



Article

A Lightweight Neural Network Model for Disease Risk Prediction in Edge Intelligent Computing Architecture

Feng Zhou ¹, Shijing Hu ^{1,*}, Xin Du ¹, Xiaoli Wan ² and Jie Wu ¹

¹ School of Computer Science, Fudan University, Shanghai 200438, China; 19110240047@fudan.edu.cn (F.Z.); xdu20@fudan.edu.cn (X.D.); jwu@fudan.edu.cn (J.W.)

² Information Center, Zhejiang International Business Group, Hangzhou 310000, China; wanxl@zibchina.com

* Correspondence: sjhu21@m.fudan.edu.cn

Abstract: In the current field of disease risk prediction research, there are many methods of using servers for centralized computing to train and infer prediction models. However, this centralized computing method increases storage space, the load on network bandwidth, and the computing pressure on the central server. In this article, we design an image preprocessing method and propose a lightweight neural network model called Linge (Lightweight Neural Network Models for the Edge). We propose a distributed intelligent edge computing technology based on the federated learning algorithm for disease risk prediction. The intelligent edge computing method we proposed for disease risk prediction directly performs prediction model training and inference at the edge without increasing storage space. It also reduces the load on network bandwidth and reduces the computing pressure on the server. The lightweight neural network model we designed has only 7.63 MB of parameters and only takes up 155.28 MB of memory. In the experiment with the Linge model compared with the EfficientNetV2 model, the accuracy and precision increased by 2%, the recall rate increased by 1%, the specificity increased by 4%, the F1 score increased by 3%, and the AUC (Area Under the Curve) value increased by 2%.

Keywords: federated learning; edge computing; deep learning; image classification; disease risk prediction



Citation: Zhou, F.; Hu, S.; Du, X.; Wan, X.; Wu, J. A Lightweight Neural Network Model for Disease Risk Prediction in Edge Intelligent Computing Architecture. *Future Internet* **2024**, *16*, 75. <https://doi.org/10.3390/fi16030075>

Academic Editor: Symeon Papavassiliou

Received: 22 January 2024

Revised: 13 February 2024

Accepted: 17 February 2024

Published: 26 February 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Current research on disease risk prediction mainly uses physiological indicators and risk factors to predict disease risk [1], images to predict disease risk [2], and audio to predict disease risk [3]. These studies have achieved good results. In these studies, the risk of diabetes was predicted with 99.8% accuracy, the risk of Parkinson's disease was predicted with 98% accuracy, and the risk of laryngeal air cyst was predicted with 98.5% accuracy. Some researchers use the prediction of mean arterial blood pressure in patients with sepsis to assist in treating septic shock [4].

The current application scenarios of disease risk prediction models mainly include hospitals, health management centers, insurance institutions, and elderly care institutions. Hospitals use disease risk prediction models to assist doctors in providing medical services to patients. The health management center uses disease risk prediction models to predict health risks for users and develops health intervention plans based on health analysis reports. Insurance institutions use disease risk prediction models to assist staff in making business risk judgments. Elderly care institutions use disease risk prediction models to periodically provide disease risk predictions to the elderly, improving their health status and reducing the pressure on public health services.

Although the application scenarios of disease risk prediction models are diverse, according to our research, these application scenarios also have standard rules. When designing and implementing disease risk prediction models using centralized computing methods, these application scenarios must go through model design, sample processing,

model training, and optimization. In the sample processing stage, when different institutions or enterprises use centralized computing methods, they will summarize the data of each branch unit. This method increases the overhead of data storage space and brings hidden dangers to data transmission security and integrity. In the model training and optimization phase, different institutions or enterprises use central server calculations when using centralized computing methods. This method increases the burden on the central server and wastes edge server resources. The usage of our proposed distributed intelligent edge computing technology for disease risk prediction is shown in Figure 1.

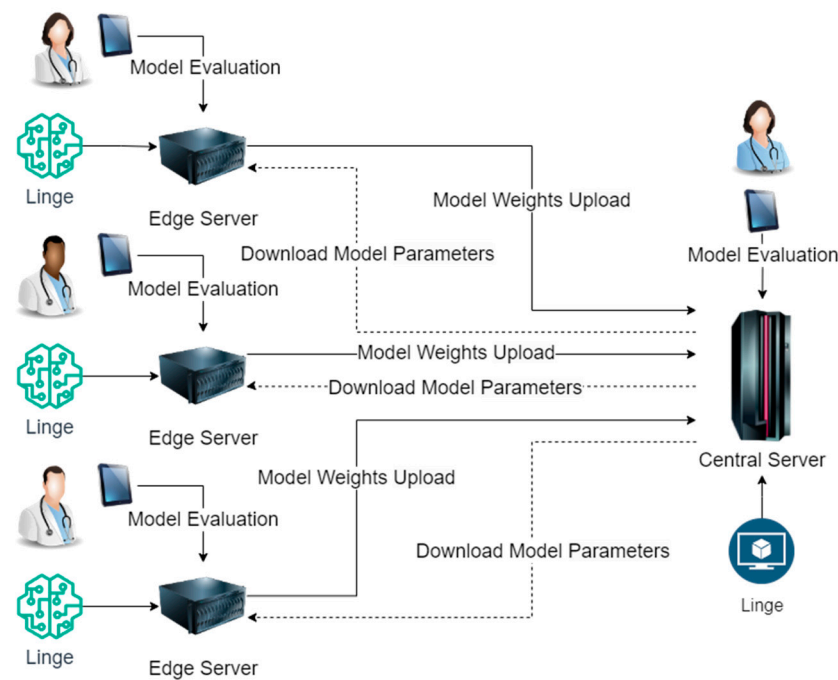


Figure 1. Implementation Process of Distributed Intelligent Edge Computing Approach for Disease Risk Prediction.

As shown in Figure 1, the lightweight neural network model Linge (Lightweight Neural Network Models for the Edge) we designed will be deployed on the enterprise or institution's central server and each branch unit's edge servers. The prediction model's training and inference are completed on each branch unit's edge servers. The prediction model on the edge server of each branch unit will upload the model weights output to the central server after training and download the latest model parameters from the central server. During the inference process of the prediction model, disease risk prediction experts evaluate and correct the inference results of the model on the central server and each edge server. Using this distributed intelligent edge computing method to train disease risk prediction models does not increase data storage space, completely eliminates hidden dangers in data transmission security and data transmission integrity, and does not increase computing pressure on the central server.

The research goal of this article is to achieve effective prediction of disease risk in diverse disease risk prediction application scenarios based on federated learning algorithms, edge computing, the MRI (Magnetic Resonance Imaging) image preprocessing method designed in this article, and lightweight neural network models. In this article's research on effective disease risk prediction, the main work carried out is as follows:

- To further improve the quality of the data set, reduce the cost of feature extraction by the model, and improve the model's accuracy, we designed an MRI image preprocessing method.
- For the configuration attributes of the edge server, we designed a new lightweight neural network model based on the lightweight attention mechanism.

- For diverse application scenarios, we propose a distributed intelligent edge computing technology for disease risk prediction based on federated learning algorithms, edge computing, the MRI image preprocessing method designed in this article, and lightweight neural network models.

The introductory section of this article describes the background of our research. The related work section describes the current research status. The Preliminaries section describes related techniques. The model design and implementation section describes the designed image preprocessing method, lightweight neural network model, and distributed intelligent edge computing technology for disease risk prediction. The experimental results analysis and conclusion sections describe the experiments during the research process of this article, summarize the research results, and discuss future research.

2. Related Works

Many current disease risk prediction studies are based on federated learning or edge computing. For example, to achieve accurate heart disease prediction on the medical Internet of Things platform, Y. Pan et al. proposed a multilayer perceptron model based on a convolutional neural network to help doctors effectively diagnose heart disease patients on the cloud platform [5]. This model achieved an accuracy of 94.9% in experiments. To perform training tasks on wearable devices [6], Yeting Guo et al. proposed a federated edge learning system for efficient privacy-preserving mobile healthcare based on federated learning. S. Hakak et al. proposed a conceptual framework for leveraging edge computing to support healthcare analytics based on user-generated data [7]. This framework can be extended to develop distributed disease management systems based on personal health data.

To solve the problems of network congestion and low response speed that occur when implementing clinical decision support systems using traditional methods [8], Z. Xue et al. proposed a technology that integrates mobile edge computing and software-defined networks. D. Gupta et al. proposed an anomaly detection model based on federated learning to solve the anomaly detection problem in centralized healthcare ecosystems [9], which often suffer from severe response time delays and high-performance overhead. To reduce the cost of data transmission to the cloud in the healthcare Internet of Things [10], W. Y. B. Lim et al. proposed a dynamic contract design based on federated learning and edge computing architecture for innovative medical applications. Since the urban digital twin system relies on long-term collected data to make appropriate decisions to solve the limitations when major infectious disease emergencies occur [11], J. Pang et al. proposed a framework that integrates the urban digital twin system with federated learning. V. Gomathy et al., based on edge computing methods and linear regression [12], studied polluted air and mortality caused by COVID-19 (Corona Virus Disease 2019) and concluded that in areas with air pollution, the mortality caused by COVID-19 is 77% higher. D. Y. Zhang et al. proposed a new federated learning framework to solve the class imbalance problem in abnormal health detection [13]. This framework achieves an F1 score of 0.816 in driver drowsiness detection applications.

Q. Wu et al. proposed a new cloud edge-based federated learning framework for home health monitoring [14]. This framework learns a global model shared in the cloud from multiple homes at the network's edge and achieves data privacy protection by saving user data locally. This framework achieved an accuracy of 95.87% in experiments. D. C. Nguyen et al. proposed a new blockchain-based framework to implement COVID-19 detection using generative adversarial networks in edge cloud computing [15]. This framework implements the joint design of federated learning and GAN (Generative Adversarial Networks) in a distributed medical network with edge cloud computing. This framework achieved an accuracy of 97.5% in experiments.

To solve the problem of high latency in the healthcare system due to its reliance on central servers [16], V. Hayyolalam et al. used edge computing to move computing and storage resources closer to end users. This method utilizes a metaheuristic-based feature selection method of the Black Widow Optimization (BWO) algorithm to detect heart dis-

ease in patients. Experimental results show that they achieved an accuracy of 90.11%. To improve the confidence of the prediction model [17], Linardos et al. proposed a simulated federated learning research method on cardiovascular diseases based on CNN (Convolutional Neural Networks). To realize human motion recognition based on federated learning through wearable devices in intelligent medical systems [18], Arikumar KS et al. proposed a federated learning-based human motion recognition method based on bidirectional long short-term memory (BiLSTM). The accuracy of this method in experiments reached 99.67%. To protect the data privacy of healthcare applications that rely on the Internet of Things [19], H. Elayan et al. proposed a deep federated learning framework for healthcare data monitoring and analysis using IoT devices. This framework achieved an accuracy of 84.8% and an AUC (Area Under the Curve) of 97% in experiments on detecting skin diseases.

To solve the privacy and security issues during interactions caused by potential patient data leakage in the healthcare Internet of Things [20], Z. Lian et al. proposed a decentralized, efficient, and privacy-enhanced federated edge learning system based on convolutional neural networks. This system achieved an accuracy of 87% in experiments on the skin cancer data set. B. T. H. Dang et al. proposed a collaboration framework [21]. This framework is used to train convolutional neural network-based heart disease prediction models. This framework uses federated learning to implement model training using distributed data stored individually on multiple machines. This framework uses the MIT-BIH arrhythmia data set to train the model. In experiments, this framework achieved an accuracy of 98.92%, a recall of 97.41%, a precision of 99.23%, and an F1 score of 98.02%.

To solve the security and privacy issues of the medical Internet of Things under edge intelligent computing [22], R. Wang et al. proposed a privacy protection scheme for federated learning under edge computing. Xiaomin Ouyang et al. proposed an end-to-end system integrating multimodal sensors and federated learning algorithms to achieve digital biomarker detection of multidimensional Alzheimer's disease in natural living environments [23]. The system achieved an accuracy of 95% in experiments detecting digital biomarkers and an average accuracy of 87.5% in experiments identifying Alzheimer's disease. L. Zhang et al. proposed a privacy-preserving federated learning method based on homomorphic encryption and deep neural network (DNN) for the Internet of Things healthcare system [24]. This method achieved an accuracy of 76.9% in experiments using the HAM10000 data set.

As can be seen from the above description and Table 1, current research based on machine learning, federated learning, and edge computing has achieved good results. For example, the accuracy of predicting the risk of cardiovascular disease reaches 99.67%, the accuracy of predicting the risk of skin cancer reaches 87%, and the accuracy of predicting the risk of heart disease reaches 98.92%. However, in the above studies, the neural network model parameters used are too large when predicting disease risk. In the application scenarios of federated learning and edge computing, they are limited by hardware resource requirements and are challenging to promote widely. The lightweight neural network model designed in this article has only 7.63 MB of parameters and only occupies 155.28 MB of memory. In the experiment, the accuracy of stroke risk prediction was 96%, the precision rate was 95%, the recall rate was 93%, the specificity was 95%, the F1 score was 98%, and the AUC was 97%. This experimental result shows that the distributed intelligent edge computing technology proposed in this article for disease risk prediction can provide practical support for disease risk prediction.

Table 1. Related Research Statistics.

Researcher	Research Objectives	Basic Algorithm	Accuracy
Q. Wu et al. [14]	Health Monitoring	CNN	95.87%
D. C. Nguyen et al. [15]	COVID-19	GAN	97.5%
Linardos et al. [17]	Cardiovascular Diseases	BiLSTM	99.67%
Z. Lian et al. [20]	Skin Cancer	CNN	87%
B. T. H. Dang et al. [21]	Heart Disease	CNN	98.92%

3. Preliminaries

The data sets used in the experiments of this article are the publicly available “Acute Ischemic Stroke MRI” data set and subject data, which are all image data. Therefore, when designing the lightweight neural network model structure, this article refers to the lightweight attention mechanism and the structure of the EfficientNetV2 network model.

3.1. Channel Attention Mechanism

SENET is the champion model of ImageNet 2017 [25]. SENET is used in the core MBConv module and Fused-MBConv module of the EfficientNetV2 network model to improve network performance. The SENET module structure is shown in Figure 2.

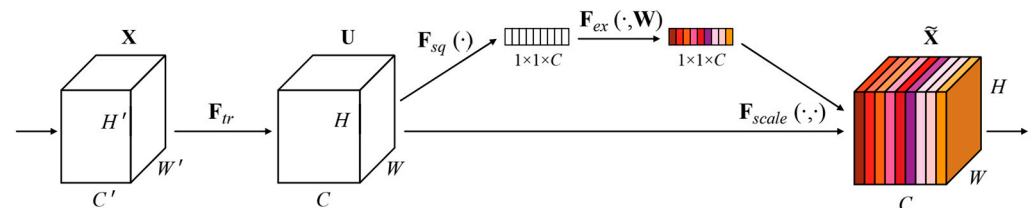


Figure 2. The SENET module structure [25].

In Figure 2, given an input X with a feature channel number of C' , a feature with a feature channel number of C is obtained after a series of convolution and other transformations [25].

ECA-Net (Efficient Channel Attention Network) is an attention model used for computer vision tasks [26], designed to enhance the ability of neural networks to model image features. The overall structure of the ECA-Net module is shown in Figure 3 below.

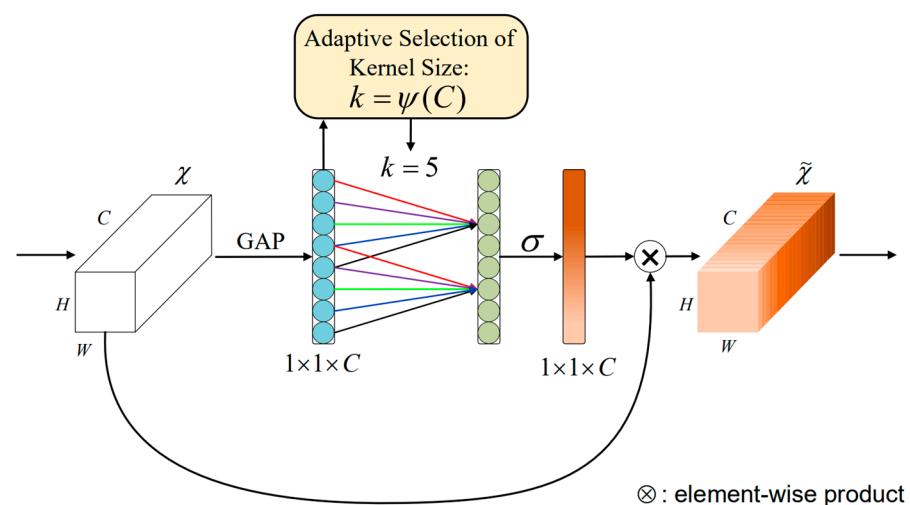


Figure 3. The ECA-Net module structure [26].

Figure 3 gives the aggregated features obtained through Global Average Pooling GAP (Global Average Pooling) [26]. ECA-Net generates channel weights by performing fast one-dimensional convolutions of size k , with a convolution kernel size k representing the coverage of local cross-channel interactions, where k is adaptively determined by the mapping of the channel dimension C . After global average pooling without reducing the dimensionality, ECA-Net captures local cross-channel interactive information by using each channel and its adjacent channels. This method ensures model efficiency and computational effect.

3.2. Lightweight Neural Network

Berkeley and Stanford researchers proposed that the parameters of the SqueezeNet network model are only 1/50 of Alexnet [27], but it achieves similar effects to Alexnet

on ImageNet. Due to too few parameters, SqueezeNet is less effective in expressing complex problems. Google researchers proposed the MobileNet network model using depthwise separable convolutions [28]. MobileNetV2 optimizes the model structure based on MobileNet, which not only improves the performance of the network model but also reduces the calculation amount of the model. MNasNet integrates the inverted residual block designed based on MobileNetV2 as a building block into NAS [29]. Since MNasNet performs very well in block search but needs to be more comprehensive in the search of each layer width, MobileNetV3 uses MNasNet to search blocks [30]. For a series of models using the same building block, refer to the coefficients in MobileNet and MobileNetV2 and quickly obtain models of different sizes by directly increasing the width, depth, and resolution [31]. However, balancing the relationship between width, depth, and resolution has become vital.

To balance the relationship between width, depth, and resolution, the EfficientNet network model came into being [32]. However, this network model consumes much video memory when the input image is large. In addition, the EfficientNet network model has a significant overhead in reading and writing data. To further improve the EfficientNet network model, the EfficientNetV2 network model was born, which combines training-aware NAS and scaling [33].

4. Model Design and Implementation

This section introduces the MRI image sample preprocessing method, lightweight neural network model, distributed intelligent edge computing technology implementation, and model algorithm for disease risk prediction designed in this article.

4.1. MRI Image Sample Preprocessing Method

MRI (Magnetic Resonance Imaging) images provide doctors with clear, high-resolution images of the internal tissues and organs of the body. These images can reveal abnormal structures and help doctors accurately diagnose and develop treatment plans. To further improve the accuracy and reliability of the model, we designed a method to preprocess MRI image samples. The specific process is summarized as shown in Figure 4.

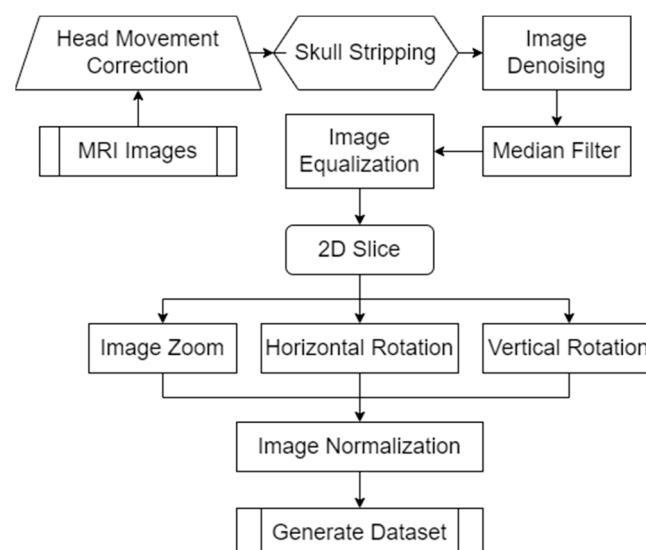


Figure 4. MRI image sample preprocessing method flow.

Since each MRI image is not entirely aligned, the MRI image sample preprocessing method we designed is shown in Figure 4. The image is first corrected for head motion. Since MRI images contain some non-brain structures (such as skulls), we delete non-brain structures in the images to avoid increasing the computational load and improving model efficiency. To further improve the training efficiency of the model, we denoise the image and

use median filtering to denoise the spatial domain of the image. To extract more detailed features in the image, we equalize the image. Since MRI images are three-dimensional data, the model designed in this article inputs two-dimensional images. Therefore, in the MRI image sample preprocessing method we designed, only the coronal, sagittal, and transverse planes in the MRI images are extracted. To prevent the model from overfitting and improve the generalization ability of the model, we scale and rotate the MRI images. Finally, the MRI image sample preprocessing method we designed normalizes the data by traversing each pixel in the MRI image matrix. The normalization operation formula is summarized in Formula (1) during the processing process.

$$\hat{I} = \log_{10}(I) / \log_{10}(\max(I)) \quad (1)$$

I in Formula (1) represents the original image, $\hat{I}$ represents the normalized image, $\log_{10}(I)$ represents the logarithmic transformation function with base 10, and $\max(I)$ represents the maximum value in the image data.

4.2. Lightweight Neural Network Model

To improve the feature extraction capability and robustness of the model and focus more on the information of practical feature areas in the image, this article designed the EMBCConv module and EFused-MBCConv module concerning the EfficientNetV2 model structure. The structures of the EMBCConv module and EFused-MBCConv module are shown in Figure 5.

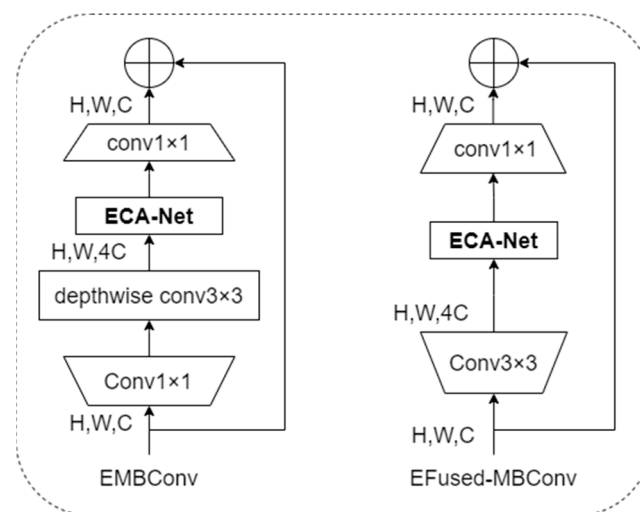


Figure 5. Structure of EMBCConv module and EFused-MBCConv module.

As shown in Figure 5, this article integrates the lightweight attention mechanism into the EMBCConv module and EFused-MBCConv module in the lightweight neural network model structure. The EfficientNetv2-S network model is the lightest model among the EfficientNetv2 network models. However, problems like large computational workload, long training time, and giant model space during training still need to be solved. Considering the application scenarios of distributed intelligent edge computing for disease risk prediction to allow model training and inference to be completed at the edge, this article refers to the backbone network structure of the EfficientNetv2-S network model and designs the Linge model. Because EMB-Conv has a smaller expansion ratio than EFused-MBCConv, more minor expansion has less memory access overhead. Therefore, in the structure of the lightweight neural network model Linge designed in this article, EFused-MBCConv is used in the Stage1 and Stage2 stages, and the EMBCConv module is used in the Stage3, Stage4, Stage5, and Stage6 stages. This adjustment increases the size of the model's receptive field,

reduces the depth and complexity of the model, and further improves the lightweight of the model. The structure of the new lightweight neural network model is shown in Table 2.

Table 2. Linge network model structure.

Stage	Operator	Stride	#Channels	#Layers
0	Conv3x3	2	24	1
1	EFused-MBConv1, k3x3	1	24	2
2	EFused-MBConv4, k3x3	2	48	4
3	EMBConv4, k3x3	2	64	4
4	EMBConv4, k3x3, ECA0.25	2	128	6
5	EMBConv6, k3x3, ECA0.25	1	160	9
6	EMBConv6, k3x3, ECA0.25	2	256	15
7	Conv1x1 & Pooling & FC	-	1280	1

Table 2, ECA0.25 indicates that the number of nodes in the first fully connected layer in the ECA module is 1/4 of the number of feature matrix channels input to the EMBConv module. The new lightweight neural network model reduces memory access overhead and improves feature extraction capabilities and model robustness.

4.3. Implementation of Distributed Intelligent Edge Computing Technology for Disease Risk Prediction

The implementation process of the distributed intelligent edge computing technology proposed in this article for disease risk prediction is summarized as shown in Figure 6.

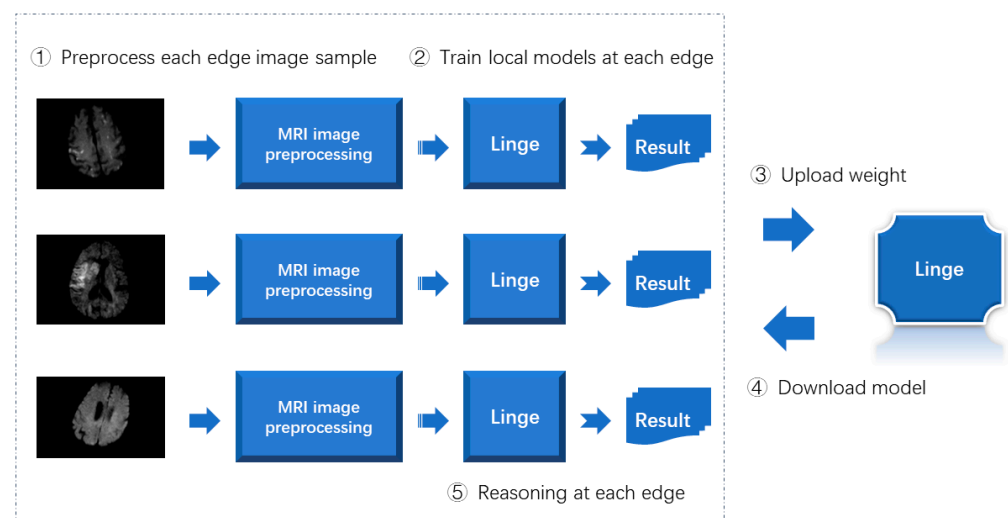


Figure 6. Implementation process of distributed intelligent edge computing technology for disease risk prediction.

As shown in Figure 6, when implementing distributed intelligent edge computing technology for disease risk prediction, the image samples at each edge are first processed using the MRI image preprocessing method designed in this article. Secondly, the pre-processed image samples train the lightweight neural network model designed in this article. Then, upload the model weights obtained by training to the central server. Then, each edge end downloads the most complete model. Finally, the inference is completed by each edge end.

The model training, verification, and testing of distributed intelligent edge computing technology for disease risk prediction uses public MRI image data sets. To verify the versatility of the distributed intelligent edge computing technology proposed in this article for disease risk prediction, this article uses the subject's MRI image data set to verify the designed prediction model. Experimental results show that the distributed

intelligent edge computing technology proposed in this article for disease risk prediction has good versatility.

4.4. Model Algorithm

The algorithm process of the distributed intelligent edge computing technology proposed in this article for disease risk prediction is summarized as follows:

In Algorithm 1, $\{I_1, I_2, \dots, I_n\}$ represents the “Acute Ischemic Stroke MRI” data set published on Kaggle. T represents the training sample data, V represents the verification sample data, S represents the test sample data, and epoch represents the number of training iterations.

Algorithm 1: Algorithms for intelligent edge computing technology

Input: $\{I_1, I_2, \dots, I_n\}$

Output: Disease Risk Prediction Model

```

1   $T, V, S \leftarrow$  use the image feature preprocessing method designed in this article to preprocess
    $\{I_1, I_2, \dots, I_n\}$ 
2  for  $n = 0$  to epoch do
3      Training a predictive model using Linge network
4      Evaluate training effectiveness
5      Model parameter optimization
6      if the evaluation indicators are qualified:
7          Output prediction model
8          break
9      else:
10          $n = n + 1$ 
11  Output the model with the highest evaluation
12  The edge side uploads model weights to the central server
13  The edge downloads the model from the central server and performs inference on the edge.
```

In Algorithm 1, each edge end uses the image preprocessing method to first preprocess the data set published on Kaggle. Then, the preprocessed samples were used to train the lightweight neural network model designed in this article. Secondly, each edge end uploads the model weight to the central server. The edge then downloads the central server model. Finally, the edge side completes the inference.

5. Experimental Results and Analysis

This section describes the sample data, evaluation metrics, experimental environment, prediction results, model performance indicator comparison, and model optimization comparison used in the experimental process of the proposed distributed intelligent edge computing technology for disease risk prediction. The federated learning framework we used in the experiment is TensorFlow Federated, and the deep learning framework is Tensorflow-GPU 2.6.0.

5.1. Sample Data

The data set used in this article is the “Acute Ischemic Stroke MRI” data set publicly available on Kaggle. The Neurology Department of Turgut Özal University Medical College Hospital collected this brain image data set. The Ethics Committee of the Faculty of Medicine of Turgut Özal University approved this study [34].

The sample data in Table 3 use MRI images of patients with acute ischemic infarction who were admitted to the hospital in 2021. This data set consists of ischemic acute infarction and standard images. To further verify the model’s generalization ability, we used subject data from a nursing home to test the model further [34].

Table 3. Related Research Statistics [34].

Diffusion MRIs	Male	Female	Total	Male Age	Female Age	Number of MRIs
Ischemic Acute	33	44	77	73.36 ± 13.52	70.39 ± 15.17	1002
Infarction Healthy	21	16	37	68.9 ± 8.72	71.43 ± 16.39	1008

5.2. Evaluation Metrics

The indicators used to evaluate the prediction model in this article are accuracy, precision, recall, specificity, F1 score, and AUC value.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (2)$$

$$Precision = \frac{TP}{TP + FP} \times 100\% \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (4)$$

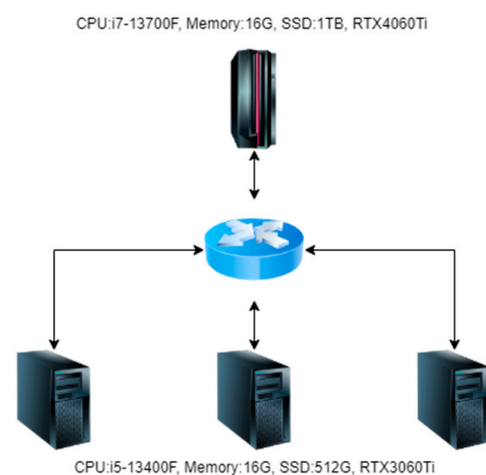
$$Specificity = \frac{TN}{TP + FN} \times 100\% \quad (5)$$

$$F1 = \frac{2TP}{2TP + FP + FN} \times 100\% \quad (6)$$

In Formulas (2)–(6), *TP* means that the correct prediction is a positive example; that is, the prediction is a positive example and the prediction result is correct [1]. In Formulas (2)–(6), *TP* means that the correct prediction is a positive example, and *TN* means that the correct prediction is a counterexample. *FP* means that the wrong prediction is a positive example. For example, *FN* indicates that the wrong prediction is a counterexample [2,3].

5.3. Experimental Environment

When conducting experiments in this article, four machines were used, one of which was used as the central server and three as edge servers. The topology diagram is shown in Figure 7.

**Figure 7.** Experimental environment topology diagram.

As shown in Figure 7, in the hardware configuration of the central server during the experiment, the CPU is i7-13700F, the memory is 32 GB, the SSD is 3 TB, and the graphics card is RTX4060Ti. In the hardware configuration of the edge server during the experiment, the CPU is i5-13400F, the memory is 16 GB, the SSD is 1 TB, and the graphics card is

RTX3060Ti. The operating systems of both the central and edge servers use the Ubuntu 20.04 LTS Server.

5.4. Prediction Results

In image classification prediction research, standard lightweight neural network models include LeViT, EfficientNetV2, MobileViTv2, EdgeViTs, EdgeNeXt, and AFFNet. Lightweight neural network models such as LeViT, EfficientNetV2, MobileViTv2, EdgeViTs, EdgeNeXt, and AFFNet are relatively common lightweight neural network models that achieve SOTA performance. Therefore, in order to understand the experimental effect of the lightweight neural network model proposed in this article, we conducted comparative experiments with network models such as LeViT, EfficientNetV2, MobileViTv2, EdgeViTs, EdgeNeXt, and AFFNet. The “Acute Ischemic Stroke MRI” data set published on Kaggle is used in experiments in papers published by researchers such as B. Tasci. Therefore, we chose to use this data set in the experiment, divided the training set, verification set, and test set according to the ratio of 8:1:1, and applied it to the comparative experiments of all models. When conducting experiments, we used all Adam optimizers; the batch_size settings were all 16, and the learning_rate settings were all 0.0001. The results of the experiment are shown in Table 4.

Table 4. Comparison of experimental results of various network models on public data sets.

Basic Algorithm	Accuracy	Precision	Recall	Specificity	F1	AUC
LeViT [35]	0.84	0.83	0.81	0.79	0.85	0.92
EfficientNetV2 [33]	0.94	0.93	0.92	0.91	0.95	0.95
MobileViTv2 [36]	0.86	0.84	0.85	0.83	0.87	0.93
EdgeViTs [37]	0.87	0.85	0.88	0.86	0.91	0.92
EdgeNeXt [38]	0.89	0.87	0.90	0.88	0.93	0.91
AFFNet [39]	0.93	0.92	0.91	0.89	0.94	0.96
Linge	0.96	0.95	0.93	0.95	0.98	0.97

Table 4 shows the experimental results on the public data set “Acute Ischemic Stroke MRI” introduced in this article. As seen from Table 4, in the experiment using public data sets to compare various network models, the EdgeNeXt and AFFNet network models achieved better experimental results. The lightweight neural network model proposed in this article has the best experimental results. When conducting comparative experiments, we accepted 350 MRI image data from two categories (with and without stroke risk, 175 images, respectively) provided by a medical institution. We organized them into a subject data set. This subject data set is used as a test data set for further testing of the model. To gain an in-depth understanding of the comparative experimental effects, we further tested each network model using the subject data set. The experimental results are shown in Table 5.

Table 5. Comparison of experimental results of various network models on the subject data set.

Basic Algorithm	Accuracy	Precision	Recall	Specificity	F1	AUC
LeViT [35]	0.83	0.82	0.77	0.78	0.81	0.93
EfficientNetV2 [33]	0.95	0.94	0.91	0.93	0.92	0.94
MobileViTv2 [36]	0.85	0.87	0.80	0.81	0.83	0.94
EdgeViTs [37]	0.88	0.89	0.85	0.83	0.84	0.93
EdgeNeXt [38]	0.90	0.91	0.87	0.89	0.86	0.92
AFFNet [39]	0.92	0.93	0.89	0.91	0.90	0.95
Linge	0.97	0.96	0.94	0.96	0.97	0.95

As shown in Table 5, in the experiment using the subject data set to compare various network models, the EdgeNeXt and AFFNet network models achieved better experimen-

tal results. The lightweight neural network model proposed in this article has the best experimental results.

This experiment shows that the lightweight neural network model proposed in this article can effectively support our proposed intelligent edge computing method for disease risk prediction.

5.5. Model Performance Indicator Comparison

To gain an in-depth understanding of the various performance effects of the lightweight neural network we proposed, this article conducted comparative experiments using lightweight neural network models LeViT, EfficientNetV2, MobileViTv2, EdgeViTs, EdgeNeXt, and AFFNet. The experimental comparison results are shown in Table 6.

Table 6. Comparison of model performance indicators with experimental results.

Basic Algorithm	Parameter Quantity/MB	Number of Operations/G	Memory Capacity Occupied/MB
LeViT [35]	11.93	11.97	192.97
EfficientNetV2 [33]	11.78	11.39	188.85
MobileViTv2 [36]	11.25	11.26	186.78
EdgeViTs [37]	10.27	10.86	185.63
EdgeNeXt [38]	9.55	9.98	170.35
AFFNet [39]	8.98	6.59	160.79
Linge	7.63	5.91	155.28

Table 6 shows that the proposed lightweight neural network's performance effects are significantly better than the lightweight neural network models LeViT, EfficientNetV2, MobileViTv2, EdgeViTs, EdgeNeXt, and AFFNet. Model performance index comparison experimental results show that the lightweight neural network model we proposed has the lowest number of parameters, the smallest number of operations, and the smallest occupied memory capacity compared to the above six models, and the model's overall performance is the best.

To further understand the timing performance of our proposed model, we used the Android platform to conduct comparative experiments on the inference time of the Linge model and lightweight neural network models such as LeViT, EfficientNetV2, MobileViTv2, EdgeViTs, EdgeNeXt, and AFFNet. The comparative experimental results are shown in Table 7.

Table 7. Model timing performance comparison experimental results.

Basic Algorithm	Inference Time (Seconds)/Photo
LeViT [35]	5.38
EfficientNetV2 [33]	4.89
MobileViTv2 [36]	4.61
EdgeViTs [37]	4.57
EdgeNeXt [38]	4.15
AFFNet [39]	3.93
Linge	3.26

Among the hardware platforms used in our inference speed performance comparison experiments, the operating system is Android 11, the display chip is Mali G52 2EE, the CPU is Allwinner A133, the running memory is 4 GB, and the memory capacity is 64 GB. As shown in Table 7, the proposed model has the fastest inference speed on the same hardware platform.

5.6. Model Optimization Comparison

To improve the performance of the Linge network model we proposed, this article adjusted the Linge network model's relevant parameters in training the prediction model using the "Acute Ischemic Stroke MRI" data set published on Kaggle. The experimental comparison results before and after adjusting the Linge network model parameters are shown in Figure 8.

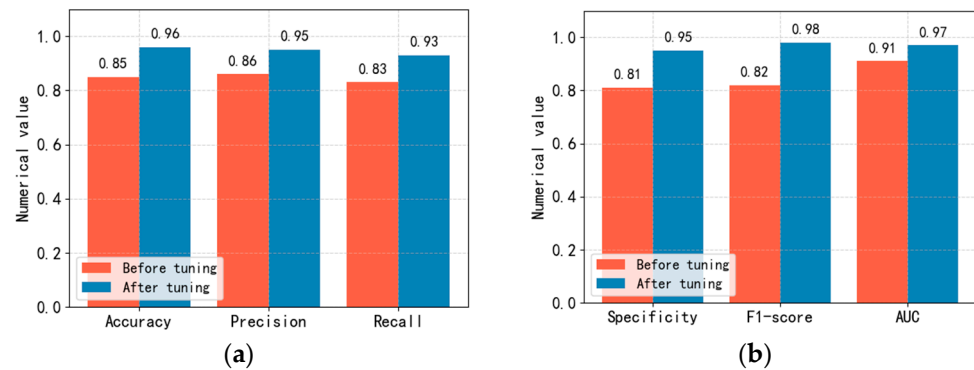


Figure 8. Comparison of accuracy, precision, recall, specificity, F1 score and AUC values before and after adjustment of Linge model parameters. (a): Comparison of accuracy, precision and recall. (b): Comparison of specificity, F1 score and AUC value.

In optimizing model parameters in this article, we mainly adjusted the learning rate, batch size, epochs, activation function, optimization algorithm, and Dropout ratio and conducted related ablation experiments. When conducting ablation experiments, our evaluation metrics are accuracy, precision, recall, specificity, F1 score, and AUC value. To obtain the optimal step size updated in each iteration, we conducted comparative experiments on learning_rate and finally set the learning rate to 0.0001. To obtain the optimal number of samples used to update parameters in each iteration, we conducted comparative experiments on the batch size and set the batch size to 16. During the model training process, we found that the LOSS of the model began to converge at epochs 6300. In the comparative experiment of activation functions, we found that when using the SiLU activation function, the evaluation indicators of the model are relatively the highest. In the comparative experiment of optimization algorithms, we found that when using the Adam optimizer, the evaluation indicators of the model are relatively the highest. In the Dropout proportion comparison experiment, we found that when the random deactivation is set to 0.78, the evaluation indicators of the model are relatively the highest. Figure 8 shows that various evaluation indicators of the prediction model have improved to a certain extent after adjusting relevant parameters.

This section describes in detail the experimental process of our proposed distributed intelligent edge computing technology for disease risk prediction from sample data, evaluation indicators, prediction results, model performance indicators, and model optimization.

6. Conclusions

To meet the diverse application scenarios of disease risk prediction, we propose distributed intelligent edge computing technology for disease risk prediction. To reduce the computing pressure on edge and central servers, we propose a lightweight neural network model. The Linge network model parameter size is only 7.63 MB, and the memory only takes up 155.28 MB. In experiments on stroke risk prediction, the Linge network model achieved an accuracy of 96%, a precision of 95%, a recall of 93%, a specificity of 95%, an F1 score of 98%, and an AUC of 97%. According to the characteristics of MRI images, we further designed an MRI image preprocessing method to improve the model's confidence and generalization ability. The data set used in the experiments of this article is the "Acute Ischemic Stroke MRI" data set publicly available on Kaggle. To verify the distributed

intelligent edge computing technology for disease risk prediction proposed in this article, we conducted verification experiments using subject MRI data. The verification experiment results show that our proposed distributed intelligent edge computing technology for disease risk prediction can be well applied to diverse business scenarios. In future research, we will collect and organize training samples covering more diseases to improve the prediction range of the model.

Author Contributions: F.Z. and S.H. wrote the main manuscript text. X.D. and X.W. provided the idea. J.W. prepared the data and figures. All authors reviewed the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: The work of this paper is supported by the National Key Research and Development Program of China (2019YFB1405000), and the National Natural Science Foundation of China under Grant (Nos. 61873309, 92046024, 92146002).

Data Availability Statement: The “Acute Ischemic Stroke MRI” data set is from <https://www.kaggle.com/datasets/buraktaci/mri-stroke> (accessed on 15 October 2023).

Acknowledgments: Thanks to Peijie Wang, attending TCM physician at Ningbo Hospital of Traditional Chinese Medicine, Zhejiang University of Chinese Medicine, for his medical guidance and cooperation.

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

References

1. Zhou, F.; Hu, S.; Du, X.; Wan, X.; Lu, Z.; Wu, J. Lidom: A Disease Risk Prediction Model Based on LightGBM Applied to Nursing Homes. *Electronics* **2023**, *12*, 1009. [\[CrossRef\]](#)
2. Zhou, F.; Hu, S.; Wan, X.; Lu, Z.; Wu, J. Diplin: A Disease Risk Prediction Model Based on EfficientNetV2 and Transfer Learning Applied to Nursing Homes. *Electronics* **2023**, *12*, 2581. [\[CrossRef\]](#)
3. Zhou, F.; Hu, S.; Wan, X.; Lu, Z.; Wu, J. Risevi: A Disease Risk Prediction Model Based on Vision Transformer Applied to Nursing Homes. *Electronics* **2023**, *12*, 3206. [\[CrossRef\]](#)
4. Tang, Y.; Brown, S.M.; Sorensen, J.; Harley, J.B. Physiology-Informed Real-Time Mean Arterial Blood Pressure Learning and Prediction for Septic Patients Receiving Norepinephrine. *IEEE Trans. Biomed. Eng.* **2021**, *68*, 181–191. [\[CrossRef\]](#)
5. Pan, Y.; Fu, M.; Cheng, B.; Tao, X.; Guo, J. Enhanced Deep Learning Assisted Convolutional Neural Network for Heart Disease Prediction on the Internet of Medical Things Platform. *IEEE Access* **2020**, *8*, 189503–189512. [\[CrossRef\]](#)
6. Guo, Y.; Liu, F.; Cai, Z.; Chen, L.; Xiao, N. FEEL: A Federated Edge Learning System for Efficient and Privacy-Preserving Mobile Healthcare. In Proceedings of the 49th International Conference on Parallel Processing (ICPP '20), Edmonton, AB, Canada, 17–20 August 2020; Association for Computing Machinery: New York, NY, USA, 2020; Article 9. pp. 1–11. [\[CrossRef\]](#)
7. Hakak, S.; Ray, S.; Khan, W.Z.; Scheme, E. A Framework for Edge-Assisted Healthcare Data Analytics using Federated Learning. In Proceedings of the 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 10–13 December 2020; pp. 3423–3427. [\[CrossRef\]](#)
8. Xue, Z.; Zhou, P.; Xu, Z.; Wang, X.; Xie, Y.; Ding, X.; Wen, S. A Resource-Constrained and Privacy-Preserving Edge-Computing-Enabled Clinical Decision System: A Federated Reinforcement Learning Approach. *IEEE Internet Things J.* **2021**, *8*, 9122–9138. [\[CrossRef\]](#)
9. Gupta, D.; Kayode, O.; Bhatt, S.; Gupta, M.; Tosun, A.S. Hierarchical Federated Learning based Anomaly Detection using Digital Twins for Smart Healthcare. In Proceedings of the 2021 IEEE 7th International Conference on Collaboration and Internet Computing (CIC), Atlanta, GA, USA, 13–15 December 2021; pp. 16–25. [\[CrossRef\]](#)
10. Lim, W.Y.B.; Garg, S.; Xiong, Z.; Niyato, D.; Leung, C.; Miao, C.; Guizani, M. Dynamic Contract Design for Federated Learning in Smart Healthcare Applications. *IEEE Internet Things J.* **2021**, *8*, 16853–16862. [\[CrossRef\]](#)
11. Pang, J.; Huang, Y.; Xie, Z.; Li, J.; Cai, Z. Collaborative city digital twin for the COVID-19 pandemic: A federated learning solution. *Tsinghua Sci. Technol.* **2021**, *26*, 759–771. [\[CrossRef\]](#)
12. Gomathy, V.; Janarthanan, K.; Al-Turjman, F.; Sitharthan, R.; Rajesh, M.; Vengatesan, K.; Reshma, T.P. Investigating the Spread of Coronavirus Disease via Edge-AI and Air Pollution Correlation. *ACM Trans. Internet Technol.* **2021**, *21*, 105. [\[CrossRef\]](#)
13. Zhang, D.Y.; Kou, Z.; Wang, D. FedSens: A Federated Learning Approach for Smart Health Sensing with Class Imbalance in Resource Constrained Edge Computing. In Proceedings of the IEEE INFOCOM 2021—IEEE Conference on Computer Communications, Vancouver, BC, Canada, 10–13 May 2021; pp. 1–10. [\[CrossRef\]](#)
14. Wu, Q.; Chen, X.; Zhou, Z.; Zhang, J. FedHome: Cloud-Edge Based Personalized Federated Learning for In-Home Health Monitoring. *IEEE Trans. Mob. Comput.* **2022**, *21*, 2818–2832. [\[CrossRef\]](#)
15. Nguyen, D.C.; Ding, M.; Pathirana, P.N.; Seneviratne, A.; Zomaya, A.Y. Federated Learning for COVID-19 Detection With Generative Adversarial Networks in Edge Cloud Computing. *IEEE Internet Things J.* **2022**, *9*, 10257–10271. [\[CrossRef\]](#)

16. Hayyolalam, V.; Otoum, S.; Özkasap, Ö. A Hybrid Edge-assisted Machine Learning Approach for Detecting Heart Disease. In Proceedings of the ICC 2022—IEEE International Conference on Communications, Seoul, Republic of Korea, 16–20 May 2022; pp. 2966–2971. [\[CrossRef\]](#)
17. Linardos, A.; Kushibar, K.; Walsh, S.; Gkontra, P.; Lekadir, K. Federated learning for multi-center imaging diagnostics: A simulation study in cardiovascular disease. *Sci. Rep.* **2022**, *12*, 3551. [\[CrossRef\]](#)
18. Arikumar, K.S.; Prathiba, S.B.; Alazab, M.; Gadekallu, T.R.; Pandya, S.; Khan, J.M.; Moorthy, R.S. FL-PMI: Federated Learning-Based Person Movement Identification through Wearable Devices in Smart Healthcare Systems. *Sensors* **2022**, *22*, 1377. [\[CrossRef\]](#)
19. Elayan, H.; Aloqaily, M.; Guizani, M. Sustainability of Healthcare Data Analysis IoT-Based Systems Using Deep Federated Learning. *IEEE Internet Things J.* **2022**, *9*, 7338–7346. [\[CrossRef\]](#)
20. Lian, Z.; Yang, Q.; Wang, W.; Zeng, Q.; Alazab, M.; Zhao, H.; Su, C. DEEP-FEL: Decentralized, Efficient and Privacy-Enhanced Federated Edge Learning for Healthcare Cyber Physical Systems. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 3558–3569. [\[CrossRef\]](#)
21. Dang, B.T.H.; Luan, P.H.; Ngan, V.D.T.; Trong, N.T.; Duy, P.T.; Pham, V.-H. TrustFedHealth: Federated Learning with Homomorphic Encryption and Blockchain for Heart Disease Prediction in the Smart Healthcare. In Proceedings of the 2023 International Conference on Advanced Technologies for Communications (ATC), Da Nang, Vietnam, 19–21 October 2023; pp. 178–183. [\[CrossRef\]](#)
22. Wang, R.; Lai, J.; Zhang, Z.; Li, X.; Vijayakumar, P.; Karuppiah, M. Privacy-Preserving Federated Learning for Internet of Medical Things Under Edge Computing. *IEEE J. Biomed. Health Inform.* **2023**, *27*, 854–865. [\[CrossRef\]](#) [\[PubMed\]](#)
23. Ouyang, X. Design and Deployment of Multi-Modal Federated Learning Systems for Alzheimer’s Disease Monitoring. In Proceedings of the 21st Annual International Conference on Mobile Systems, Applications and Services (MobiSys ’23), Helsinki, Finland, 20–22 June 2023; Association for Computing Machinery: New York, NY, USA, 2023; pp. 612–614. [\[CrossRef\]](#)
24. Zhang, L.; Xu, J.; Vijayakumar, P.; Sharma, P.K.; Ghosh, U. Homomorphic Encryption-Based Privacy-Preserving Federated Learning in IoT-Enabled Healthcare System. *IEEE Trans. Netw. Sci. Eng.* **2023**, *10*, 2864–2880. [\[CrossRef\]](#)
25. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141. [\[CrossRef\]](#)
26. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. *arXiv* **2020**, arXiv:1910.03151.
27. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
28. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
29. Tan, M.; Chen, B.; Pang, R.; Vasudevan, V.; Sandler, M.; Howard, A.; Le, Q.V. MnasNet: Platform-Aware Neural Architecture Search for Mobile. *arXiv* **2019**, arXiv:1807.11626.
30. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. *arXiv* **2019**, arXiv:1801.04381.
31. Howard, A.; Sandler, M.; Chu, G.; Chen, L.-C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for MobileNetV3. *arXiv* **2019**, arXiv:1905.02244.
32. Tan, M.; Le, Q.V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv* **2020**, arXiv:1905.11946.
33. Tan, M.; Le, Q.V. EfficientNetV2: Smaller Models and Faster Training. *arXiv* **2021**, arXiv:2104.00298.
34. Tasci, B.; Tasci, I. Deep feature extraction based brain image classification model using preprocessed images: PDRNet. *Biomed. Signal Process. Control* **2022**, *78*, 103948. [\[CrossRef\]](#)
35. Graham, B.; El-Nouby, A.; Touvron, H.; Stock, P.; Joulin, A.; Jégou, H.; Douze, M. LeViT: A Vision Transformer in ConvNet’s Clothing for Faster Inference. *arXiv* **2021**, arXiv:2104.01136.
36. Mehta, S.; Rastegari, M. Separable Self-attention for Mobile Vision Transformers. *arXiv* **2022**, arXiv:2206.02680.
37. Pan, J.; Bulat, A.; Tan, F.; Zhu, X.; Dudziak, L.; Li, H.; Tzimiropoulos, G.; Martinez, B. EdgeViTs: Competing Light-weight CNNs on Mobile Devices with Vision Transformers. *arXiv* **2022**, arXiv:2205.03436.
38. Maaz, M.; Shaker, A.; Cholakkal, H.; Khan, S.; Zamir, S.W.; Anwer, R.M.; Khan, F.S. EdgeNeXt: Efficiently Amalgamated CNN-Transformer Architecture for Mobile Vision Applications. *arXiv* **2022**, arXiv:2206.10589.
39. Huang, Z.; Zhang, Z.; Lan, C.; Zha, Z.-J.; Lu, Y.; Guo, B. Adaptive Frequency Filters As Efficient Global Token Mixers. *arXiv* **2023**, arXiv:2307.14008.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.