high-order statistical representation into a residual convolutional network (attention high-order deep network (AHoNet). Adds non-dimensionality reduction and local cross-channel interaction to achieve local salient deep features with normalization (NORM). Hybrid DL method: CNN + FE + NORM.	BreakHis dataset accuracy: 99.29%. BACH dataset accuracy: 85%.	Performance was more competitive than previously described models. A very good model to be tested with other types of image data.
[45]	BreakHis and FNAC	Binary	Twenty-eight hybrid architectures combining seven recent deep learning techniques for feature extraction (DenseNet 201, Inception V3, Inception ResNet V2, MobileNet V2, ResNet 50, VGG16, and VGG19) and four classifiers (MLP, SVM, DT, and KNN) for binary classification. Hybrid DL method: CNN + classifier.	DenseNet 201 (MDEN) with MLP showed the best performance. FNAC accuracy: 99.29%. BreakHis accuracy: 40×: 92.61%; 100×: 92%; 200×: 93.93%; 400×: 91.73%.	The proposed model requires accuracy improvements.

Table 3. *Cont.*

Reference	Dataset	Class	Methodology	Results	Advantages/Shortcomings
[46]	Public image database	Binary and multi-class	Ensemble (ENS) for color adjustment methods with VGG-19 architectures. Hybrid DL method: CNN + ENS.	Accuracy: 91.93%. AUC: 97.72%.	The proposed model with color adjustment was shown to be an acceptable classification solution.
[47]	PCam Kaggle		Hybrid deep learning (CNN-GRU). Hybrid DL method: CNN and GRU.	Accuracy: 86.21%. Precision: 85.50%. Sensitivity: 85.60%. Specificity: 84.71%. F1-score: 88%. AUC: 0.89.	The proposed model showed low accuracy. GRU seemingly could not improve the method's performance.
[48]	Histopathological data and ultrasound data		Automatic framework for reliable breast cancer classification CNN and TL. Embeds Manta Ray Foraging Optimization (MRFO) as metaheuristic optimization (OPT) to improve the framework's adaptability. Hybrid DL method: CNN + TL + OPT.	Histopathological data accuracy: 97.73%. Ultrasound data accuracy: 99.01%.	The proposed framework was superior to previously tested methods. MRFO has the potential to be applied to other types of images.
[49]	BreakHis invasive ductal carcinoma (IDC) dataset		CNN with logistic regression (LR), random forest (RF), k-nearest neighbor (K-NN), support vector machine (SVM), linear SVM, Gaussian Naïve Bayesian (GNB), and decision tree (DT) processes. Hybrid DL method: CNN + classifier.	Invasive ductal carcinoma: Range accuracy: 80–86%; range precision: 92–94%; range recall: 91–96%; range F1-score: 94–96%. BreakHis: range accuracy: 91–94%; range precision: 91–95%; range recall: 93–96%; F1-score: 95–98%.	Improvements in accuracy, precision, recall, and F1 score were shown. Hyperparameter tuning, pre-processing, and image augmentation may be used to achieve better classification performance.
[1]	BreakHis		The novel model fusion framework utilizes online mutual knowledge transfer (MF-OMKT) to classify histopathological breast cancer images. Imitates mutual communication and learning. Hybrid DL method: CNN + MF.	Accuracy range: Binary [99.27%, 99.84%]; Multi-class [96.14%, 97.53%].	The proposed framework demonstrated good accuracy. The authors suggested evaluating a fusion strategy with other cancerous image data.
[50]	BreakHis histopathological dataset—ICIAR 2018		DenseNet architecture and DenseNet architecture with image level mean. Hybrid DL method: CNN + NORM.	BreakHis: Binary DenseNet accuracy: 96.55%. Multi-class DenseNet accuracy: 91.82%. Binary DenseNet with image level mean accuracy: 91.72%. Multi-class with image level mean DenseNet accuracy: 96.72%. Competitive accuracies of 93.25% and 92.3% at the patient and image levels, respectively.	The proposed framework outperformed previously assessed methods. The normalization method seemed to influence improvements in results.

Table 3. *Cont.*

Reference	Dataset	Class	Methodology	Results	Advantages/Shortcomings
[51]	Cancer images	Not available	Wavelet transform (WT) process is applied to noisy images; VggNet-16 model Hybrid DL method: CNN + preprocessing.	Best accuracy for the dataset with Gaussian noise of a 0.3 intensity: 86.9%.	Accuracy was low. WT was not found to be suitable for image data, though it may be applicable to other types of data.

One plausible explanation for these findings is that the histopathological diagnosis of breast cancer classification has been researched several times. CNNs currently comprise one of the best approaches for the classification process. Previous researchers have studied several techniques related to CNNs, such as CNNs with ensembles, CNNs with feature extraction, CNNs with model fusion, and CNNs with classifiers. For example, Li et al. [1] introduced a novel model fusion framework based on knowledge transfer (MF-OMKT) for binary histopathological breast cancer image classification. It demonstrated better performance than that reported in other studies, with an accuracy of 99.84%. Their findings showed an improvement of 0.55% compared with a CNN with feature extraction and normalization. On the other hand, Karthik et al. [41] calculated the same accuracy of 99.55% with two ensemble learning strategies, namely, channel and spatial attention, integrated with custom deep architectures. The ensemble model reduced the margin of error and improved the overall classification performance. This kind of approach is usually called ensemble learning and involves fusing multiple learning algorithms. Ensemble learning could produce a robust and reliable model with good generalization performance. However, the results were still low in some cases, as seen in Figure 3.

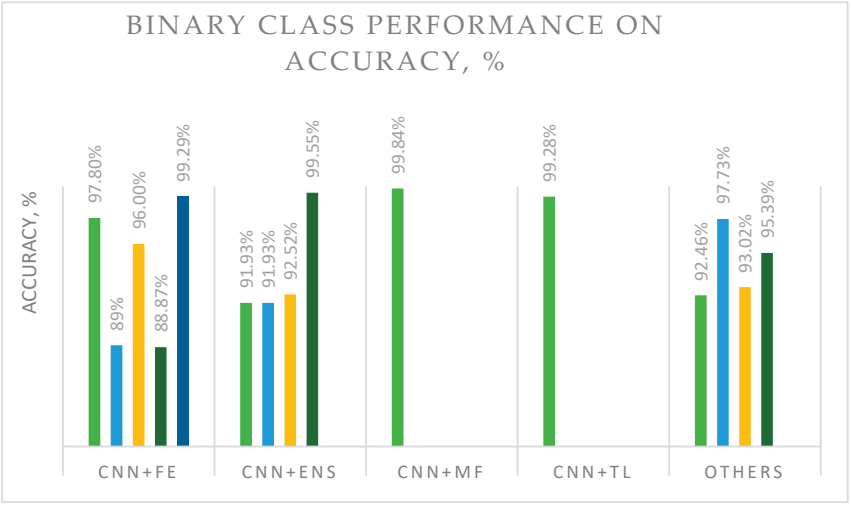


Figure 3. Binary classification accuracy of the popular and best classification approaches.

Another improved model used CNN with pre-processing, showing accuracies of 0.44% for 40× and 400× magnifications, respectively, of the BreakHis dataset. These results were nearly identical to those of a previous study by Li et al. [1] who utilized CNN and filtering; LightXception achieved an accuracy of 97.42%, a recall of 97.42%, and a precision of 97.42% in that study. Another study by Yang and Guan [9,10] classified pathological medical images of breast cancer using the BreakHis image dataset and an improved network DenseNet201-MSD model; the new DL model classified pathological images of the BreakHis dataset with

accuracies of 99.4%, 98.8%, 98.2%, and 99.4% at four magnifications. These results were consistent with those of [48], which used CNN with transfer learning and integrated Manta Ray Foraging Optimization (MRFO) as a metaheuristic optimization method to increase the adaptability of image features. As a nature-inspired algorithm, Manta Ray Foraging improved classification performance, but it requires more testing on several parameters. Additionally, Burçak and Uuz [52] concluded that CNN models are robust feature selection strategies in four categories of histopathological images. Their findings contradicted the findings of Rashmi et al. [36], who proposed a classification method based on a CNN and a color channel with an attention module (CWA-Net).

Amin et al. [22] proposed a hybrid semantic model that employed pre-trained Xception and deeplabv3+ models to classify microscopic cancer images into malignant and benign classes, with 95% and 99% accuracy for benign and malignant, respectively. Compared with previously published methodologies, the proposed framework demonstrated exceptional performance. Numerous scholarly articles have studied the classification and analysis of breast cancer using histopathological images. In addition, histopathological images of breast cancer patients are increasingly classified using deep CNNs. Most of the research solutions dealt with histopathological breast cancer modalities. The BreakHis dataset has mainly been popular in the binary classification of malignant and benign cancers. Different methodological approaches have demonstrated a variety of accuracy levels for the studied datasets. Much more research is expected to consider sub-classes of datasets, such as regarding a diversity of magnification to improve the performance of models, as seen in Table 3.

The authors of this study established two categories of classification approaches, namely, binary and multi-class DL solutions. Figures 3 and 4 illustrate these two approaches, with a focus on the best performance and popular hybrid solutions. Hybrid DL models were found to demonstrate better accuracy in pathological image classification. For binary classification, five different methods were used in recent studies: a CNN with feature extraction (CNN + FE), a CNN with ensemble (CNN + ENS), a CNN with model fusion (CNN + MF), a CNN with transfer learning (CNN + TL), and others. Binary classification is based on popular BreakHis image data. It can be seen from Figure 3 that the CNN + FE and CNN + ENS models have been the most popular approaches in the binary classification of breast cancer based on histopathological images. Additionally, the classification method based on CNN with model fusion demonstrated a higher accuracy level (more than 99.84%) than other studied approaches, as illustrated in Figure 3. In this case, fusion adaptation showed significant influence on feature extraction. Other approaches showed accuracies ranging from 89% to 99.55%. For instance, the recent DL with ensemble approach demonstrated an accuracy of 91% to 92%. Therefore, more research is required to improve model classification accuracy, although the CNN + MF model has been shown to be the best of the currently described methods. In addition, statistical analysis methods such as the t-test must be performed to confirm the significance of the studied methods.

For multi-class classification, studies have used a CNN with pre-processing, augmentation, and ensemble (CNN + PRE + AUG + ENS); a CNN with feature extraction (CNN + FE); a CNN with model fusion (CNN + MF); a CNN with normalization (CNN-Norm); and a CNN alone. Figure 3 demonstrates the performance of DL variants for multi-class histopathological image classification. The CNN + PRE + AUG + ENS model outperformed other methods, with an accuracy of 100%.

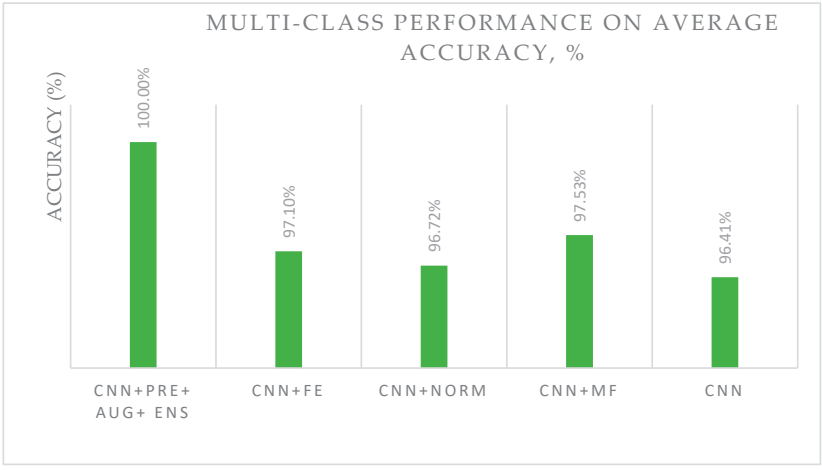


Figure 4. Multi-class classification accuracy of the popular and best classification approaches.

4. Discussion

DL technological advancements are propelling the growth of automated breast cancer diagnosis. A breast cancer diagnosis can be performed with various image modalities, such as histopathological images. One of the most significant challenges in DL has long-been the accurate and automatic classification of pathological medical images. Additionally, the use of deeper layers in neural networks enables higher abstraction levels and more precise data analysis. Therefore, neural networks have become increasingly popular in evaluate the performance of classification approaches in recent years. Researchers have proposed many CNN variants to determine the best method for classifying histopathological images, as discussed in this paper. The findings of this study are intended to help identify the best performing CNN methods.

Several established methods can be used to detect and classify benign and malignant cancers based on deep feature characteristics. Ensemble learning and embedded fusion models have shown better performance than other integration methods. Furthermore, a CNN with model fusion is a powerful tool for precise feature extraction and histopathological image classification. The suggested idea of adapting an online mutual knowledge transfer strategy as a fusion strategy embedded in CNNs could be promising for other types of breast cancer detection.

Different levels of accuracy were demonstrated by several hybrid CNN methods. Some showed an excellent accuracy of more than 97%, and others showed an accuracy of below 97%. The combination of a CNN with FE and ENS has shown different levels of accuracy for the same dataset depending on the data variance, feature selection, and methodological approach used in binary and multi-class classification. It is evident that the fusion strategy has a high viability in binary classification solutions but a low viability for multi-class classification solutions. Another promising strategy with strong performance is the combination of the pre-processing, augmentation, ensemble, and CNN methods.

The progression of DL has resulted In the production of promising solutions for the binary and multi-class classification of breast cancer images, with a primary focus on histopathology. It is hoped that more health information on areas such as the brain, eyes, chest, heart, abdomen, musculoskeletal system, and other human body regions will be incorporated into DL models. The findings presented in Figures 3 and 4 and Table 3 can serve as a foundation for developing DL models. Incorporating pre-processing, feature extraction, and augmentation methods into models is one way to improve their performance. In addition, the use of a fusion strategy is likely to produce favorable outcomes for binary classification and could be improved to suit multi-class image data.

5. Conclusions

A correct diagnosis is necessary for the comprehensive treatment of breast cancer in a short time. Accordingly, lives can be saved with the accurate classification of benign and malignant cancers. DL performance depends on the input images' type, size, and characteristics. Many previous studies embedded augmentation operations or strategies into DL methods, especially CNNs, to improve classification accuracy for histopathological images. Similarly, several methods have been proposed to enhance image features. However, a review of DL models and their performance for histopathological breast cancer images is lacking for both binary and multi-class classification solutions. Therefore, this work investigated DL and its integration with other feature extraction and normalization methods in histopathological image classification. This review paper is intended to aid the creation of better breast cancer classification designs and methodologies to assist in the identification process of this cancer. Furthermore, the proposed CNN hybrid architecture simplifies the detection and classification of cancer cells in histopathological images, potentially leading to the earlier detection of breast cancer and an increase in women's survival rates. More research should be conducted on methods, beginning with studies of pre-processing, feature extraction, and classification using various breast cancer images. Furthermore, a new strategy for improving classification performance in histopathological images should be imposed on hybridization with computational optimization algorithms such as cuckoo search, the firefly algorithm, and particle swarm optimization to find local and global image features that lead to better classification performance.

Author Contributions: Conceptualization, M.Y.; methodology, T.H. and M.Y.; formal analysis, T.H.; investigation, H.S. and W.A.M.; resources, M.Y.; writing—original draft preparation, M.Y. and H.S.; writing—review and editing, H.S. and W.A.M.; visualization, J.M.Z. and K.K.; supervision, M.Y.; project administration, H.S. and W.A.M.; funding acquisition, M.Y. and J.M.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Universiti Teknologi MARA and Universitas Indonesia under the Strategic Partnership Research Grant: 100-RMC 5/3/SRP (053/2021).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to acknowledge Universiti Teknologi MARA, Universitas Indonesia, Research Management Center, and Institute for Big Data Analytics and Artificial Intelligence (IBDAAI) for the financial support provided to this research project.

Conflicts of Interest: The authors declare they have no conflict of interest to report regarding the present study.

References

1. Li, G.; Li, C.; Wu, G.; Xu, G.; Zhou, Y.; Zhang, H. MF-OMKT: Model Fusion Based on Online Mutual Knowledge Transfer for Breast Cancer Histopathological Image Classification. *Artif. Intell. Med.* **2022**, *134*, 102433. [CrossRef] [PubMed]
2. Dar, R.A.; Rasool, M.; Assad, A. Breast Cancer Detection Using Deep Learning: Datasets, Methods, and Challenges Ahead. *Comput. Biol. Med.* **2022**, *149*, 106073. [CrossRef]
3. Ukwuoma, C.C.; Hossain, M.A.; Jackson, J.K.; Nneji, G.U.; Monday, H.N.; Qin, Z. Multi-Classification of Breast Cancer Lesions in Histopathological Images Using DEEP_Pachi: Multiple Self-Attention Head. *Diagnostics* **2022**, *12*, 1152. [CrossRef] [PubMed]
4. Sharma, S.; Mehra, R. Conventional Machine Learning and Deep Learning Approach for Multi-Classification of Breast Cancer Histopathology Images—A Comparative Insight. *J. Digit. Imaging* **2020**, *33*, 632–654. [CrossRef] [PubMed]
5. Ameh Joseph, A.; Abdullahi, M.; Junaidu, S.B.; Hassan Ibrahim, H.; Chiroma, H. Improved Multi-Classification of Breast Cancer Histopathological Images Using Handcrafted Features and Deep Neural Network (Dense Layer). *Intell. Syst. Appl.* **2022**, *14*, 200066. [CrossRef]
6. Sali, R.; Adewole, S.; Ehsan, L.; Denson, L.A.; Kelly, P.; Amadi, B.C.; Holtz, L.; Ali, S.A.; Moore, S.R.; Syed, S.; et al. Hierarchical Deep Convolutional Neural Networks for Multi-Category Diagnosis of Gastrointestinal Disorders on Histopathological Images. In Proceedings of the 2020 IEEE International Conference on Healthcare Informatics, ICHI, Oldenburg, Germany, 30 November–3 December 2020; IEEE: Piscatvie, NJ, USA, 2020.

7. Saturi, R.; Prem Chand, P. A Novel Variant-Optimized Search Algorithm for Nuclei Detection in Histopathology Breast Cancer Images. In *Lecture Notes in Networks and Systems*; Springer: Singapore, 2022; Volume 286.
8. Liu, M.; He, Y.; Wu, M.; Zeng, C. Breast Histopathological Image Classification Method Based on Autoencoder and Siamese Framework. *Information* **2022**, *13*, 107. [CrossRef]
9. Yang, J.; Ju, J.; Guo, L.; Ji, B.; Shi, S.; Yang, Z.; Gao, S.; Yuan, X.; Tian, G.; Liang, Y.; et al. Prediction of HER2-Positive Breast Cancer Recurrence and Metastasis Risk from Histopathological Images and Clinical Information via Multimodal Deep Learning. *Comput. Struct. Biotechnol. J.* **2022**, *20*, 333–342. [CrossRef]
10. Yang, Y.; Guan, C. Classification of Histopathological Images of Breast Cancer Using an Improved Convolutional Neural Network Model. *J. Xray Sci. Technol.* **2022**, *30*, 33–44. [CrossRef]
11. Mohamed, A.; Amer, E.; Eldin, N.; Hossam, M.; Elmasry, N.; Adnan, G.T. The Impact of Data Processing and Ensemble on Breast Cancer Detection Using Deep Learning. *J. Comput. Commun.* **2022**, *1*, 27–37. [CrossRef]
12. Munappy, A.R.; Bosch, J.; Olsson, H.H.; Arpteg, A.; Brinne, B. Data Management for Production Quality Deep Learning Models: Challenges and Solutions. *J. Syst. Softw.* **2022**, *191*, 111359. [CrossRef]
13. Franceschini, S.; Grelet, C.; Leblois, J.; Gengler, N.; Soyeurt, H. Can Unsupervised Learning Methods Applied to Milk Recording Big Data Provide New Insights into Dairy Cow Health? *J. Dairy Sci.* **2022**, *105*, 6760–6772. [CrossRef]
14. Yang, C.-T.; Kristiani, E.; Leong, Y.K.; Chang, J.-S. Big Data and Machine Learning Driven Bioprocessing—Recent Trends and Critical Analysis. *Bioresour. Technol.* **2023**, *372*, 128625. [CrossRef]
15. Nwonye, M.J.; Narasimhan, V.L.; Mbero, Z.A. Sensitivity Analysis of Coronary Heart Disease Using Two Deep Learning Algorithms CNN RNN. In *2021 IST-Africa Conference (IST-Africa)*; IEEE: Piscatvie, NJ, USA, 2021.
16. Chauhan, N.K.; Singh, K. A Review on Conventional Machine Learning vs. deep learning. In *2018 International Conference on Computing, Power and Communication Technologies (GUCON)*; IEEE: Piscatvie, NJ, USA, 2019; pp. 347–352.
17. Vankdothu, R.; Hameed, M.A. Brain Tumor MRI Images Identification and Classification Based on the Recurrent Convolutional Neural Network. *Meas. Sens.* **2022**, *24*, 100412. [CrossRef]
18. Bagherzadeh, S.; Maghooli, K.; Shalbaf, A.; Maghsoudi, A. Recognition of Emotional States Using Frequency Effective Connectivity Maps through Transfer Learning Approach from Electroencephalogram Signals. *Biomed. Signal Process Control* **2022**, *75*, 103544. [CrossRef]
19. Elangovan, P.; Nath, M.K. En-ConvNet: A Novel Approach for Glaucoma Detection from Color Fundus Images Using Ensemble of Deep Convolutional Neural Networks. *Int. J. Imaging Syst. Technol.* **2022**, *32*, 2034–2048. [CrossRef]
20. Cui, R.; Liu, M.; Alzheimer's Disease Neuroimaging Initiative. RNN-Based Longitudinal Analysis for Diagnosis of Alzheimer's Disease. *Comput. Med. Imaging Graph.* **2019**, *73*, 1–10. [CrossRef]
21. Freeborough, W.; van Zyl, T. Investigating Explainability Methods in Recurrent Neural Network Architectures for Financial Time Series Data. *Appl. Sci.* **2022**, *12*, 1427. [CrossRef]
22. Amin, J.; Sharif, M.; Fernandes, S.L.; Wang, S.-H.; Saba, T.; Khan, A.R. Breast Microscopic Cancer Segmentation and Classification Using Unique 4-Qubit-Quantum Model. *Microsc. Res. Tech.* **2022**, *85*, 1926–1936. [CrossRef]
23. Alqahtani, Y.; Mandawkar, U.; Sharma, A.; Hasan, M.N.S.; Kulkarni, M.H.; Sugumar, R. Breast Cancer Pathological Image Classification Based on the Multiscale CNN Squeeze Model. *Comput. Intell. Neurosci.* **2022**, *2022*, 7075408. [CrossRef]
24. Gour, M.; Jain, S.; Sunil Kumar, T. Residual Learning Based CNN for Breast Cancer Histopathological Image Classification. *Int. J. Imaging Syst. Technol.* **2020**, *30*, 621–635. [CrossRef]
25. Munn, Z.; Peters, M.D.J.; Stern, C.; Tufanaru, C.; McArthur, A.; Aromataris, E. Systematic Review or Scoping Review? Guidance for Authors When Choosing between a Systematic or Scoping Review Approach. *BMC Med. Res. Methodol.* **2018**, *18*, 143. [CrossRef] [PubMed]
26. Gough, D.; Oliver, S.; Thomas, J. (Eds.) *An Introduction to Systematic Reviews(2017)*; SAGE: New York, NY, USA, 2017.
27. Cho, S.H.; Shin, I.S. A Reporting Quality Assessment of Systematic Reviews and Meta-Analyses in Sports Physical Therapy: A Review of Reviews. *Healthcare* **2021**, *9*, 1368. [CrossRef] [PubMed]
28. Alias, N.A.; Mustafa, W.A.; Jamlos, M.A.; Alquran, H.; Hanafi, H.F.; Ismail, S.; Ab Rahman, K.S. Pap Smear Images Classification Using Machine Learning: A Literature Matrix. *Diagnostics* **2022**, *12*, 2900. [CrossRef] [PubMed]
29. Chattopadhyay, S.; Dey, A.; Singh, P.K.; Oliva, D.; Cuevas, E.; Sarkar, R. MTRRE-Net: A Deep Learning Model for Detection of Breast Cancer from Histopathological Images. *Comput. Biol. Med.* **2022**, *150*, 106155. [CrossRef] [PubMed]
30. Chattopadhyay, S.; Dey, A.; Singh, P.K.; Sarkar, R. DRDA-Net: Dense Residual Dual-Shuffle Attention Network for Breast Cancer Classification Using Histopathological Images. *Comput. Biol. Med.* **2022**, *145*, 105437. [CrossRef]
31. Sharma, M.; Mandloi, A.; Bhattacharya, M. A Novel DeepML Framework for Multi-Classification of Breast Cancer Based on Transfer Learning. *Int. J. Imaging Syst. Technol.* **2022**, *32*, 1963–1977. [CrossRef]
32. Nakach, F.-Z.; Zerouaoui, H.; Idri, A. Hybrid Deep Boosting Ensembles for Histopathological Breast Cancer Classification. *Health Technol.* **2022**, *12*, 1043–1060. [CrossRef]
33. Kim, S.; Yoon, K. Deep and Lightweight Neural Network for Histopathological Image Classification. *J. Mob. Multimed.* **2022**, *18*, 1913–1930. [CrossRef]
34. Alkhathlan, L.; Saudagar, A.K.J. Predicting and Classifying Breast Cancer Using Machine Learning. *J. Comput. Biol.* **2022**, *29*, 497–514. [CrossRef]

35. Xu, Y.; dos Santos, M.A.; Souza, L.F.F.; Marques, A.G.; Zhang, L.; da Costa Nascimento, J.J.; de Albuquerque, V.H.C.; Rebouças Filho, P.P. New Fully Automatic Approach for Tissue Identification in Histopathological Examinations Using Transfer Learning. *IET Image Process* **2022**, *16*, 2875–2889. [CrossRef]
36. Rashmi, R.; Prasad, K.; Udupa, C.B.K. Region-Based Feature Enhancement Using Channel-Wise Attention for Classification of Breast Histopathological Images. *Neural Comput. Appl.* **2022**, 1–16. [CrossRef]
37. Wakili, M.A.; Shehu, H.A.; Sharif, M.H.; Sharif, M.H.U.; Umar, A.; Kusetogullari, H.; Ince, I.F.; Uyaver, S. Classification of Breast Cancer Histopathological Images Using DenseNet and Transfer Learning. *Comput. Intell. Neurosci.* **2022**, *2022*, 8904768. [CrossRef]
38. Iqbal, S.; Qureshi, A.N. Deep-Hist: Breast Cancer Diagnosis through Histopathological Images Using Convolution Neural Network. *J. Intell. Fuzzy Syst.* **2022**, *43*, 1347–1364. [CrossRef]
39. Kumar, S.; Sharma, S. Sub-Classification of Invasive and Non-Invasive Cancer from Magnification Independent Histopathological Images Using Hybrid Neural Networks. *Evol. Intell.* **2022**, *15*, 1531–1543. [CrossRef]
40. Abbasniya, M.R.; Sheikholeslamzadeh, S.A.; Nasiri, H.; Emami, S. Classification of Breast Tumors Based on Histopathology Images Using Deep Features and Ensemble of Gradient Boosting Methods. *Comput. Electr. Eng.* **2022**, *103*, 108382. [CrossRef]
41. Karthik, R.; Menaka, R.; Siddharth, M. v Classification of Breast Cancer from Histopathology Images Using an Ensemble of Deep Multiscale Networks. *Biocybern. Biomed. Eng.* **2022**, *42*, 963–976. [CrossRef]
42. el Agouri, H.; Azizi, M.; el Attar, H.; el Khannoussi, M.; Ibrahimi, A.; Kabbaj, R.; Kadiri, H.; BekarSabein, S.; EchCharif, S.; Mounjid, C.; et al. Assessment of Deep Learning Algorithms to Predict Histopathological Diagnosis of Breast Cancer: First Moroccan Prospective Study on a Private Dataset. *BMC Res. Notes* **2022**, *15*, 66. [CrossRef]
43. He, Z.; Lin, M.; Xu, Z.; Yao, Z.; Chen, H.; Alhudhaif, A.; Alenezi, F. Deconv-Transformer (DecT): A Histopathological Image Classification Model for Breast Cancer Based on Color Deconvolution and Transformer Architecture. *Inf. Sci.* **2022**, *608*, 1093–1112. [CrossRef]
44. Zou, Y.; Zhang, J.; Huang, S.; Liu, B. Breast Cancer Histopathological Image Classification Using Attention High-Order Deep Network. *Int. J. Imaging Syst. Technol.* **2022**, *32*, 266–279. [CrossRef]
45. Zerouaoui, H.; Idri, A. Deep Hybrid Architectures for Binary Classification of Medical Breast Cancer Images. *Biomed. Signal Process Control* **2022**, *71*, 103226. [CrossRef]
46. Luz, D.S.; Lima, T.J.B.; Silva, R.R.V.; Magalhães, D.M.V.; Araujo, F.H.D. Automatic Detection Metastasis in Breast Histopathological Images Based on Ensemble Learning and Color Adjustment. *Biomed. Signal Process Control* **2022**, *75*, 103564. [CrossRef]
47. Wang, X.; Ahmad, I.; Javeed, D.; Zaidi, S.A.; Alotaibi, F.M.; Ghoneim, M.E.; Daradkeh, Y.I.; Asghar, J.; Eldin, E.T. Intelligent Hybrid Deep Learning Model for Breast Cancer Detection. *Electronics* **2022**, *11*, 2767. [CrossRef]
48. Baghdadi, N.A.; Malki, A.; Balaha, H.M.; AbdulAzeem, Y.; Badawy, M.; Elhosseini, M. Classification of Breast Cancer Using a Manta-Ray Foraging Optimized Transfer Learning Framework. *PeerJ Comput. Sci.* **2022**, *8*, e1054. [CrossRef] [PubMed]
49. Deepa, B.G.; Senthil, S. Predicting Invasive Ductal Carcinoma Tissues in Whole Slide Images of Breast Cancer by Using Convolutional Neural Network Model and Multiple Classifiers. *Multimed. Tools Appl.* **2022**, *81*, 8575–8596. [CrossRef]
50. Abdul Jawad, M.; Khursheed, F. Deep and Dense Convolutional Neural Network for Multi Category Classification of Magnification Specific and Magnification Independent Breast Cancer Histopathological Images. *Biomed. Signal Process Control* **2022**, *78*, 103935. [CrossRef]
51. Cengiz, E.; Kelek, M.M.; Oğuz, Y.; Yllmaz, C. Classification of Breast Cancer with Deep Learning from Noisy Images Using Wavelet Transform. *Biomed. Tech.* **2022**, *67*, 143–150. [CrossRef]
52. Burçak, K.C.; Uğuz, H. A New Hybrid Breast Cancer Diagnosis Model Using Deep Learning Model and ReliefF. *Traitement Du Signal* **2022**, *39*, 521–529. [CrossRef]

diagnostics

MDPI

Article

Improvement of Time Forecasting Models Using Machine Learning for Future Pandemic Applications Based on COVID-19 Data 2020–2022

Abdul Aziz K Abdul Hamid [1,2], Wan Imanul Aisyah Wan Mohamad Nawi [1], Muhamad Safiih Lola [1,3,*], Wan Azani Mustafa [4,5,*], Siti Madhihah Abdul Malik [1], Syerrina Zakaria [1], Elayaraja Aruchunan [6], Nurul Hila Zainuddin [7], R.U. Gobithaasan [1,3] and Mohd Tajuddin Abdullah [8,9]

1 Faculty of Ocean Engineering Technology and Informatics, Universiti Malaysia Terengganu, Kuala Nerus 21030, Terengganu, Malaysia
2 Special Interest Group on Applied Informatics and Intelligent Applications (AINIA), Universiti Malaysia Terengganu, Kuala Nerus 21030, Terengganu, Malaysia
3 Special Interest Group on Modeling and Data Analytics (SIGMDA), Universiti Malaysia Terengganu, Kuala Nerus 21030, Terengganu, Malaysia
4 Faculty of Electronic Engineering & Technology, Pauh Putra Campus, Universiti Malaysia Perlis (UniMAP), Arau 02600, Perlis, Malaysia
5 Centre of Excellence for Advanced Computing, Pauh Putra Campus, Universiti Malaysia Perlis (UniMAP), Arau 02600, Perlis, Malaysia
6 Faculty of Science, Institute of Mathematical Sciences, Universiti Malaya, Kuala Lumpur 50603, Malaysia
7 Mathematics Department, Faculty of Science and Mathematics, Universiti Pendidikan Sultan Idris, Tanjong Malim 53900, Perak Darul Ridzuan, Malaysia
8 Faculty of Fisheries and Food Science, Universiti Malaysia Terengganu, Kuala Nerus 21030, Terengganu, Malaysia
9 Fellow Academy of Sciences Malaysia, Level 20, West Wing Tingkat 20, Menara MATRADE, Jalan Sultan Haji Ahmad Shah, Kuala Lumpur 50480, Malaysia
* Correspondence: safiihmd@umt.edu.my (M.S.L.); wanazani@unimap.edu.my (W.A.M.)

Citation: K Abdul Hamid, A.A.; Wan Mohamad Nawi, W.I.A.; Lola, M.S.; Mustafa, W.A.; Abdul Malik, S.M.; Zakaria, S.; Aruchunan, E.; Zainuddin, N.H.; Gobithaasan, R.U.; Abdullah, M.T. Improvement of Time Forecasting Models Using Machine Learning for Future Pandemic Applications Based on COVID-19 Data 2020–2022. *Diagnostics* **2023**, *13*, 1121. https://doi.org/10.3390/diagnostics13061121

Academic Editor: Dechang Chen

Received: 22 January 2023
Revised: 17 February 2023
Accepted: 20 February 2023
Published: 15 March 2023

Abstract: Improving forecasts, particularly the accuracy, efficiency, and precision of time-series forecasts, is becoming critical for authorities to predict, monitor, and prevent the spread of the Coronavirus disease. However, the results obtained from the predictive models are imprecise and inefficient because the dataset contains linear and non-linear patterns, respectively. Linear models such as autoregressive integrated moving average cannot be used effectively to predict complex time series, so nonlinear approaches are better suited for such a purpose. Therefore, to achieve a more accurate and efficient predictive value of COVID-19 that is closer to the true value of COVID-19, a hybrid approach was implemented. Therefore, the objectives of this study are twofold. The first objective is to propose intelligence-based prediction methods to achieve better prediction results called autoregressive integrated moving average–least-squares support vector machine. The second objective is to investigate the performance of these proposed models by comparing them with the autoregressive integrated moving average, support vector machine, least-squares support vector machine, and autoregressive integrated moving average–support vector machine. Our investigation is based on three COVID-19 real datasets, i.e., daily new cases data, daily new death cases data, and daily new recovered cases data. Then, statistical measures such as mean square error, root mean square error, mean absolute error, and mean absolute percentage error were performed to verify that the proposed models are better than the autoregressive integrated moving average, support vector machine model, least-squares support vector machine, and autoregressive integrated moving average–support vector machine. Empirical results using three recent datasets of known the Coronavirus Disease-19 cases in Malaysia show that the proposed model generates the smallest mean square error, root mean square error, mean absolute error, and mean absolute percentage error values for training and testing datasets compared to the autoregressive integrated moving average, support vector machine, least-squares support vector machine, and autoregressive integrated moving average–support vector machine models. This means that the predicted value of the proposed model is closer to the true value. These results demonstrate that the proposed model can generate

estimates more accurately and efficiently. Compared to the autoregressive integrated moving average, support vector machine, least-squares support vector machine, and autoregressive integrated moving average–support vector machine models, our proposed models perform much better in terms of percent error reduction for both training and testing all datasets. Therefore, the proposed model is possibly the most efficient and effective way to improve prediction for future pandemic performance with a higher level of accuracy and efficiency.

Keywords: COVID-19 pandemic; machine learning; hybrid models; forecasting; public health; accuracy and efficiency

1. Introduction

The city of Wuhan in Hubei Province, China, made history as the first point of the spread of the coronavirus disease (COVID-19) due to severe respiratory syndrome. On January 31, the World Health Organization (WHO) declared for the first time that COVID-19 is a "public health emergency of international concern" [1]. The virus was originally thought to have come from a fish market in Wuhan. On 11 January 2020, the gene sequence that China openly shared through personal contacts fueled its rapid spread, with a total of 9,129,146 confirmed cases, including 473,797 deaths worldwide as of 24 June 2020 [2]. However, as of 1 May 2021, COVID-19 has infected more than 151 million people and caused three million deaths worldwide. Countries such as the USA, Brazil, Russia, Spain, UK, Italy, France, Germany, China, India, Iran, and Pakistan have been hit the hardest by COVID-19. The first cases of COVID-19 reported in Malaysia on 2 January 2020 were detected by Chinese tourists entering the country from Singapore [3]. Only single-digit daily cases were reported in the initial phase, but this increased to 235 by 26 March [4]. The number of daily cases in Malaysia continued to increase exponentially by reaching around 20,000 in August 2021. The Malaysian government declared the implementation the Movement Control Order (MCO) from 18 March to 3 May 2020, the Conditional MCO (CMCO) from 4 May to 9 June 2020, and the Recovery MCO (RMCO) from 10 June 2020 to 31 March 2021. All travel and socio-economic activities (religious and cultural gatherings were not allowed) have been restricted across the country to keep new infections at bay and avoid overloading the country's healthcare system during this time. All government and private offices and educational institutions, including transportation hubs, have been closed, citizens have been ordered to stay at home, and interstate travel has been banned, with fines of up to RM 10,000 for violators.

Since the WHO declared the COVID-19 outbreak a pandemic, not only governments from around the world, but also dedicated medical institutions have made many efforts to find vaccines and treatments to control the spread of the virus. Statisticians and public health scientists have also performed extensive statistical modelling, especially regarding the forecasting of COVID-19 cases, to help the health system prevent the contagion catastrophe. In this scenario, the ability to most effectively determine the growth rate at which the epidemic is spreading is very critical to counterattack and help governments, social planning, and policy making to accurately address the epidemic. Therefore, the motivation behind this research compared to the existing research is to (i) develop the most accurate and efficient predictive model related to the spread of COVID-19 in Malaysia and (ii) compare the performance of this new model with autoregressive integrated moving average (ARIMA), support vector machine (SVM), least-squares support vector machine (LSSVM), autoregressive integrated moving average-support vector machine ARIMA–SVM, and autoregressive integrated moving average–least-squares support vector machine models (ARIMA–LSSVM).

During the pandemic, many studies have been conducted using various mathematical and statistical models to predict the spread of the COVID-19 pandemic. One of the most popular time-series prognostic models for analyzing and predicting disease spread is the

ARIMA (p, d, q) model [5–7]. Predicting new daily cases of COVID-19 was a difficult task, as cases increased daily. In the first wave, the pattern of COVID-19 cases has been continuously increasing for a period and then decreasing. However, for the second wave, it appears to be picking up again, and some of the COVID-19 cases are difficult to predict. In this scenario, some researchers predict the pattern of COVID-19 using ARIMA [8–14]. However, the ARIMA model has a limitation in that it typically can only handle a linear time-series data structure [15]. ARIMA model approximations are insufficient to pose a time-series prediction obstacle for researchers, especially for nonlinear patterns [16]. Despite its superior performance, the classification performance of Support Vector Machines (SVMs) and the generalizability of the classifier are often affected by the dimension or number of feature variables used, as mentioned by Lee [17]. As a result of the development of Vector Machines models, this process will be able to provide the most accurate and efficient result in each prediction case. SVMs, first introduced in 1995 by Vladimir Vapnik [18] in the field of statistical learning theory and structural risk minimization, have proven useful in a variety of prediction problems and classifications. SVMs could also manage or address difficulties such as non-linearity, local minimum, and high dimension where the ARIMA model could not [15,19–21]. SVM models have recently been used to handle problems such as nonlinear, local minimum, and high dimension. SVM can even guarantee higher accuracy for long-term predictions compared to other computational approaches in many practical applications. However, the single SVM model as a single ARIMA model also has some limitations, as the SVM model can only handle non-linear data and not linear data. With the limitations of a single ARIMA and SVM model, as well as an in-depth analysis of time-series prediction, hybrid approaches have become the best approach to overcome both limitations, and have a very significant impact in many areas due to their dynamic nature and higher level of predicting accuracy, efficiency, and precision. This approach is crucial because of the problems encountered in time-series forecasting, where almost all real time series contain linear and nonlinear correlation patterns between the data. Recently, the hybridization of prediction methods has been used with great success to achieve higher prediction accuracy [15,16,19,20,22–26].

Regarding the spread of COVID-19, the hybrid time-series model approach is crucial for predicting the impact of the COVID-19 outbreak, and has proven successful in predicting COVID-19 [27–33]. This study aims to (a) propose the ARIMA–LSSVM hybrid model approach to achieve better forecast results when it is able to produce the best estimator, i.e., produce small error terms; additionally, it aims to (b) examine the performance of the proposed models by comparing them to ARIMA and SVM models using three daily cases of COVID-19 data in Malaysia, that is, daily new positive cases, daily new deaths, and daily new recovered cases. Despite recent advances in time series and on COVID-19, the modelling process does not include COVID-19 cases specifically in Malaysia to help authorities manage the spread of this outbreak by producing more efficient, more accurate data, and more accurate forecasting results.

This study makes a significant contribution to the field of pandemic prediction and prevention by introducing novel approaches to dealing with COVID-19 data. Rather than relying on traditional methods, this research utilizes evidence-based prediction techniques, which have been shown to be more accurate and efficient. The use of these intelligent forecasting models enables local health authorities to create more precise and effective preventive measures, especially in the face of future outbreaks.

This study is particularly innovative in its use of hybrid forecasting models by machine learning for Malaysia's future pandemics, such as avian flu or novel coronavirus strains. According to Moore [34], the scenario is for the next possible new pandemic of avian influenza virus strain H7N9 or a novel coronavirus. The predictive models developed are more precise, accurate, and efficient in anticipating the dynamic spread of the virus. This approach has been tested on real-world data, including daily new cases, daily new death cases, and daily new recovered cases of COVID-19, making it a valuable tool for public health officials and researchers. This research also has significant implications for

future outbreaks, particularly in countries with tropical rainforests such as Malaysia. By predicting the spread of COVID-19 early on, this model can help policymakers build better healthcare facilities, take legislative action, and avoid economic losses. While a vaccine is now available, this model remains useful in accurately forecasting and preventing the impact of future pandemics, including those caused by new virus strains.

This study's innovative and evidence-based methods make a valuable contribution to pandemic prediction and prevention, providing significant insights that can be used to mitigate the impact of future outbreaks. The implications of this research extend to public health authorities, policymakers, and researchers worldwide, offering powerful tools for mitigating the devastating effects of pandemics. The remainder of this paper is structured as follows. Materials and Methods goes into detail about the method we used to develop our proposed model. The hybrid ARIMA–SVM model used in this study is then briefly described. The results and discussion present the performance of our proposed model based on three known COVID-19 case datasets. Finally, we wrap up the article and make suggestions for future research.

2. Materials and Methods

2.1. ARIMA Modelling

The ARIMA (p, d, q) autoregressive integrated moving average model is one of the families in time-series forecasting that is widely used for time-series forecasting series datasets due to its flexibility with different time categories [16]. It also explicitly considers several standard patterns in time-series analysis, allowing for a powerful and easy-to-use way to produce accurate time-series forecasts. However, limitations may occur due to the existence of assumptions of a linear form that represents a linear relationship between the future value of the time series with the current value, the past value, and random noise in the model [15–17,21,26]. In the ARIMA model, p and q are the numbers of the autoregressive and moving average terms, and they are always listed in the order of the model, while d is the integer representing the differential order. The ARIMA model type with mean μ is represented mathematically as follows:

$$y_t = \beta + \theta_1 y_{t-1} + \theta_2 y_{t-2} + \cdots + \theta_p y_{t-p} + e_t - \varnothing_1 e_{t-1} - \varnothing_2 e_{t-2} - \cdots - \varnothing_q e_{t-q} \quad (1)$$

where y_t and e_t are the actual value and the random error at time t, respectively. Both are assumed to be independently and identically distributed (*iid*) with a mean 0 and a constant variance of σ^2; θ_i $(i = 1, 2, \ldots, q)$ and $\varnothing_j (j = 0, 1, 2, \ldots, q)$ are the model parameters that need to be predicted.

2.2. Support Vector Machines Modelling

The Support Vector Machine (SVM) introduced by Vladimir Vapnik [18], which incorporates statistical learning theory, can handle larger dimensional data better, even with a small number of instances generalizability. Because the models select boundary support vectors from the input data, they process the data quickly. The SVM regression function is written as follows.

For linear and regressive dataset $\{x_i, y_i\}$ the function is formulated as follows:

$$f(x) = w^T x + b \quad (2)$$

The coefficient w and b are estimated by minimizing.

$$\frac{1}{2} w^T w + C \frac{1}{n} \sum_{i-1}^{n} L_\varepsilon(y_i, f(x_i)) \quad (3)$$

where ℓ_ε is called the ε-intensive loss function and is formulated as follows:

$$\ell_\varepsilon(y, f(x)) = \begin{cases} 0 \text{ if } |y - f(x)| \le e \\ |y - f(x)| \quad \text{others} \end{cases} \tag{4}$$

Equation (3) can be transformed to the following constrained formulation by introducing positive slack variables ξ and ξ_i^*:

$$\begin{gathered} min \frac{1}{2} w^T w + C \sum_{i=1}^{n} (\xi_i + \xi_i^*) \\ w x_i + b_i - y_i \le \xi + \xi_i^* \\ -w x_i - b_i + y_i \le \xi + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \\ i = 1, 2, \ldots, N \end{gathered} \tag{5}$$

We always use dual theory to convert the above formula into a convex quadratic programming problem when solving it. Adding the Lagrange Equation (5) results in the following term:

$$min \frac{1}{2} \sum_{i,j=1}^{n} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) \alpha_i^T \alpha_j - \sum_{i=1}^{n} \alpha_i^* (y_i - e) - \alpha_i (y_i + e) \tag{6}$$

Subject to

$$\sum_{i=1}^{y} (\alpha_i - \alpha_i^*) = 0\,, \quad \alpha_i, \alpha_i^* \in [0, C]$$

When a dataset cannot be regressed linearly, we map it to a high dimension feature space and regress it linearly. The following is the formulation:

$$min \frac{1}{2} \sum_{i,j=1}^{n} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) \varphi(x_i)^T \varphi(x_j) - \sum_{i=1}^{n} \alpha_i^* (y_i - e) - \alpha_i (y_i + e) \tag{7}$$

Subject to

$$\sum_{i=1}^{n} (\alpha_i - \alpha_i^*) = 0\,, \quad \alpha_i, \alpha_i^* \in [0, C]$$

Let $K(X_i, X_j) = \{\varphi(X_i) \cdot \varphi(X_j)\} = \varphi^T(X_j) \varphi(X_i); K(x, x)$ is the inner product of feature space and is called kernel function. Any symmetric function that satisfies Mercer condition can be used as Kernel function [19]. The Gaussian kernel function is specified in this study.

$$K\left(x_i, x_j\right) = exp(-||x_i - x_i||^2 / (2\sigma^2)) \tag{8}$$

SVMs were used to estimate the nonlinear behaviour of the forecasting dataset because Gaussian kernels perform well under general smoothness assumptions [22].

2.3. Least-Square Support Vector Machines Modelling

The Least-Squares Support Vector Machines (LSSVM) proposed by Suykens and Vandewalle [35] is a modification of the standard SVM. LSSVM formulates the training process by solving linear problem quicker than SVM through quadratic programming. Additionally, this model is also more time efficient when analysing huge data. Consider a given training set $\{(x_i,\ y_j),\ i = 1, 2, \ldots n\}$ with $x_i \in R^n$ as input data and $y_i \in R$ as output data. LSSVM defines the regression function as:

$$\min J\ (\omega, e) = \frac{1}{2} \omega^T \omega + \frac{C}{2} \sum_{i=1}^{n} e_i^2 \tag{9}$$

Subject to

$$y_i = \omega^T \varphi(x_i) + b + e_i; \;\; i = 1, 2, 3 \ldots n$$

where ω is the weight vector; y is the regularization parameter where it determines the trade-off between the training error minimization and smoothness of the estimated function; e is the approximation error; $\varphi(.)$ is the nonlinear function; and b is the bias term. Constrained optimization of Equation (9) can be translated to unconstrained optimization by constructing Lagrange function. This can be obtained by using Karush–Kuhn–Tucker (KKT) condition, where it partially differentiates with respect to $\omega, b, e,$ and $\varphi(.)$:

$$\begin{bmatrix} 0 & I_v^t \\ I_v & \Omega + c^{-1} I \end{bmatrix} \begin{bmatrix} b \\ a \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix}$$

where $y = [y_1, \ldots, y_n]^T$; $I_v = [1, \ldots, 1]^T$; $a = [a_1, \ldots, a_n]^T$; $\Omega = \left\{ \Omega_{ij} \middle| \, i, j = 1 \ldots n \right\}$, $\Omega_{ij} = \varphi(x_i)^T \varphi(x_i) = K(x_i, \, x_j)$; $K(.)$ is the Radial Basis Function (RBF) kernel function that obtains a and b by calculating linear operations.

2.4. Proposed Hybrid Model

Despite the various time-series models presented, the accuracy, efficiency, and precision of time-series forecasts are becoming crucial for many decision-making processes today. However, these factors do not appear in ARIMA and SVM models. This is also the main reason why the time-series forecasting model is crucial, more demanding, and dynamic, as well as actively researched in many fields of study. ARIMA and SVM models have also prevailed in their linear or nonlinear domains [15,25,26]. However, none of these are generic principles that can be generalized to all situations. Therefore, a hybrid approach using both linear and non-linear modelling capabilities is recommended. This approach is mainly proposed to improve the overall prediction effectiveness. Therefore, there is no research on how to improve the effectiveness of predictive models created in Malaysia, particularly in the case of COVID-19.

There are two reasons for using hybrid models in this study. First, a single ARIMA and SVM model may not be sufficient to identify all the time series' characteristics. The second assumption is that one or both cannot recognise the actual data generation process. This study's hybrid models were built in two stages. Part I discusses linear autocorrelation composition, and Part II discusses nonlinear components. Thus,

$$y_t = \ell_t + \mathcal{N}_t \tag{10}$$

where ℓ_t and $\mathcal{N}_t$ are denoted as the linear composition and the nonlinear component, respectively. Based on the data, these two parts must be approximated. Part I focuses on linear modelling, which employs the ARIMA model to model the linear composition. The model from the first model included residuals, which are nonlinear interactions that cannot be modelled by a linear model or possibly a linear relationship. Thus,

$$\ell_t = \left[\sum_{i=1}^{p} \theta_i z_{t-i} - \sum_{i=1}^{p} \varnothing_j e_{t-j} \right] + \varepsilon_t = \hat{\ell}_t + \varepsilon_t \tag{11}$$

Let ε_i denoted as the residual from the linear model at time t. Then,

$$\varepsilon_t = y_t - \hat{\ell}_t$$

where $\hat{\ell}_t$ is the predicted value for time t from the estimated relationship in (1), with ε_t as the residual at time t from the linear model. The residual dataset after ARIMA fitting will only contain non-linear relationships that can be represented by a linear model [15]. The first stage results, which include forecast values and residuals from linear modelling, are then used in Part II.

Following Part II, the emphasis is on nonlinear modelling, where LSSVM is used to predict the nonlinear connection that occurs in residuals of linear modelling and original data. Then, the residual can be calculated using LSSVM by modelling various configurations as follows:

Part II focuses on nonlinear modelling, and LSSVM is used to model the nonlinear (possibly linear) relationship that occurs in residuals of linear modelling as well as original data. The residual can then be calculated using LSSVM by modelling different configurations as follows:

$$\varepsilon_t = f\left(\varepsilon_{t-1}, \varepsilon_{t-2}, \ldots \varepsilon_{t-n}\right) + e_t \tag{12}$$

$$\varepsilon_t = f\left(\varepsilon_{t-1}, \varepsilon_{t-12}\right) + e_t \tag{13}$$

$$y_t = f\left(y_{t-1}, y_{t-12}, \ \hat{\ell}_t\right) + e_t \tag{14}$$

$$y_t = f\left(y_{t-1}, y_{t-12}\right) + e_t \tag{15}$$

where f is a nonlinear function determined by the LSSVMs model and e_t is the random errors. Thus, the hybrid forecast is

$$\hat{y}_t = \hat{\ell}_t + \hat{\mathcal{N}}_t \tag{16}$$

Equations (12) and (13) can be identified as $\hat{\mathcal{N}}_t$, therefore the forecasted values can be achieved by summation of linear and nonlinear components. Figure 1 shows the functional flowchart of hybrid models.

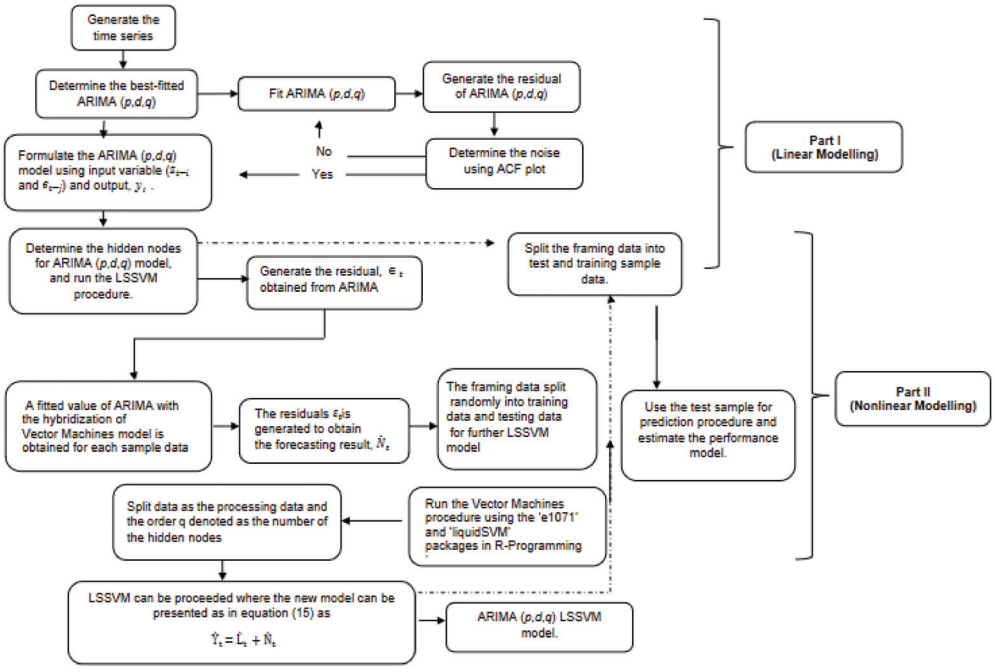


Figure 1. Flowchart process for hybrid ARIMA-LSSVM models.

In short, the proposed hybrid process methodology is divided into two parts. The ARIMA model is used to analyse the linear composition problem in Part I. Part II develops an LSSVM model to model the residuals from Part I. Because the ARIMA model in Part I cannot handle the nonlinear component of the data, the residuals of the linear model will include information about the nonlinearity. The LSSVM results can be utilised as

forecasts of the ARIMA model's error terms. The hybrid model defines various patterns by combining the distinct features and strengths of the ARIMA and LSSVM models. As a result, it is more effective to model linear and non-linear patterns separately with two different models and then re-hybridize the forecast results to improve overall modelling and forecasting performance.

2.5. Proposed Algorithm

Step 1: Three selected time series of COVID-19 cases datasets (1 October 2020–4 November 2022), namely daily new positive cases, daily new deaths cases, and daily new recovered cases, are generated in R programming Language.
Step 2: Each of the generated datasets is defined as $\{X_{1i} = x_{11}, x_{12}, x_{13}, \ldots, x_{n1}\}$, $\{X_{2i} = x_{21}, x_{22}, x_{23}, \ldots, x_{2n}\}$, and $\{X_{3i} = x_{31}, x_{32}, x_{33}, \ldots, x_{3n}\}$ for daily new positive cases, daily new deaths cases, and daily new recovered cases, respectively. Then, the best ARIMA (p, d, q) is selected after checking the autocorrelation function (ACF) plot of ARIMA (p, d, q) residuals. The best fitted value for daily new positive cases is ARIMA (2, 1, 2), while it is ARIMA (1, 1, 2) and ARIMA (0, 1, 1) for daily new fatalities cases and daily new recovered cases of COVID-19, respectively.
Step 3: The fitted value, $y_{t-i} = (y_{t-1}, y_{t-2}, \ldots, y_{t-m})$ and the residuals $\varepsilon_{t-i} = (\varepsilon_{t-1}, \varepsilon_{t-2}, \ldots, \varepsilon_{t-n})$.
Step 4: Combine the values in step 3 as a set of input variables to obtain the output y_t
Step 5: The ARIMA (p, d, q) is defined by the order of q. According to the information in step 4, Vector Machines is carried out to examine the residuals to obtain the output L_t using R-programming Language.
Step 6: A fitted value of ARIMA with the hybridization of Vector Machines model is obtained for all sample data. Then, the residuals ε_t is generated to obtain the forecasting result $\widehat{N}_t$.
Step 7: The framing data split randomly into training data and testing data for further Vector Machines modelling. Run the Vector Machines procedure using the "e1071" and "liquidSVM" package in R-Programming Language.
Step 8: The two modifiable parameters of the LSSVM technique (γ and σ) derived by objective function minimization such as mean square error (MSE). The grid-search method updates the parameters exponentially in the specified range using predetermined equidistant steps.
Step 9: Assume the split data as the processing data and the order q as in Step 5. Therefore, the combine forecast as in Equation (16): $\hat{y}_t = \hat{\ell}_t + \hat{N}_t$
Step 10: Estimate the model performance using the statistical measurement which are MSE, RMSE, MAE, and MAPE.

2.6. Forecasting Evaluation Criteria

In order to assess the overall performance of the proposed hybrid models, the one of a kind statistical measurements standard which accompanied by [15,16,36] including MAE (Mean Absolute Error), MAPE (Mean Absolute Percentage Error), MSE (Mean Squared Error), and RMSE (Root Mean Squared Error) are used.

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n}\left|\widehat{y}_t - y_t\right|$$

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{\widehat{y}_t - y_t}{y_t}\right| \times 100$$

$$\text{MSE} = \frac{1}{n}\sum_{t=1}^{n}(\widehat{y}_t - y_t)^2$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(\widehat{y}_t - y_t)^2} = \sqrt{\text{MSE}}$$

In time-series analysis, measurement tools such as Akaike's information criterion (AIC) and the Bayesian information criterion (BIC) are commonly used to determine the appropriate length for distributed lag for the ARIMA model. As a result, model selection is based on the model with the lowest AIC and BIC values to provide measures of model performance, resulting in the selection of the best ARIMA model. Meanwhile, three parameters such as *C* are used as measurement tools to determine the best fitted model for LSSVMs models. Meanwhile, for the LSSVMs models, two parameters such as γ and σ are used as the measurement tools to determine the best fitted model.

Incorrect LSSVM model parameter selection can lead to over or underfitting of the training data. The parameter sets of the LSSVMs model with the lowest MSE value, as with the ARIMA model, will be selected for use in the best fitting model. As a result, for the hybrid models, the ARIMA first functioned as a pre-processor, filtering the linear pattern of datasets. The ARIMA model's error term is then fed into the SVM in the hybrid models. LSSVMs were used to reduce the ARIMA error function.

3. Results and Discussion

3.1. Application of the Hybrid Model of COVID-19 in Malaysia

This section examined the proposed model's performance in two ways: first, the performance of the proposed models compared to ARIMA, SVM, LSSVM, ARIMA–SVM, and ARIMA–LSSVM models; second, the percentage improvement of the proposed models compared to ARIMA and SVM models. Since the World Health Organization (WHO) declared COVID-19 to be a worldwide pandemic, the COVID-19 time-series datasets have been extensively studied. The predictive capability of the developed novel models was then compared using three well-known datasets of daily COVID-19 cases in Malaysia—daily new positive cases data, daily new fatalities cases data, and daily new recovered cases data—to demonstrate the performance of the proposed model in terms of accuracy, effectively, and accurately. All these data are reported from the 1 October 2020 to 4 November 2022 and retrieved from the COVIDNOW website at https://covidnow.moh.gov.my/, accessed on 10 January 2023.

The minimum value of new death, new cases, and new recovered cases in Table 1 is 0, 2600, and 1.8, respectively, while the maximum value of new cases, death, and recovered cases is 33,872.0, 592, and 33,406, respectively. Similarly, the mean and median for new cases, deaths, and recovered cases are 6322.7, 47.51, and 6415.5, respectively, where the parentheses indicate the median (3471, 11, 3447.0). The first quartile values for daily new cases, death cases, and recover cases are 1922, 4, and 1843, respectively. The number of daily new cases, deaths, and recoveries in the third quartile is 6824, 58, and 6775, respectively. Furthermore, the standard deviations for new cases, deaths, and recoveries are 7097.8, 81.12, and 7058.3 percentiles, respectively.

Table 1. Descriptive statistics of COVID-19 daily new cases, death, and recovered cases of Malaysia.

	New Case	New Death	New Recovered
Min	2.60000×10^2	0	1.8
1st Qu	1.9220×10^3	4	1.8430×10^3
Median	3.4710×10^3	11	3.4470×10^3
Mean	6.4155×10^3	47.5098	6.3227×10^3
3rd Qu	6.8240×10^3	58	6.7750×10^3
Max	3.3406×10^4	592	3.3872×10^4
SD	7.0978×10^3	81.1215	7.0583×10^3

Furthermore, this section discusses the process of proposed models for both parts, i.e., Part I (Linear Modelling) and Part II (Nonlinear Modelling), using three well-known COVID-19 datasets, namely daily new positive cases, daily new deaths cases, and daily new recovered cases, to demonstrate the effectiveness of the proposed models. Both linear

and nonlinear modelling, as well as the data used in this study, are carried out using R programming.

Part I (Linear Modelling)—ARIMA is used to generate the best ARIMA model for the daily new positive case dataset (2, 1, 2). ARIMA is the best fitting ARIMA model for the daily new death case dataset (1, 1, 2). Meanwhile, the best ARIMA model is reported as ARIMA in the case of the daily new recovered cases dataset (0, 1, 1). Table 2 summarizes the results of this ARIMA (p, d, q) model. Table 3 displays the estimates for all parameters. The p-values for all parameters are small, as shown in this table. As a result, for confirmed, recovered, and death cases, the models were statistically significant and could be used to forecast the future [37,38].

Table 2. The best ARIMA (p, d, q) model selection.

COVID-19 Daily Cases	ARIMA (p, d, q)	AIC	BIC
Daily New Positive Cases	(2, 1, 2)	12,564.54	12,587.73
Daily New Deaths Cases	(1, 1, 2)	6930.12	6948.63
Daily New Recovered Cases	(0, 1, 1)	13,044.74	13,054.01

Table 3. Parameter estimates of ARIMA models and their p-values.

Model Parameters	Estimate	Z-Stat	p-Value
New Case ARIMA (2, 1, 2)			
θ_1	1.2408	120.085	2.2×10^{-16}
θ_2	−0.9715	−98.320	2.2×10^{-16}
φ_1	−1.2628	−42.225	2.2×10^{-16}
φ_2	0.8738	48.102	2.2×10^{-16}
Recovered Case ARIMA (0, 1, 1)			2.2×10^{-16}
φ_1	−0.3473	−9.953	2.2×10^{-16}
Death Case ARIMA (1, 1, 2)			2.2×10^{-16}
θ_1	0.8595	19.852	2.2×10^{-16}
φ_1	−1.6196	−35.651	2.2×10^{-16}
φ_2	0.7039	20.432	2.2×10^{-16}

Part II (Nonlinear Modelling)—Based on the concepts of support vector machine design and the use of pruning algorithms in R-programming software, an optimal machine learning algorithm was created. For the daily new positive COVID-19 cases datasets, parameters $\gamma = 264$, $\sigma = 0.008$ show the smallest values of MSE i.e., 6,661,412 (see Table 4). Therefore, this parameters value was selected for use in the best-fitting model for the datasets of daily new positive COVID-19 cases. Whereas the smallest value of MSE is 250.887 and 21114252 (Table 4), with parameters $\gamma = 877$, $\sigma = 0.006$ and $\gamma = 334$, $\sigma = 0.008$ are selected as the best-fitting model for daily new death and daily new recovered cases of COVID-19, respectively.

Table 4. LSSVMs Model Parameters for the daily new COVID-19 cases datasets.

COVID-19 Daily Cases	LSSVM Parameter	MSE
Daily New Positive Cases	$\gamma = 11$, $\sigma = 0.008$	11,432,512
	$\gamma = 38$, $\sigma = 0.008$	10,235,488
	$\gamma = 74$, $\sigma = 0.008$	9,025,413
	$\gamma = 110$, $\sigma = 0.008$	8,014,123
	$\gamma = 264$, $\sigma = 0.008$	6,661,412

Table 4. *Cont.*

COVID-19 Daily Cases	LSSVM Parameter	MSE
Daily New Deaths Cases	$\gamma = 25$, $\sigma = 0.006$	1678.364
	$\gamma = 56$, $\sigma = 0.006$	1233.481
	$\gamma = 277$, $\sigma = 0.006$	965.143
	$\gamma = 436$, $\sigma = 0.006$	554.368
	$\gamma = 877$, $\sigma = 0.006$	250.887
Daily New Recovered Cases	$\gamma = 54$, $\sigma = 0.008$	28,412,113
	$\gamma = 89$, $\sigma = 0.008$	27,140,039
	$\gamma = 125$, $\sigma = 0.008$	26,412,142
	$\gamma = 275$, $\sigma = 0.008$	23,032,256
	$\gamma = 334$, $\sigma = 0.008$	21,114,252

3.1.1. New Positive Cases Data Forecasts

The daily new positive cases datasets series contains 765 data points and is recoded from 1 October 2020 to 4 November 2022 (see Figure 2). The number of daily new positive COVID-19 cases in Malaysia has increased significantly twice since July 2021, but has now dropped below 5000 new cases. However, it has continued and increased to a maximum of 33,406.00 around March–April 2022. This figure is expected to fall precipitously until 5 November 2022. The COVID-19 datasets have been extensively used with a wide range of linear and nonlinear time-series models, including ARIMA and machine learning methods [7–9,11,13,16,19–26]. The analysis of daily new positive cases of COVID-19 is critical as an indicator of the effectiveness of preventive measures that have been taken, are being taken, and will be taken by authorities to control the spread of this epidemic more effectively.

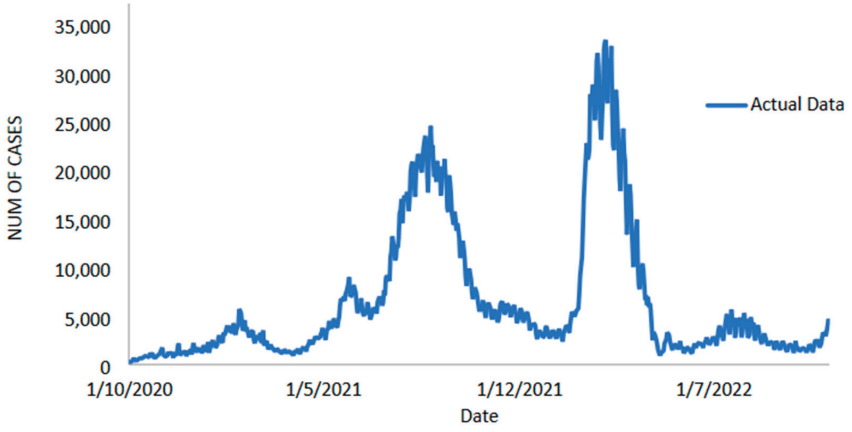


Figure 2. Malaysian daily new positive COVID-19 cases (1 October 2020 to 4 November 2022).

Therefore, a similar approach to that used by Aisyah et al. [15] is used to investigate the performance of the proposal models on daily new positive cases of COVID-19 datasets, where the dataset is divided into two samples, known as training sample and testing sample. According to Aisyah et al. [15] and Nurul Hila et al. [16], datasets should be divided into two parts to achieve the best results: 70–80% for training and the remaining 20–30% for testing [39,40]. The training data are used to assemble the models, while the testing data are used to evaluate the forecasting performances of the models based on statistical measurements. Thus, the daily new positive cases of the COVID-19 dataset are divided into two samples in this study: the training dataset and the test dataset. The training datasets contain 612 observations from day 1 to day 612, accounting for 80% of

the datasets used exclusively to formulate from 1 October 2020 to 4 June 2022. In order to evaluate the forecasting performance of proposed models, the test sample datasets used approximately 153 observations from days 613–765 (20%) from the 5 June 2022 to the 4 November 2022

Table 5 displays the performance of the proposed model on the daily new positive COVID-19 case datasets. The proposed models produced results in terms of measurement error terms, namely MSE and MAE, which have smaller values of 10634.1142 and 46.54471, respectively. Similar results were obtained from the testing datasets, with MSE, MAPE, RMSE, and MAE values of 25478.114, 0.01547, 159.6182, and 75.6987, respectively. The findings are examined in greater detail using figures such as those shown in Figure 3a–e based on these numerical results. The estimated values for the proposed model (test sample) of daily new positive COVID-19 cases are shown in this figure. The proposed model line, as seen in this figure, closely matches the actual data. Figure 4a–e show the estimated values of our model for test sample data of ARIMA, SVM, LSSVM, ARIMA-SVM, and ARIMA–LSSVM models for COVID-19 cases. A comparison of the proposed model's (ARIMA–LSSVM) lines for the test sample (Figure 4e) with the lines from the ARIMA, SVM, LSSVM, and ARIMA–SVM models (Figure 4a–d) clearly shows that the proposed model's lines are somewhat close to the actual data. When we compared the performance of our proposal models to the performance of ARIMA, SVM, LSSVM, and ARIMA–SVM models, we discovered that our proposal models are efficient, accurate, and precise. In addition, the number of daily new positive COVID-19 cases is plotted, as shown in Figure 5. The daily new positive cases of COVID-19 for Malaysia are forecasted based on this figure for the next three weeks.

Table 5. Performance measures of the proposed model for daily new positive COVID-19 cases datasets.

MODELS	TRAIN		TEST			
	MSE	MAE	MSE	MAPE	RMSE	MAE
ARIMA	929,843.169	611.0274	298,988.28	0.15167	546.7982	397.57
SVM	8,355,184.483	2001.644	274,588.16	0.15421	524.0116	390.3848
LSSVM	1084.1527	739.5387	83,026.550	0.07580	288.1432	205.6450
ARIMA–SVM	42,552.7137	90.34845	61,223.474	0.05633	247.4337	146.9841
ARIMA–LSSVM	10,634.1142	46.54471	25,478.114	0.01547	159.6182	75.6987

Based on Table 6, we examined the performance of the proposed models for the daily newly positive COVID-19 cases dataset by comparing the percentages of MSE, MAPE, RMSE, and MAE. The study hypothesis investigates the assumptions of the proposed hybrid model (ARIMA–LSSVM) approach to single ARIMA, SVM, LSSVM models, as well as hybrid ARIMA–LSSVM models. The proposed model outperformed the ARIMA–SVM model in MAE, MAPE, MSE, and RMSE, with improvements of 48.50%, 72.54%, 58.39%, and 35.50%, where the parentheses indicate an ARIMA, SVM, and LSSVM model, respectively, that results in the following: 80.96%, 89.80%, 91.48%, 70.81%; 80.61%, 89.97%, 90.72%, 69.54%; 63.20%, 79.59%, 69.31%, 44.60%. As a result of these findings (Tables 5 and 6 and Figures 3–5), it is possible to conclude that the proposed model produced greater accuracy and efficiency than ARIMA and SVM.

Table 6. Percentage improvement of the proposed models with other forecasting models (the COVID-19 cases of daily new positive cases).

Model	MAE	MAPE	MSE	RMSE
ARIMA	80.9596549	89.80022417	91.47855762	70.80857252
SVM	80.60920917	89.96822515	90.72133554	69.53918577
LSSVM	63.18962289	79.59102902	69.31329316	44.60455773
ARIMA–SVM	48.49871517	72.5368365	58.38505669	35.49051726

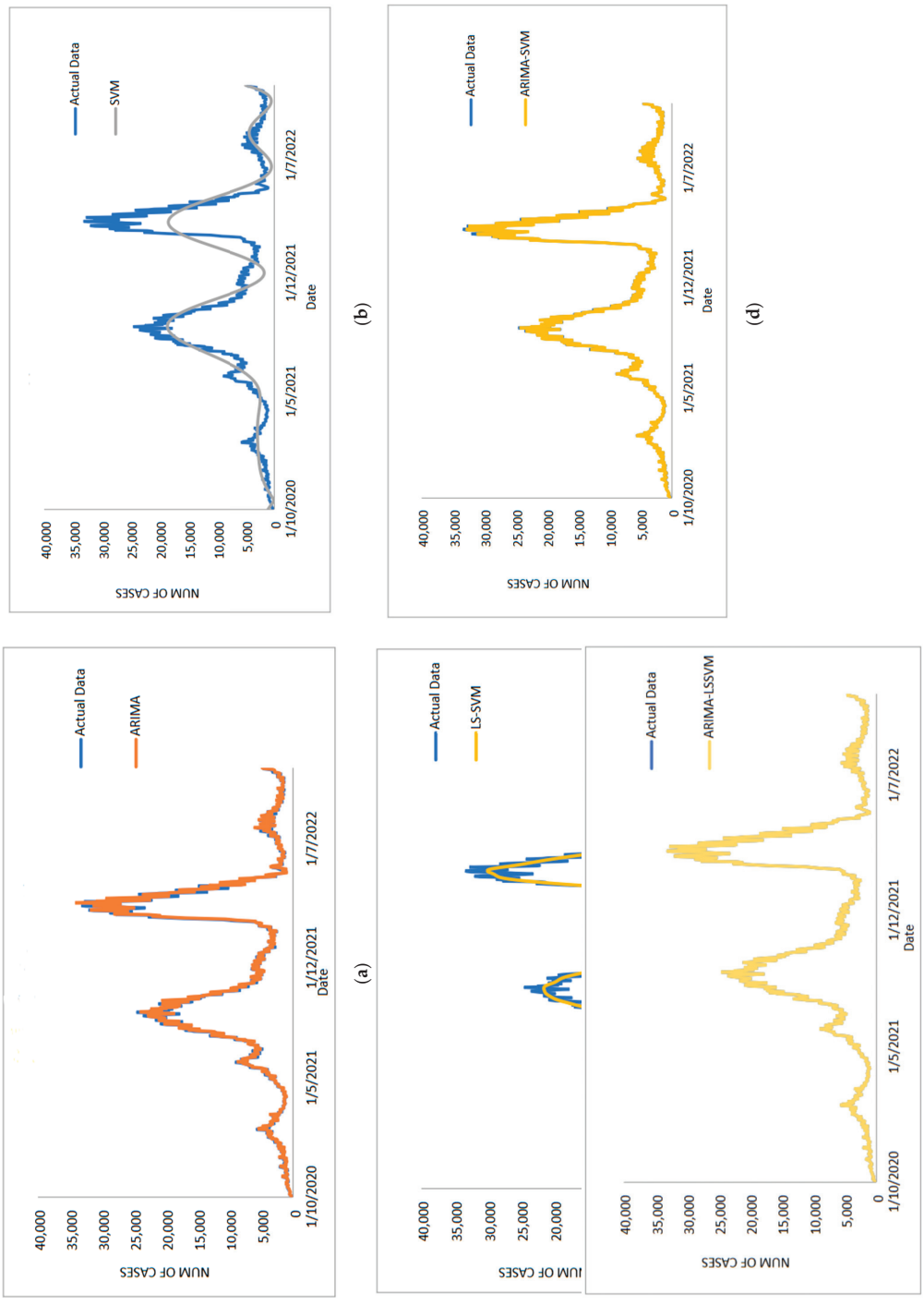
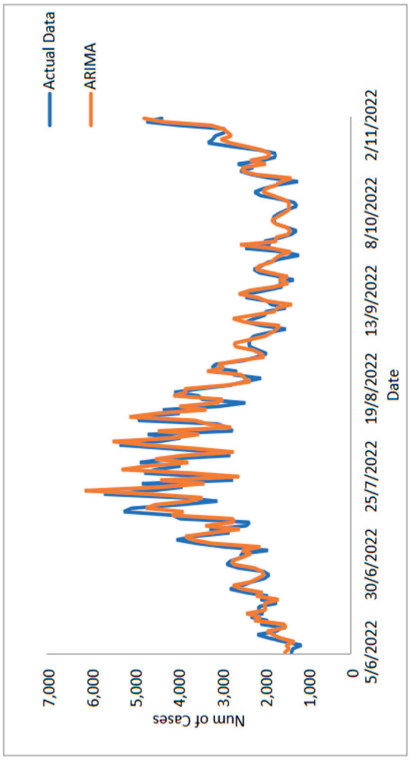
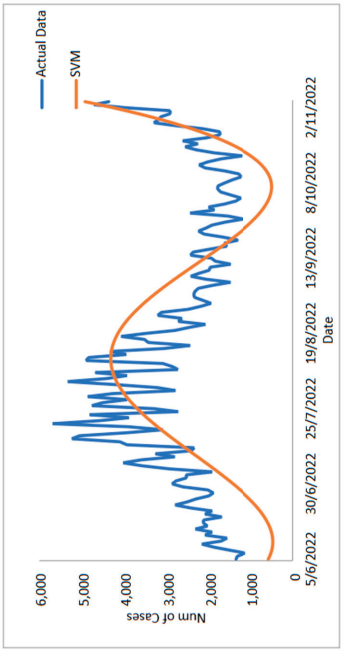


Figure 3. Results obtained from the proposed model for daily new positive COVID-19 cases dataset: (**a**) actual data vs. ARIMA model, (**b**) actual data vs. LSSVM models, (**c**) actual data vs. SVM model, (**d**) actual data vs. ARIMA–SVM models, (**e**) actual data vs. ARIMA–LSSVM models.

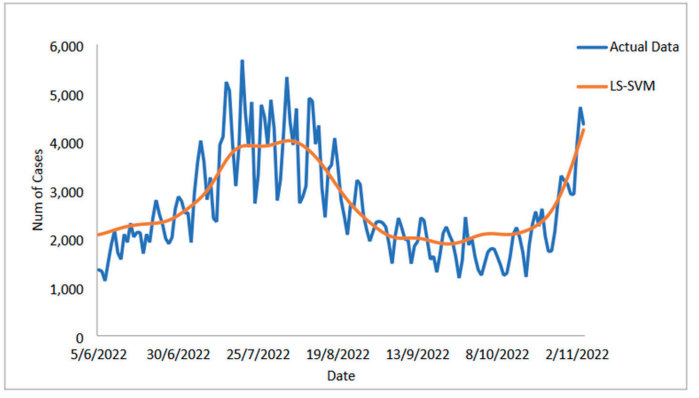


(a)

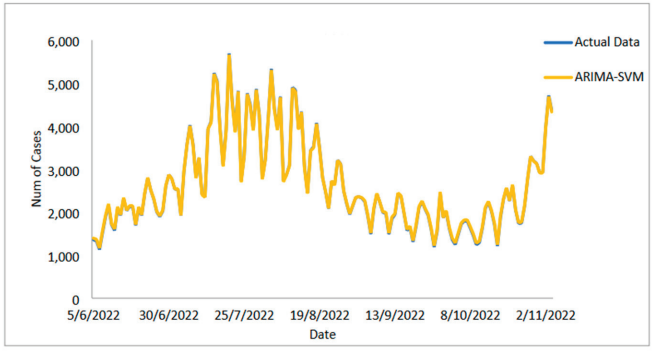


(b)

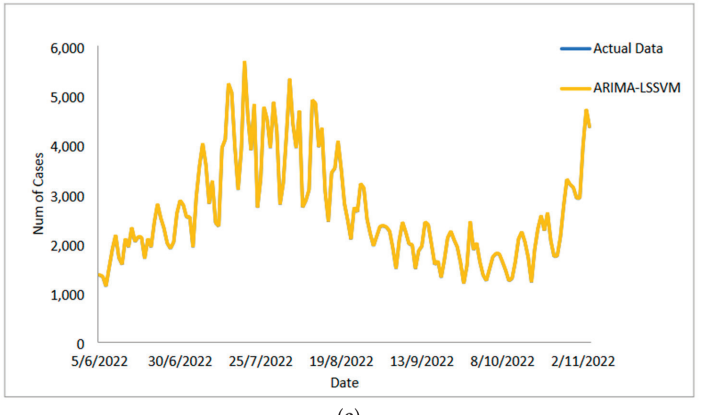
Figure 4. *Cont.*



(c)



(d)



(e)

Figure 4. Models' prediction of daily new positive COVID-19 cases dataset (20% test sample): (**a**) actual data vs. ARIMA model, (**b**) actual data vs. SVM model, (**c**) actual data vs. LSSVM models, (**d**) actual data vs. ARIMA–SVM models, (**e**) actual data vs. ARIMA–LSSVM models.

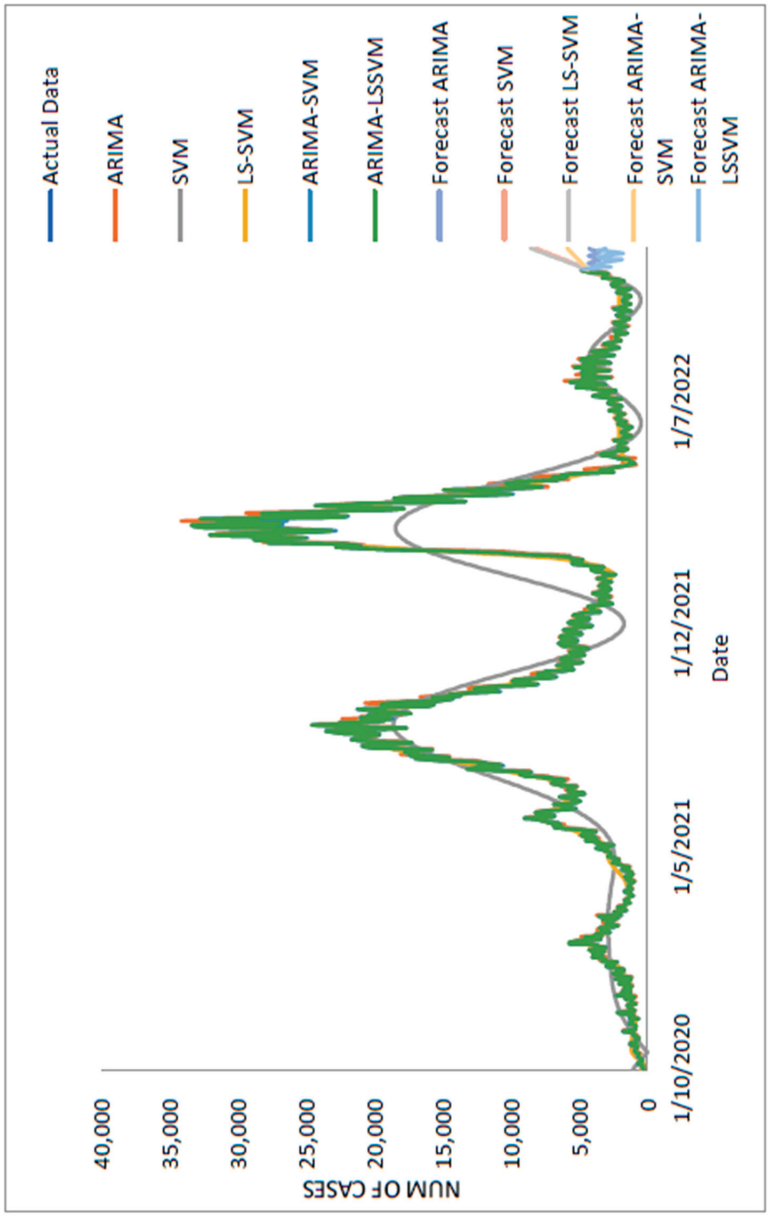


Figure 5. Actual and three weeks ahead forecasted values of ARIMA, SVM, LSSVM, ARIMA–SVM, and ARIMA–LSSVM models for new cases of COVID-19 of the 80% training and 20% testing set.

3.1.2. New Deaths Cases Data Forecasts

In addition to the Malaysian daily new positive COVID-19 cases datasets, the Malaysian daily new deaths cases datasets are taken into account and used to evaluate the performance of the proposed models. This dataset, like the daily new positive dataset and the daily new death case dataset, has a recording period of 1 October 2020 to 4 November 2022 (see Figure 6) and contains 765 data points divided into two samples. As the number of daily positive COVID-19 cases reported rises, so does the number of deaths, which now stands at around 600. The training dataset contains 612 observations (80%) from 1 October 2020 to 4 June 2022, and the test sample contains approximately 153 observations (20%) from 5 June 2022 to 4 November 2022 to evaluate the prediction performance of the proposed model.

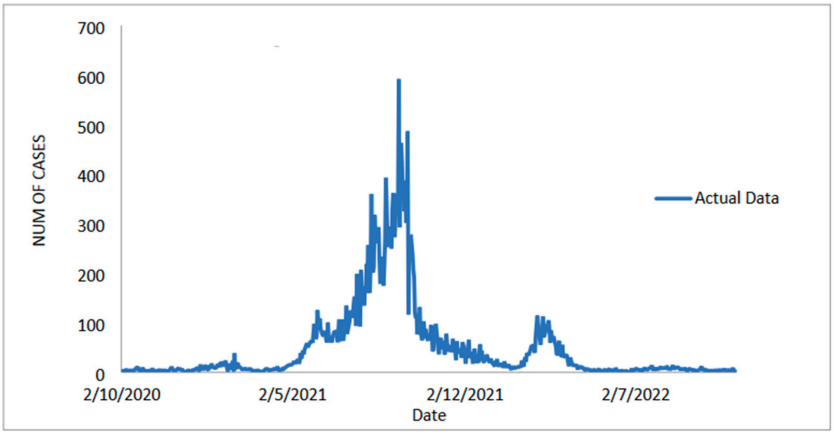


Figure 6. Malaysian daily new deaths COVID-19 cases (1 October 2020 to 4 November 2022).

A similar approach to the daily new positive cases of the COVID-19 dataset was used to study the performance of the proposed model on the daily new death cases of the COVID-19 dataset. The dataset was divided into two samples, namely training sample and testing sample. It accounts for approximately 80% of the daily new death cases in the COVID-19 dataset for the training sample (involving 612 observations with the period 1 October 2020 until 4 June 2022). The remaining 20% is for the test sample, which includes approximately 153 observations from 5 June 2022 to 4 November 2022.

As shown in Table 7, the performance of the proposed models using the daily new deaths datasets from COVID-19 is first characterized by statistical measurements such as MSE, MAPE, RMSE, and MAE. The results for the training data in this table show that the proposed model produces the smallest MSE and MAE values of 19.6422 and 1.03218, respectively, when compared to ARIMA, SVM, LSSVM and ARIMA–SVM. The same pattern can be seen in the test data, where all the statistical measures used have the lowest values when compared to the ARIMA, SVM, LSSVM, and ARIMA–SVM models.

Table 7. Performance measures of the proposed model for daily new deaths COVID-19 cases datasets.

MODELS	TRAIN		TEST			
	MSE	MAE	MSE	MAPE	RMSE	MAE
ARIMA	697.999	11.8083	6.06741	0.56838	2.46321	1.92791
SVM	1409.19	21.8006	5.38920	0.53687	2.32146	1.85605
LSSVM	505.181	11.4309	5.38920	0.53687	2.32146	1.85605
ARIMA–SVM	49.4459	3.53812	0.92630	0.19088	0.96303	0.76230
ARIMA–LSSVM	19.6422	1.03218	0.89114	0.18741	0.94400	0.72364

The study then examines the estimated value of the suggested model for the COVID-19 case dataset for daily deaths, as shown in Figure 7a–e. This graph makes it abundantly clear that the proposed model line and the observed data are nearly identical. Additionally, Figure 8a–e each show the estimated values for the test sample for ARIMA, SVM, LSSVM, ARIMA–SVM and ARIMA–LSSVM the suggested models. Once more, it is obvious that when compared to ARIMA, SVM, LSSVM, and ARIMA–SVM models, the test sample lines for our suggested model (Figure 8e) are somewhat close to the actual data. This demonstrates that the outcomes of our suggested model are in line with prior findings and are more effective, accurate, and precise than those of ARIMA, SVM, LSSVM, and ARIMA–SVM models. The number of daily COVID-19 death cases is also plotted, just like in Figure 9. The daily new death cases of COVID-19 in Malaysia are anticipated to decline because of this number over the course of the following three weeks, indicating a downward trend.

As shown in Table 8, a similar method in the daily addition of positive COVID-19 case dataset is used to investigate the performance of the proposed model for the daily recorded death COVID-19 case dataset using percentages MSE, MAPE, RMSE, and MAE. Again, the percentage of improvement shows that our proposed model outperforms the ARIMA–LSSVM models for all statistical measures, with results of 5.07%, 1.82%, 3.80%, and 1.98%, respectively; there are also improvements (62.47%, 67.03%, 85.31%, 61.68%; 61.61%, 66.74%, 83.92%, 59.90%; 61.01%, 65.10%, 83.50%, 59.34%) for MAE, MAPE, MSE, and RMSE. The ARIMA, SVM, LSSVM, and ARIMA–SVM model results are shown in parentheses. The presented results (see Tables 7 and 8 and Figures 7–9) clearly show that our proposed model outperforms the ARIMA, SVM, LSSVM, and ARIMA–SVM models in terms of efficiency and accuracy.

Table 8. Percentage improvement of the proposed models with other forecasting models (the COVID-19 cases of daily new death cases).

Model	MAE	MAPE	MSE	RMSE
ARIMA	62.46505283	67.02734086	85.31267872	61.67602437
SVM	61.60592539	66.73588924	83.92334934	59.90434808
LSSVM	61.01182619	65.09210796	83.46433608	59.33593514
ARIMA–LSSVM	5.071494162	1.81789606	3.795746518	1.976054744

3.1.3. New Recovered Cases Data Forecasts

The investigation to study the performance of the proposed model is continued with the dataset of new daily recovered cases of COVID-19 in Malaysia. Predicting Malaysia's daily new recovered COVID-19 cases is just as important as the previous two datasets. The data used in this paper include daily observations from 1 October 2020 to 4 November 2022, for a total of 765 data points in the time series (Figure 10). The number of patients recovered from COVID-19 exhibits the same trend, with a significant increase twice. Beginning in July 2021, the number of recovered patients increases exponentially until it reaches over 22,500.00 in August 2021 (the time-series plot is shown in Figure 10) and then drops.

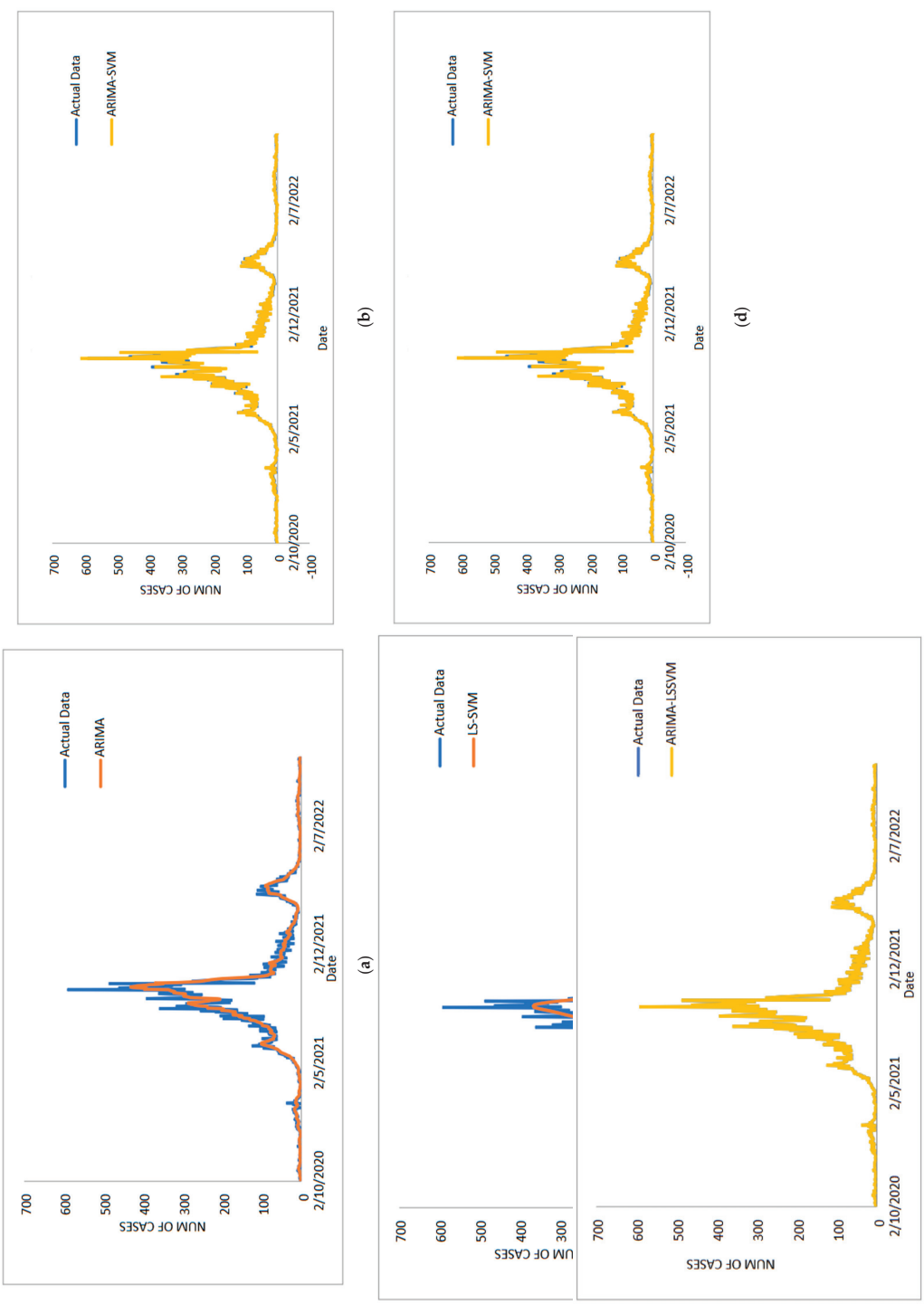
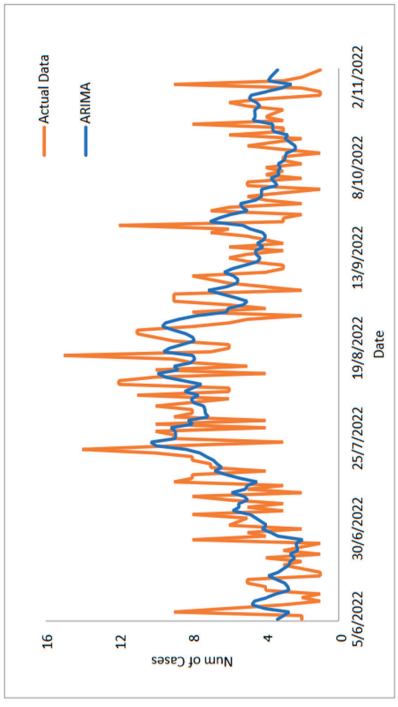
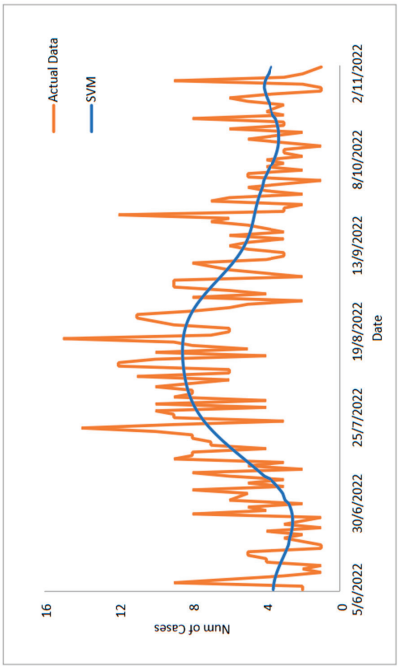


Figure 7. Results obtained from the proposed model for daily new death COVID-19 cases dataset: (**a**) actual data vs. ARIMA model, (**b**) actual data vs. LSSVM models, (**c**) actual data vs. SVM model, (**d**) actual data vs. ARIMA–SVM models, (**e**) actual data vs. ARIMA–LSSVM models.

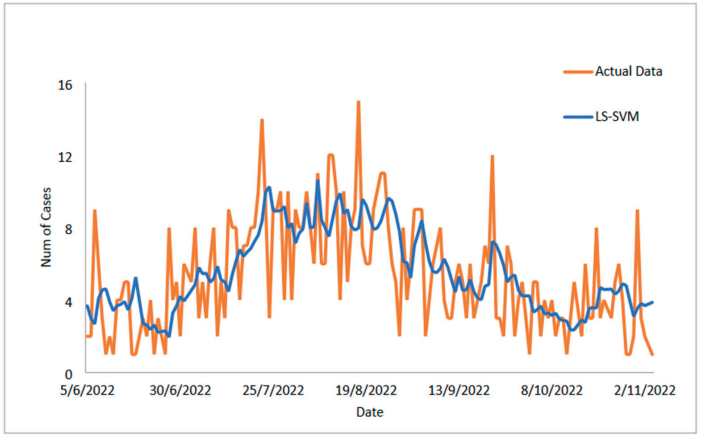


(**a**)

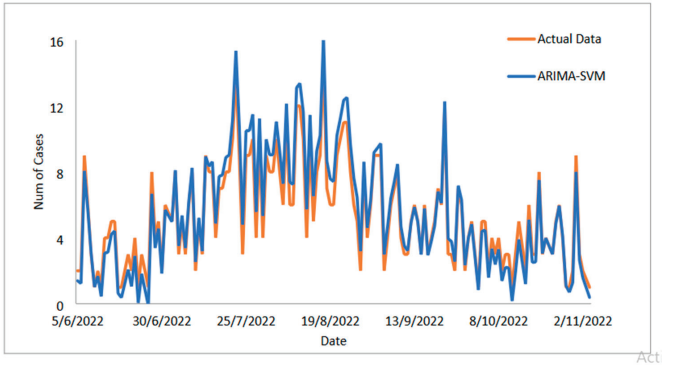


(**b**)

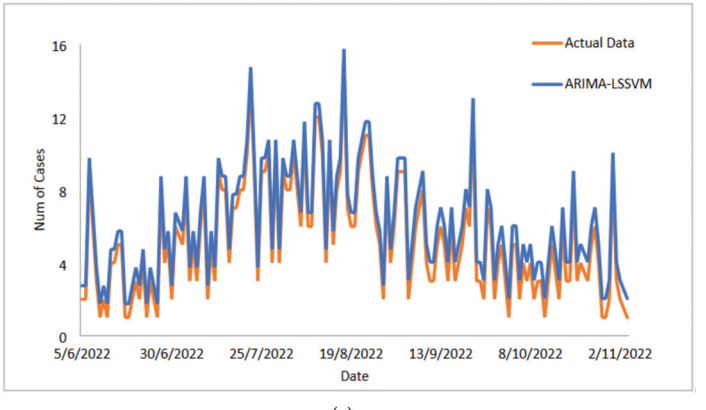
Figure 8. *Cont.*



(**c**)



(**d**)



(**e**)

Figure 8. Models' prediction of daily new death COVID-19 cases dataset (20% test sample): (**a**) actual data vs. ARIMA model, (**b**) actual data vs. SVM model, (**c**) actual data vs. LSSVM models, (**d**) actual data vs. ARIMA–SVM models, (**e**) actual data vs. ARIMA–LSSVM models.

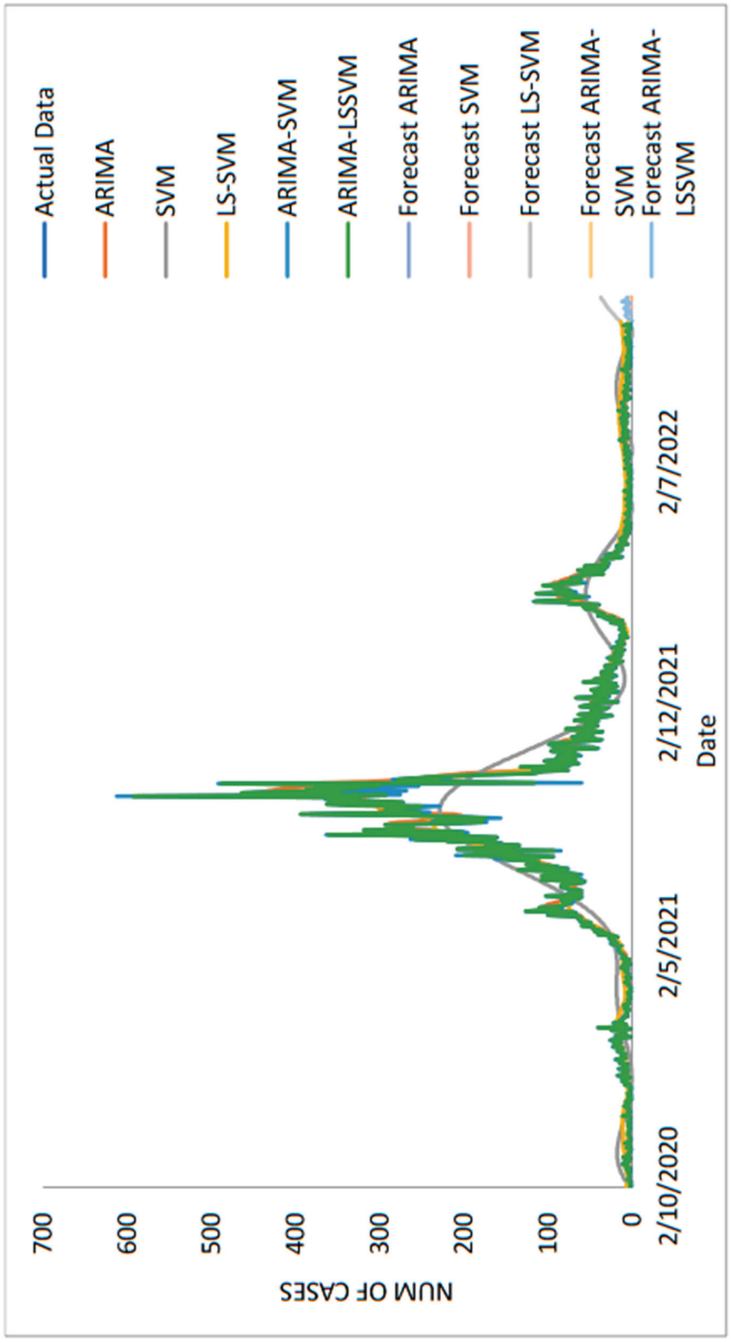


Figure 9. Actual and three-weeks-ahead forecasted values of ARIMA, SVM, LSSVM, ARIMA–SVM and ARIMA–LSSVM models for daily new deaths COVID-19 cases of the 80% training and 20% testing set.

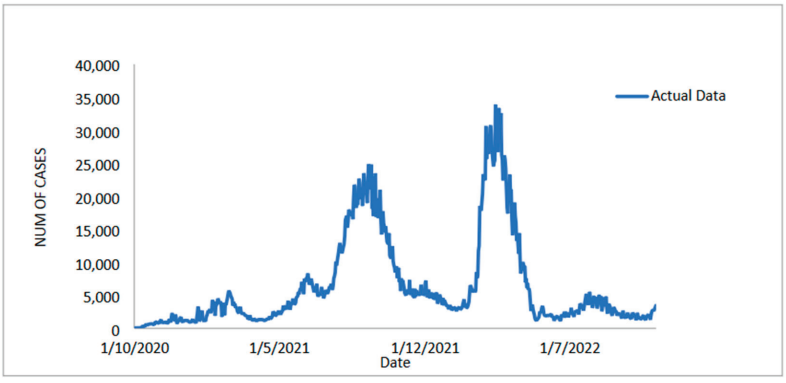


Figure 10. Malaysian daily new recovered COVID-19 cases (1 October 2020 to 4 November 2022).

However, around March–April 2022, the number of recovered COVID-19 cases increased again to a maximum of 33,872.00, then decreased and showed a relatively stable movement after that. This dataset is also divided into two samples, namely the training dataset and the test dataset. The training dataset, which included 612 observations (80%) from 1 October 2020 to 4 June 2022, was used in the same way as the previous datasets to formulate the model. In contrast, the test sample uses approximately 153 observations (20%) for the period 5 June 2022–4 November 2022. Table 9 displayed the performance of the proposed model on the daily new recovered COVID-19 case datasets based on training and testing samples. The results in Table 9 clearly show that the proposed training sample model produces the smallest MSE and MAE values, with 47602.551 and 80.2214, respectively, when compared to the MSE and MAE models of the ARIMA, SVM, LSSVM, and ARIMA–LSSVM models. For the test sample, the same scenario as the training sample produced the smallest MSE, MAPE, RMSE, and MAE with values of 13004.11, 0.0125, 114.0351, and 54.14471, respectively, when compared to ARIMA, SVM, LSSVM, ARIMA–SVM and ARIMA–LSSVM models.

Table 9. Performance measures of the proposed model for daily new recovered COVID-19 cases datasets.

MODELS	TRAIN		TEST			
	MSE	MAE	MSE	MAPE	RMSE	MAE
ARIMA	1,802,678.36	804.4378	271,462.22	0.1560	521.0203	387.2768
SVM	7,636,804.13	1890.917	239,672.00	0.1504	489.5630	371.6573
LSSVM	1,206,113.52	723.9413	149,871.53	0.1127	387.1324	285.9190
ARIMA–SVM	99,205.699	136.8519	26,108.02	0.0396	161.5797	104.1002
ARIMA–LSSVM	47,602.551	80.2214	13,004.11	0.0125	114.0351	54.14471

Figure 11a–e shows the estimated value of the dataset for daily new recovered COVID-19 cases for the test sample. Once more, this graph demonstrates how closely the predicted value from the proposed models seems to match the actual values. Figure 12a–e present an additional analysis of the outcomes of the proposed model. These plots (Figure 12a–e) show the predicted values for the test samples derived from ARIMA, SVM, LSAVM, ARIMA–SVM and ARIMA–LSSVM models. In these models, however, the proposed model is close to the true value because, as we shall see in Figure 11e, the proposed model dominates them. As shown in Figure 13, the number of daily new recovered COVID-19 cases is plotted. This figure makes it abundantly clear that the suggested model maintains the data's original sharpness. The daily new recovered COVID-19 cases for Malaysia are predicted from this figure for the upcoming three weeks, and it suggests that these cases will rise in the days to come in Malaysia.

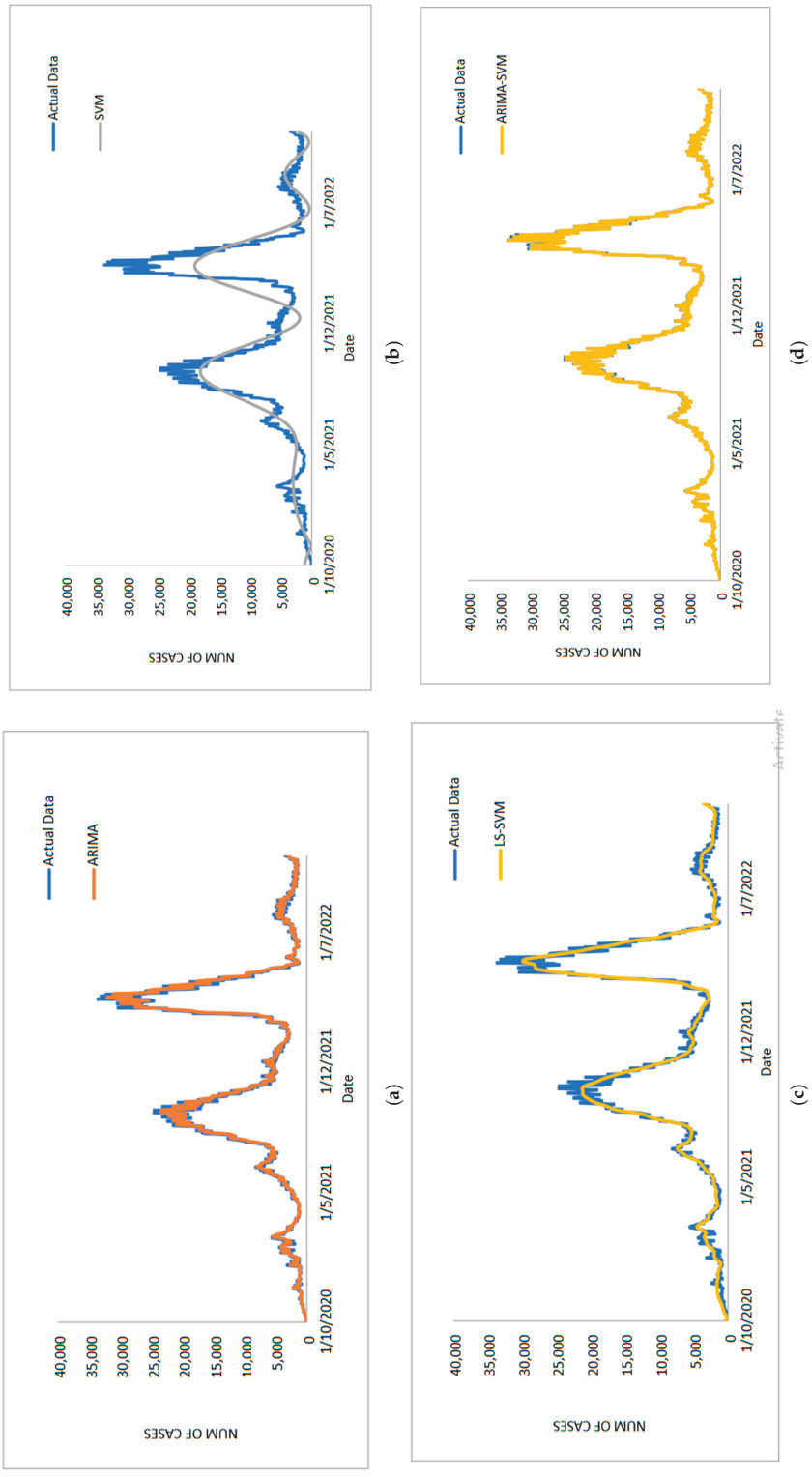
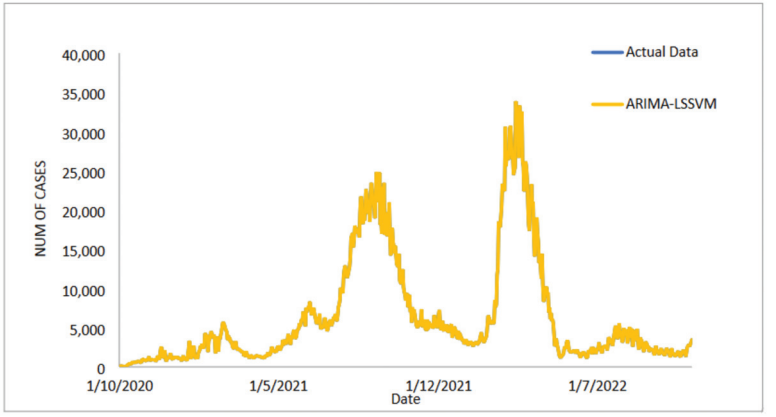


Figure 11. *Cont.*



(**e**)

Figure 11. Results obtained from the proposed model for daily new recovered COVID-19 cases dataset: (**a**) actual data vs. ARIMA model, (**b**) actual data vs. LSSVM models, (**c**) actual data vs. SVM model, (**d**) actual data vs. ARIMA–SVM models, (**e**) actual data vs. ARIMA–LSSVM models.

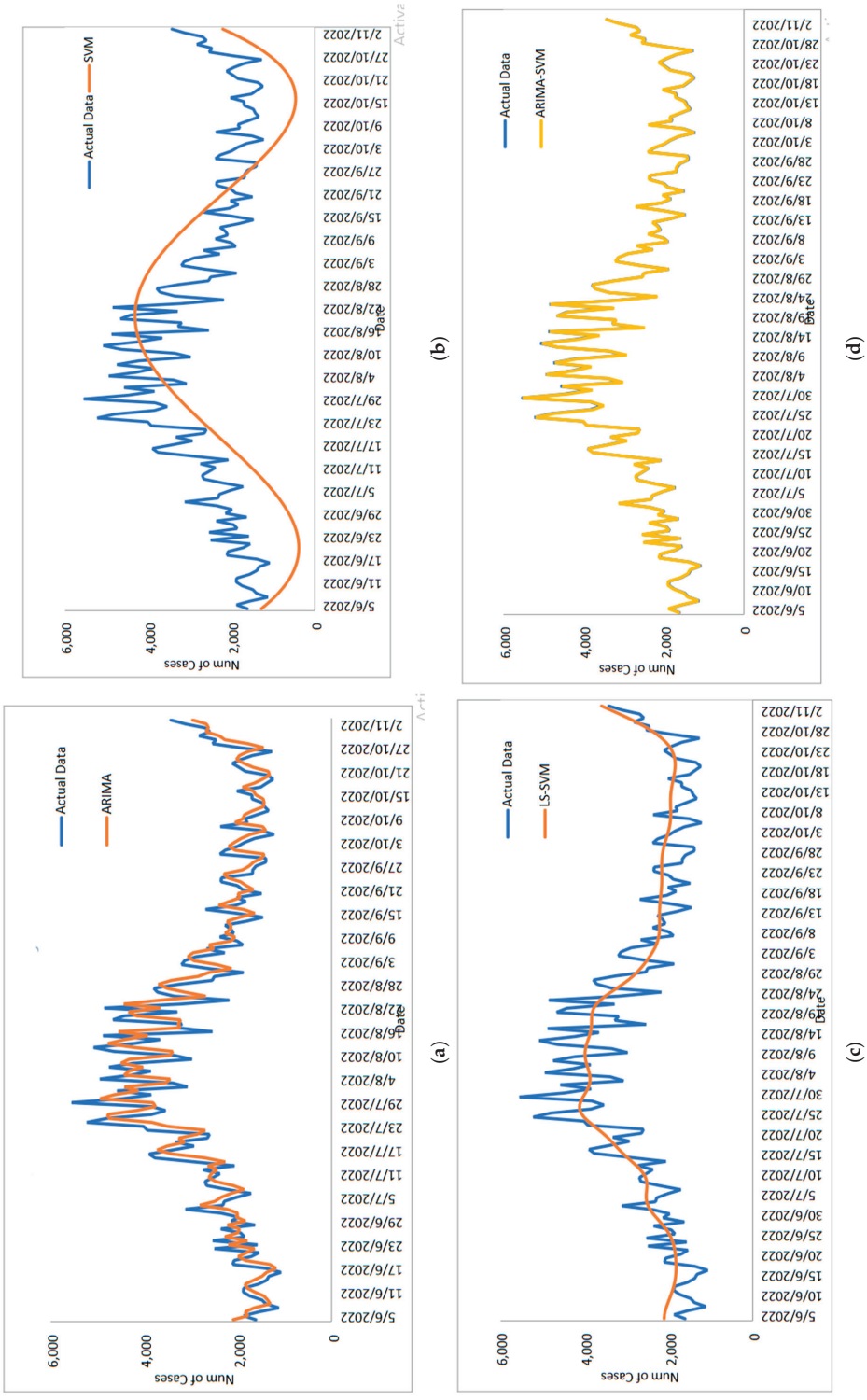
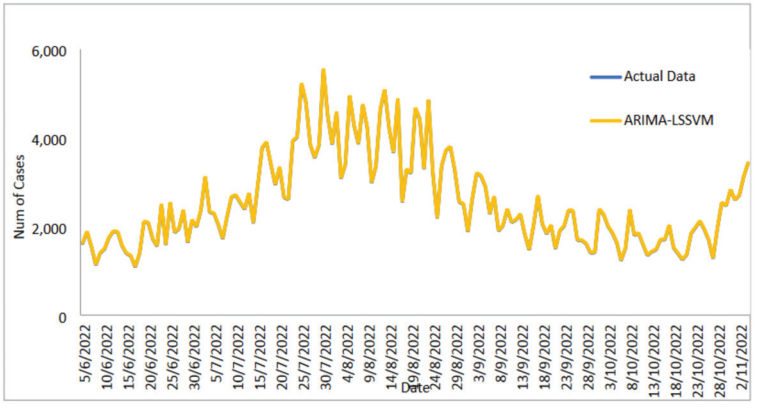


Figure 12. *Cont.*



(**e**)

Figure 12. Models' prediction of daily new recovered COVID-19 cases dataset (20% test sample): (**a**) actual data vs. ARIMA model, (**b**) actual data vs. SVM model, (**c**) actual data vs. LSSVM models, (**d**) actual data vs. ARIMA–SVM models, (**e**) actual data vs. ARIMA–LSSVM models.

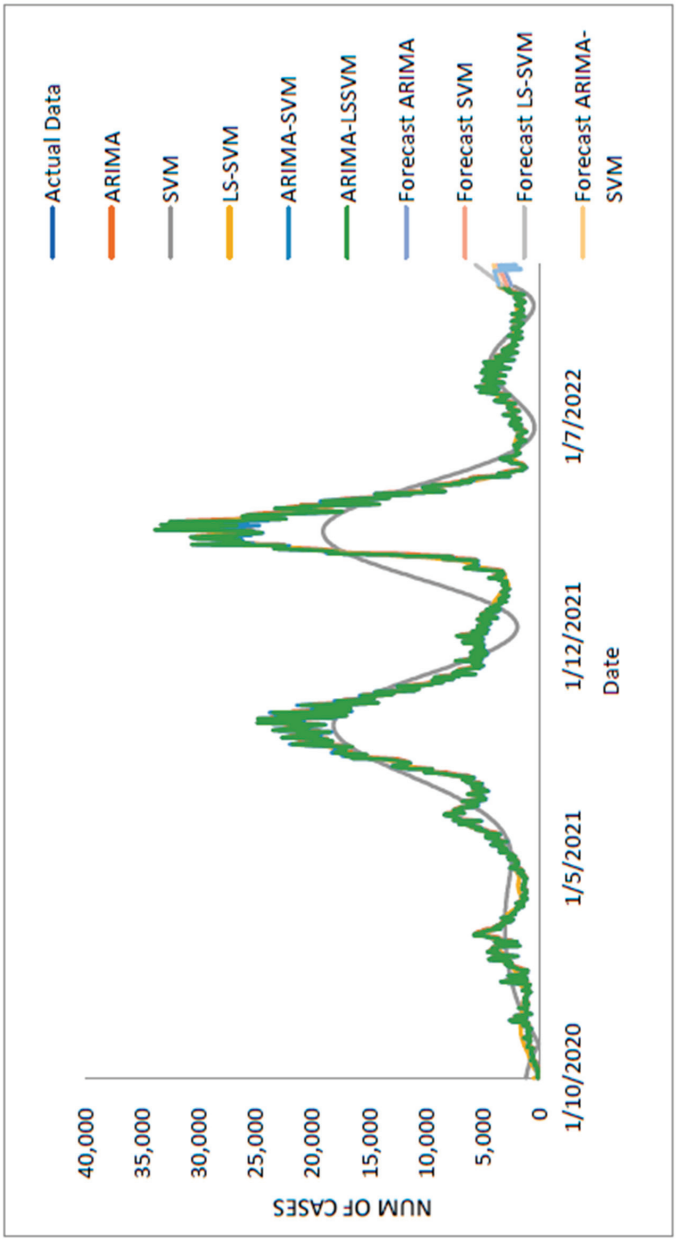


Figure 13. Actual and three weeks ahead forecasted values of ARIMA, SVM, LS–SVM, ARIMA–SVM and ARIMA–LSSVM models for daily new recovered COVID-19 cases of the 80% training and 20% testing set.

As shown in Table 10, further research was completed to determine how well the proposed models performed for the daily newly recovered COVID-19 case datasets for MSE, MAPE, RMSE, and MAE in terms of the percentage. When comparing the results of the proposed model to ARIMA, SVM, LSSVM, and ARIMA–SVM models, the results show a better improvement when looking at the percentage of improvement for statistical measurements such as MSE, MAPE, RMSE, and MAE, with results of 47.99%, 68.43%, 50.20%, and 9.42% improvement (86.02%, 91.99%, 95.21%, 78.11%; 85.43%, 91.69%, 94.57%, 76.71%; 81.06%, 88.91%, 91.32%, 70.54%). The results reported in the parentheses are the ARIMA, SVM, LSSVM, and ARIMA–SVM models. As a result, based on the findings, the proposed model has produced results that are more accurate and effective than those produced by ARIMA, SVM, LSSVM, and ARIMA–SVM models.

Table 10. Percentage improvement of the proposed models with other forecasting models (the COVID-19 cases of daily new recovered cases).

Model	MAE	MAPE	MSE	RMSE
ARIMA	86.01911863	91.98717949	95.20960596	78.11311767
SVM	85.43154944	91.68882979	94.57420558	76.70675684
LSSVM	81.06291992	88.90860692	91.32316191	70.54364347
ARIMALSSVM	47.98789051	68.43434343	50.19112901	29.42485968

4. Conclusions

In conclusion, predicting the spread of COVID-19 with accuracy and efficiency is essential but frequently challenging for decision-makers, especially the front-line workers and health care authorities. Despite what might seem to be an endless spread of COVID-19, there have been numerous efforts to develop time-series models and ongoing research to enhance forecasting model efficacy. One of the most well-liked types of hybrid models that divide time series into linear and non-linear forms is the hybrid approach. In this study, a hybrid model that combines some linear and non-linear predictions is proposed. Utilizing three well-known COVID-19 datasets—daily new positive cases, daily new death cases, and daily new recovered cases—revealed that our proposed models were demonstrated as having the highest efficiency, accuracy, and precision. In comparison to ARIMA, SVM, LSSVM, and ARIMA–SVM models, the proposed model with cross-validation check based on MSE, RMSE, MAE, and MAPE makes the most accurate predictions. In terms of performance (the proposed models compared to ARIMA, SVM, LSSVM and ARIMA–SVM models) for both the training and testing datasets, the proposed models' performance yields the smallest values of MSE, RMSE, MAE, and MAPE. This indicates that the proposed model's predicted value is more closely aligned with the observed value. Therefore, our proposed models had a higher level of precision and could be suggested for COVID-19 forecasting. It can be concluded that the proposed model may be the most efficient and effective way to increase prediction accuracy performance, especially since it is important to anticipate and stop the spread of COVID-19 cases.

5. Limitations and Future Recommendation

In this research study, an attempt was made to predict the overall number of confirmed cases, fatalities, and recoveries of COVID-19 in Malaysia. Investigating SVM performance with various kernel functions and developing the best hyperparameters for the SVM forecasting model can help to increase the forecast's accuracy in upcoming work. Since only one-step-ahead forecasting is considered in this paper, multi-step forecasts can be centralised in subsequent work. It has been demonstrated that multi-step forecasts can greatly increase the trading system's realism [41,42]. Additionally, to improve the performance of the model in terms of efficiency and accuracy of dataset prediction, hybrid approaches such as bootstrap and double bootstrap methods [16,43,44] can be considered in the hybridization of ARIMA and SVM. Given the dearth of researchers using bootstrap in daily COVID-19 forecasting cases, it is a reliable method. Numerous studies have demonstrated

that the bootstrap resampling method yields a more precise estimate [45]. Future studies should also consider (i) the clinical and behavioural aspects such as actions, cognition, and emotions and (ii) the possibility of the underreporting of cases and deaths, as well as delays in notifications, in order to avoid biased predictions, forecasts, and results.

Author Contributions: Conceptualization, A.A.K.A.H., W.I.A.W.M.N. and M.S.L.; Methodology, A.A.K.A.H., W.I.A.W.M.N., S.Z. and N.H.Z.; Software, A.A.K.A.H. and W.I.A.W.M.N.; Formal analysis, M.S.L.; Writing—original draft, W.I.A.W.M.N., M.S.L. and E.A.; Writing—review & editing, S.M.A.M., E.A., R.G. and M.T.A.; Supervision, M.S.L.; Project administration, M.S.L. and W.A.M.; Funding acquisition, M.S.L. and W.A.M. All authors have read and agreed to the published version of the manuscript.

Funding: The publication is partially sponsored by the Research Management Office, Universiti Malaysia Terengganu (UMT).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to express his gratitude to the Research Management Centre, Universiti Malaysia Terengganu (UMT) and the Research Management Centre, Universiti Malaysia Perlis (UniMAP) for partially grant of the journal publication fee as well as to the editors and the referees for careful reading and for comments which greatly improved the paper.

Conflicts of Interest: The authors declare no conflict of interest. The authors identify and declare no personal circumstances or interests that may be perceived as inappropriately influencing the representation or interpretation of the reported research results. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Abdullah, M.T.; Lola, M.S.; Hisham, A.E.; Sabreena, S.; Nor Fazila, C.M.; Idham, K.; Dennis, C.Y.T. Framework of Measures for COVID-19 Pandemic in Malaysia: Threats, Initiatives and Opportunities. *J. Sustain. Sci. Manag.* **2022**, *17*, 8–18. [CrossRef]
2. Ali, M.; Khan, D.M.; Aamir, M.; Khalil, U.; Khan, Z. Forecasting COVID-19 in Pakistan. *PLoS ONE* **2020**, *15*, e0242762. [CrossRef] [PubMed]
3. WHO. Coronavirus Disease (COVID-19) in Malaysia. 2020. Available online: https://www.who.int/malaysia/emergencies/coronavirus-disease-(COVID-19)-in-Malaysia (accessed on 23 May 2020).
4. KKM. COVID-19 Malaysia: Situasi Terkini 25 Oktober 2020. 2020. Available online: https://covid-19.moh.gov.my/terkini (accessed on 25 June 2022).
5. Gecili, E.; Ziady, A.; Szczesniak, R.D. Forecasting COVID-19 confirmed cases, deaths and recoveries: Revisiting established time series modeling through novel applications for the USA and Italy. *PLoS ONE* **2021**, *16*, e0244173. [CrossRef]
6. Awwad, F.A.; Mohamoud, M.A.; Abonazel, M.R. Estimating COVID-19 cases in Makkah region of Saudi Arabia: Space-time ARIMA modeling. *PLoS ONE* **2021**, *16*, e0250149. [CrossRef]
7. Sahai, A.K.; Rath, N.; Sood, V.; Singh, M.P. ARIMA modelling & forecasting of COVID-19 in top five affected countries. *Diabetes Metab. Syndr. Clin. Res. Rev.* **2020**, *14*, 1419–1427. [CrossRef]
8. Alzahrani, S.I.; Aljamaan, I.A.; Al-Fakih, E.A. Forecasting the Spread Of The COVID-19 Pandemic In Saudi Arabia Using ARIMA Prediction Model Under Current Public Health Interventions. *J. Infect. Public Health.* **2020**, *13*, 914–919. [CrossRef]
9. Benvenuto, D.; Giovanetti, M.; Vassallo, L.; Angeletti, S.; Ciccozzi, M. Application of the ARIMA model on the COVID-2019 epidemic dataset. *Data Brief* **2020**, *29*, 105340. [CrossRef]
10. Ceylan, Z. Estimation of COVID-19 prevalence in Italy, Spain, and France. *Sci. Total Environ.* **2020**, *729*, 138817. [CrossRef]
11. Hernandez-Matamoros, A.; Fujita, H.; Hayashi, T.; Perez-Meana, H. Forecasting of COVID19 per regions using ARIMA models and polynomial functions. *Appl. Soft Comput.* **2020**, *96*, 106610. [CrossRef]
12. Khan, F.M.; Gupta, R. ARIMA and NAR based prediction model for time series analysis of COVID-19 cases in India. *J. Saf. Sci. Resil.* **2020**, *1*, 12–18. [CrossRef]
13. Kayode, O.; Fahimah, A.; Mustapha, R.; Jacques, D. Data Analysis and Forecasting of COVID-19 Pandemic in Kuwait Based on Daily Observation and Basic Reproduction Number Dynamics. *Kuwait J. Sci. Special Issue* **2021**, 1–30. [CrossRef]
14. Rahman, M.S.; Chowdhury, A.H.; Amrin, M. Accuracy comparison of ARIMA and XGBoost forecasting models in predicting the incidence of COVID-19 in Bangladesh. *PLoS Glob. Public Health* **2022**, *2*, e0000495. [CrossRef]

15. Aisyah, W.I.W.M.N.; Muhamad Safiih, L.; Razak, Z.; Nurul Hila, Z.; Abd Aziz, K.A.H.; Elayaraja, A.; Nor Shairah, A.Z. Improved of Forecasting Sea Surface Temperature based on Hybrid ARIMA and Vector Machines Model. *Malays. J. Fundam. Appl. Sci.* **2021**, *17*, 609–620. [CrossRef]
16. Nurul Hila, Z.; Muhamad Safiih, L.; Maman Abdurachman, D.; Fadhilah, Y.; Mohd Noor Afiq, R.; Aziz, D.; Yahaya, I.; Mohd Tajuddin, A. Improvement of Time Forecasting Models using A Novel Hybridization of Bootstrap and Double Bootstrap Artificial Neural Networks. *Appl. Soft Comput. J.* **2019**, *84*, 105676. [CrossRef]
17. Lee, M.C. Using support vector machine with a hybrid feature selection method to the stock trend prediction. *J. Expert Syst. Appl.* **2009**, *36*, 10896–10904. [CrossRef]
18. Vapnik, V.N. *The Nature of Statistical Learning Theory*, 1st ed.; Springer: New York, NY, USA, 1995.
19. Sudheer, C.; Maheswaran, R.; Panigrahi, B.K.; Mathur, S. A hybrid SVM-PSO model for forecasting monthly streamflow. *Neural Comput. Appl.* **2013**, *24*, 1381–1389. [CrossRef]
20. Chakraborty, T.; Chakraborty, A.K.; Biswas, M.; Banerjee, S.; Bhattacharya, S. Unemployment Rate Forecasting: A Hybrid Approach. *Comput. Econ.* **2020**, *57*, 183–201. [CrossRef]
21. Zhang, G.P. Time series forecasting using a hybrid ARIMA and Neural Network Model. *Neurocomputing* **2003**, *50*, 159–175. [CrossRef]
22. Terui, N.; Van Dijk, H. Combined forecasts from linear and nonlinear time series models. *Int. J. Forecast.* **2002**, *18*, 421–438. [CrossRef]
23. Wang, X.; Meng, M. A Hybrid Neural Network and ARIMA Model for Energy Consumption Forecasting. *J. Comput.* **2012**, *7*, 1184–1190. [CrossRef]
24. Muhamad Safiih, L.; Nurul Hila, Z.; Mohd Tajuddin, A.; Vigneswary, P.; Mohd Noor Afiq, R.; Razak, Z.; Suffian, I.; Khalili, I. Improving the Performance of ANN-ARIMA Models for Predicting Water Quality in The Offshore Area of Kuala Terengganu, Terengganu, Malaysia. *J. Sustain. Sci. Manag.* **2018**, *13*, 27–37.
25. Pai, P.F.; Lin, C.-S. A hybrid ARIMA and Support Vector Machines Model in Stock Price Forecasting. *Int. J. Manag. Sci.* **2005**, *3*, 497–505. [CrossRef]
26. Lee, N.-U.; Shim, J.-S.; Ju, Y.-W.; Park, S.-C. Design and Implementation of the SARIMA–SVM time series analysis algorithm for the improvement of atmospheric environment forecast accuracy. *Soft Comput.* **2017**, *22*, 4275–4281. [CrossRef]
27. Hao, Y.; Xu, T.; Hu, H.; Wang, P.; Bai, Y. Prediction and analysis of Corona Virus Disease 2019. *PLoS ONE* **2020**, *15*, e0239960. [CrossRef] [PubMed]
28. Roy, S.; Ghosh, P. Factors affecting COVID-19 infected and death rates inform lockdown- related policymaking. *PLoS ONE* **2020**, *15*, e0241165. [CrossRef] [PubMed]
29. Mahdavi, M.; Choubdar, H.; Zabeh, E.; Rieder, M.; Safavi-Naeini, S.; Jobbagy, Z.; Ghorbani, A.; Abedini, A.; Kiani, A.; Khanlarzadeh, V.; et al. A machine learning based exploration of COVID-19 mortality risk. *PLoS ONE* **2021**, *16*, e0252384. [CrossRef]
30. Singhal, T. A Review of Coronavirus Disease-2019 (COVID-19). *Indian J. Pediatr.* **2020**, *87*, 281–286. [CrossRef]
31. Qu, Z.; Li, Y.; Jiang, X.; Niu, C. An innovative ensemble model based on multiple neural networks and a novel heuristic optimization algorithm for COVID-19 forecasting. *Expert Syst. Appl.* **2023**, *212*, 118746. [CrossRef]
32. Zivkovic, M.; Bacanin, N.; KVenkatachalam Nayyar, A.; Djordjevic, A.; Strumberger, I.; Al-Turjman, F. COVID-19 cases prediction by using hybrid machine learning and beetle antennae search approach. *Sustain. Cities Soc.* **2021**, *66*, 102669. [CrossRef]
33. Melin, P.; Sánchez, D.; Castro, J.R.; Castillo, O. Design of Type-3 Fuzzy Systems and Ensemble Neural Networks for COVID-19 Time Series Prediction Using a Firefly Algorithm. *Axioms* **2022**, *11*, 410. [CrossRef]
34. Sarah, M. The Future of Pandemics. News-Medical. Available online: https://www.news-medical.net/health/The-Future-of-Pandemics.aspx (accessed on 17 January 2022).
35. Suykens, J.A.K.; Vandewalle, J. Least Squares Support Vector Machine Classifiers. *Neural Process. Lett.* **1999**, *9*, 293–300. [CrossRef]
36. Naeem, M.; Yu, J.; Aamir, M.; Khan, S.A.; Adeleye, O.; Khan, Z. Comparative analysis of machine learning approaches to analyse and predict the COVID-19 outbreak. *Peer J. Comput. Sci.* **2021**, *17*, e746. [CrossRef] [PubMed]
37. Qiang, X.; Aamir, M.; Naeem, M.; Ali, S.; Aslam, A.; Shao, Z. Analysis and Forecasting COVID-19 Outbreak in Pakistan Using Decomposition and Ensemble Model. *Comput. Mater. Contin.* **2021**, *68*, 842–856. [CrossRef]
38. Adhikari, S.P.; Meng, S.; Wu, Y.-U.; Mao, Y.-P.; Ye, R.-X.; Wang, Q.-Z.; Sun, C.; Sylvia, S.; Rozelle, S.; Raat, H.; et al. Epidemiology, causes, clinical manifestation and diagnosis, prevention and control of coronavirus disease (COVID-19) during the early outbreak period: A scoping review. *Infect. Dis. Poverty* **2020**, *9*, 29. [CrossRef] [PubMed]
39. Ahmadini, A.A.H.; Naeem, M.; Aamir, M.; Dewan, R.; Alshqaq, S.S.A.; Mashwani, W.K. Analysis and Forecast of the Number of Deaths, Recovered Cases, and Confirmed Cases from COVID-19 for the Top Four Affected Countries Using Kalman Filter. *Front. Phys.* **2021**, *9*, 629320. [CrossRef]
40. Alessa, A.A.; Alotaibie, T.M.; Elmoez, Z.; Alhamad, H.E. Impact of COVID-19 on Entrepreneurship and Consumer Behaviour: A Case Study in Saudi Arabia. *J. Asian Financ. Econ. Bus.* **2021**, *8*, 201–210. [CrossRef]
41. Huck, N. Pairs trading and outranking: The multi-step-ahead forecasting case. *Eur. J. Oper. Res.* **2010**, *207*, 1702–1716. [CrossRef]
42. Nisbet, R.; Elder JMiner, G. Chapter 11—Model Evaluation and Enhancement. In *Handbook of Statistical Analysis and Data Mining Applications*; Academic Press: Cambridge, MA, USA, 2018; pp. 215–233. [CrossRef]
43. Nurul Hila, Z.; Muhamad Safiih, L. The Performance of BB-MCEWMA Model: Case Study on Sukuk Rantau Abang Capital Berhad, Malaysia. *Int. J. Appl. Bus. Econ. Res.* **2016**, *14*, 63–77.

44. Nurul Hila, Z.; Muhamad Safiih, L.; Nur Shazrahanim, K. Modelling Moving Centreline Exponentially Weighted Moving Average (MCEWMA) with bootstrap approach: Case study on sukuk musyarakah of Rantau Abang Capital Berhad, Malaysia. *Int. J. Appl. Bus. Econ. Res.* **2016**, *14*, 621–638.
45. Muhamad Safiih, L.; Nurul Hila, Z.; Mohd Noor Afiq, R.; Hizir, S. Double Bootstrap Control Chart for Monitoring SUKUK Volatility at Bursa Malaysia. *J. Teknol.* **2017**, *79*, 149–157. [CrossRef]

MDPI
St. Alban-Anlage 66
4052 Basel
Switzerland
Tel. +41 61 683 77 34
Fax +41 61 302 89 18
www.mdpi.com

Diagnostics Editorial Office
E-mail: diagnostics@mdpi.com
www.mdpi.com/journal/diagnostics

Academic Open
Access Publishing

www.mdpi.com

ISBN 978-3-0365-8069-2