*sustainability*

MDPI

*Article*

# A Multi-Criteria Approach for Arabic Dialect Sentiment Analysis for Online Reviews: Exploiting Optimal Machine Learning Algorithm Selection

**Mohamed Elhag Mohamed Abo [1,*], Norisma Idris [1,*], Rohana Mahmud [1], Atika Qazi [2], TIbrahim Abaker Targio Hashem [3], Jaafar Zubairu Maitama [1,4], Usman Naseem [5], Shah Khalid Khan [6] and Shuiqing Yang [7]**

1 Department of Artificial Intelligence, Faculty of Computer Science and Information Technology, Universiti Malaya, Kuala Lumpur 50603, Malaysia; rohanamahmud@um.edu.my (R.M.); jzmaitama.it@buk.edu.ng (J.Z.M.)
2 Centre for Lifelong Learning, Universiti Brunei Darussalam, Gadong BE1410, Brunei; atikaqazium@gmail.com
3 Department of Computer Science, College of Computing and Informatics, University of Sharjah, Sharjah 27272, United Arab Emirates; ihashem@sharjah.ac.ae
4 Department of Information Technology, Faculty of Computer Science and Information Technology, Bayero University, Kano 3011, Nigeria
5 School of Computer Science, University of Sydney, Sydney, NSW 2006, Australia; usman.naseem@sydney.edu.au
6 School of Engineering, RMIT University, Carlton, VIC 3053, Australia; shahkhalid_k@yahoo.com
7 School of Information Management and Artificial Intelligence, Zhejiang University of Finance and Economics, Hangzhou 310018, China; yangshuiqing@zufe.edu.cn
* Correspondence: aboo72me@siswa.um.edu.my (M.E.M.A.); norisma@um.edu.my (N.I.)

**Abstract:** A sentiment analysis of Arabic texts is an important task in many commercial applications such as Twitter. This study introduces a multi-criteria method to empirically assess and rank classifiers for Arabic sentiment analysis. Prominent machine learning algorithms were deployed to build classification models for Arabic sentiment analysis classifiers. Moreover, an assessment of the top five machine learning classifiers' performances measures was discussed to rank the performance of the classifier. We integrated the top five ranking methods with evaluation metrics of machine learning classifiers such as accuracy, recall, precision, F-measure, CPU Time, classification error, and area under the curve (AUC). The method was tested using Saudi Arabic product reviews to compare five popular classifiers. Our results suggest that deep learning and support vector machine (SVM) classifiers perform best with accuracy 85.25%, 82.30%; precision 85.30, 83.87%; recall 88.41%, 83.89; F-measure 86.81, 83.87%; classification error 14.75, 17.70; and AUC 0.93, 0.90, respectively. They outperform decision trees, K-nearest neighbours (K-NN), and Naïve Bayes classifiers.

**Keywords:** multiple-criteria; Arabic dialect; sentiment analysis; machine learning; performance evaluation



**Citation:** Abo, M.E.M.; Idris, N.; Mahmud, R.; Qazi, A.; Hashem, I.A.T.; Maitama, J.Z.; Naseem, U.; Khan, S.K.; Yang, S. A Multi-Criteria Approach for Arabic Dialect Sentiment Analysis for Online Reviews: Exploiting Optimal Machine Learning Algorithm Selection. *Sustainability* **2021**, *13*, 18. https://doi.org/10.3390/su131810018

Academic Editor: Amir Mosavi

Received: 20 July 2021
Accepted: 23 August 2021
Published: 7 September 2021



## 1. Introduction

Machine learning (ML) enables systems to learn, adapt, and enhance knowledge without explicit programming [1]. It was applied across diverse domains such as research and business [2,3]. There are many different ML algorithms and parameters as well as experts' knowledge about the available classification algorithms and domain in order to select an appropriate solution [4]. The fact that ML-oriented datasets have varying intrinsic characteristics, various experts in the field of data analysis may have incomplete knowledge about the application domains. Moreover, the strengths and capabilities of the candidate classifiers may not be fully understood. It is widely known [5,6] that no ML algorithm performs well for all learning problems. It is, therefore, necessary to evaluate

ML algorithms separately for each application scenario [7] because domains have diverse, relevant characteristics [8]. Machine learning algorithms perform well with multiple languages as a significant amount of works focused on the English language, followed by Arabic, Chinese, etc.

In the context of performance measures for classification-based algorithms for Arabic sentiment analysis (SA), the prominent measures employed by most of the researchers in ML comprises precision, recall, F-measure, area under the ROC curve [9], balanced accuracy, and average accuracy. However, an analysis using only one of these may select an algorithm that does not yield the optimum performance [10,11]. Moreover, for morphologically rich languages, such as Arabic or dialect Arabic, additional selection criteria are also relevant, such as training time, execution time, and results inconsistency. A multi-criteria-based evaluation approach is, therefore, necessary to identify the optimal solution [6,12]. The primary objective of this study is to introduce a multi-criteria method to identify an appropriate ML algorithm for sentiment analysis of dialect Arabic and to demonstrate it on a corpus of Saudi tweets. We have compared the performance of five machine learning algorithms classifiers: deep learning, decision tree, Naïve Bayes, K-nearest neighbours, and support vector machine with multi-criteria using dialect Arabic dataset of tweets to select better classifier for dialect Arabic. The result of the experiment shows that deep learning and support vector machine classifiers showed better performance results in terms of accuracy, precision, recall, F-measure, and AUC compared to a decision tree, K-nearest neighbours, and Naïve Bayes classifier.

In consideration of the fact that Arabic is a popular language with over 400 million speakers in North Africa, the Middle East, and the Horn of Africa [13,14], it was revealed that the Arabic language is among the world most influential languages that concur significant parts of the world with incredible speakers [15]. The Arabic language is one of the authority languages utilised at the United Nations and an official language of 27 countries. There are three forms of Arabic: Modern Standard Arabic (MSA), Classical Arabic (CA), and Dialectical Arabic (DA) [16–19]. MSA is used in formal situations and events, CA is found in religious scripts, and DA is used in everyday life, including for reviews and comments posted on social media. It includes national dialects, such as Egyptian and Moroccan. These have evidently revealed the importance of DA for research-oriented considerations among both novice and veteran researchers.

In more specific terms, dialect Arabic (DA) has been estimated to be the communication medium of almost 295 million individuals living mostly in the Middle East and North Africa [15]. The contemporary advent of a social medium of communication has increased DA exponentially across the globe; however, as a Morphologically Rich Language (MRL) where an unusual amount of data concerning syntactic units and relations communicates at the word level. The huge dialect Arabic (DA) online contents involved sentiments that need to be deciphered for more semantically oriented benefits, which empirically motivates the zeal to conduct this study.

## 2. Related Work

Recently many research was performed on sentiment analysis to identify and categorise opinions expressed in the text, especially social media, or simply for classification purposes. Some of the Machine Learning algorithms used in sentiment analysis are based on multiple criteria such as accuracy, recall, and F-measure [20–24]. Furthermore, the area under the curve, sensitivity, precision, and F-score are presented in [2,25,26]. Wegrzyn-Wolska, Bougueroua [27] conducted a content analysis of social networks by analysing the intensity and polarity of opinions and developed a system that collects, evaluates, and rates tweets automatically for trend evaluation using a Twitter dataset related to French presidential elections. In [16], supervised sentiment analysis was used for (Arabizi type) Arabic languages that use Latin characters; the result shows a recall up to 86.9%. Salameh, Mohammad [28] proposed a supervised sentiment analysis using word-level and character N-Grams. The result showed that an accuracy of 87% was obtained.

In [29], a Naïve Bayes was used to perform supervised sentiment analysis for Arabic languages. A Multi-algorithm [29] was used, with Naïve Bayes as well as a bag of words. The algorithms were applied to a dataset collected from e-commerce websites, and the result revealed that the algorithm achieved 93.87% accuracy for Naïve Bayes. Alayba, Palade [30] applied the hyper algorithm with Naive Bayes, unigram, and bigram features to the algorithm on the Twitter dataset. The result shows that the accuracy is up to 90%. Similarly, [31] used Naïve Bayes with Precision, Recall, and F-Measure metrics, from which the result sections show that the Naïve Bayes lexicon accuracy reached 90%.

Moreover, [32] applied Naïve Bayes in their experiments with a dataset collected from social networks such as Twitter, Facebook. Hathlian and Hafezs [32] addressed the issue related to the lack of lexicons for analysis and testing in the Arabic sentiment analysis context. A relatively large dataset of modern standard Arabic (MSA) comments and reviews was collected. The result for the Neive Bayes performance accuracy is up to 85%. Moreover, a supervised approach was used by [33]; the study used hyper languages Iraqi, Egyptian and Lebanese dialect Arabic with 10-fold cross-validation. The result showed an accuracy of 87.9%.

Alqarafi, Adeel [34] introduced a semi-supervised approach for constructing an annotated sentiment corpus for the Saudi dialect. The idea is to use a list of lexicons developed using embedding techniques. The annotation of a large number of the corpus is generated from Twitter by manual check in order to remove the incorrect annotated tweets. The result showed that the classifier as Naïve Bayes outperformed other classifiers by up to 91% accuracy. Cambria, Poria [35] provided a review of computational intelligence for affective computing and sentiment analysis. The authors summarise various papers on emotion and sentiment by highlighting their limitations and opportunities.

Abdulkareem and Tiun [33] conducted a sentiment analysis study using k-nearest neighbours in Arabic dialect datasets gathered from Twitter. The study illustrated that by using cross-validation (10-fold), the algorithm achieved up to 87.9% accuracy. Moreover, [36,37] used the support vector classifier and k-nearest neighbours to perform sentiment analysis in Arabic and Saudi dialects' languages. The results show that the algorithms achieved 95% Accuracy on data collected from Twitter. Khasawneh, Wahsheh [15] Conducted sentiment analysis for Arabic languages by applying a decision tree with Iterative (J48) algorithms. The classifiers were applied to a multi-domain dataset from Facebook and Twitter. The experiment results showed a high result of up to 93%. Abdulkareem and Tiun [33] used decision tree 10-fold cross-validation with other machine learning classifiers to perform a sentiment analysis experiment with hyper languages dialect Arabic collected from Twitter. The result showed an accuracy of 87.9%. A summary of selected algorithms used in Arabic sentiment analysis is shown in Table 1.

**Table 1.** A summary of the main algorithms for Machine Learning based Arabic SA.

| Ref | Algorithms | | | | | Descriptions |
|---|---|---|---|---|---|---|
| | SVM | NB | K-NN | DT | DL | |
| [38] | - | 90% | - | - | - | Accuracy |
| [32] | 89.5% | 85.1% | - | - | - | Accuracy |
| [39] | 73.5 | 71.6 | 70 | 65.1 | - | Recall (Average) |
| [15] | 93.3% | 91.8% | 31.4% | 92.4% | - | Accuracy |
| [40] | 74.7% | 74.1% | - | - | - | Accuracy |
| [41] | 61.4% | 67.9% | - | - | - | Accuracy |
| [42] | 68.2% | 61.4% | - | - | - | Accuracy |
| [43] | 73.2 | - | - | - | - | Stem + Morph + language independent features |
| [44] | 68.2 | 61.4 | - | - | - | Accuracy, Recall |
| [45] | 71.7 | 76.8 | 60 | - | - | Accuracy |
| [46] | 85 | 81.3 | 52.9 | 50 | - | Accuracy |
| [29] | 94.8 | 87.3 | 80.1 | - | - | Precision |
| [47] | 73.5 | - | 72.2 | - | - | Accuracy |

**Table 1.** *Cont.*

| Ref | Algorithms | | | | | Descriptions |
|---|---|---|---|---|---|---|
| | SVM | NB | K-NN | DT | DL | |
| [42] | 87.4 | 65.9 | - | - | - | Accuracy |
| [48] | 66.1 | - | - | - | 60.4 | Accuracy |
| [49] | 77.7 | 78.2 | - | 80.9 | - | Precision |
| [50] | 84.9 | 77.5 | - | - | - | Accuracy |
| [51] | 93.4 | - | - | - | - | Accuracy |
| [52] | 72.6 | 65.4 | - | - | - | Accuracy |
| [53] | 91.5 | - | - | - | - | Accuracy |
| [54] | 82.5 | 85.7 | 66.7 | - | - | Accuracy |
| [30] | 90.9 | 90.1 | - | - | - | Accuracy |
| [31] | 87.8 | 95.4 | - | - | - | Macro-Precision |
| [16] | 86.9 | 82.1 | - | - | - | Recall |
| [36] | 90 | - | 90.5 | - | - | Accuracy |
| [55] | 92 | - | - | - | - | Accuracy |
| [33] | - | 93.9 | 88.8 | 92.1 | - | Recall |
| [56] | 83.2 | 80.5 | 82.4 | - | - | Macro-F1 |
| [57] | 95.1 | - | - | - | - | Accuracy |
| [28] | 87 | - | - | - | - | Accuracy |
| [58] | 47.4 | 48.9 | 57.8 | 47.6 | - | Accuracy |
| [59] | 93 | - | - | - | - | Accuracy |

SVM = Support Vector Machine, DL = Deep Learning, NB = Naïve Bayes, k-NN = K-nearest Neighbours, and DT = Decision Tree.

## 3. Materials and Methods

This section outlines a five-step method to select machine learning algorithms for Arabic sentiment analysis: (1) Data collection; (2) Pre-processing; (3) Feature selection; (4) Data analysis; (5) Evaluation (Figure 1). The novel component of this method is the multiple criteria final evaluation stage. Figure 1 used to presents an overview of the five-step methods.

### *3.1. Data Collection*

A set of 11,647 Saudi dialect Arabic tweets [60,61] was used with sentiment labels from online tools. Moreover, a large number of students were invited to register in the system available online and classify the texts made up of short review sentences to positive and negative, which include 5255 positive and 6392 negative tweets. The corpus consists of only Saudi dialect Arabic reviews about various domains such as restaurants, shopping, fashion, education, entertainment, hotels, motors, and tourism. In this study, only the positive and negative reviews were considered, while neutral, sarcastic, modern standard Arabic (MSA), and unknown tweets were excluded.

### *3.2. Pre-Processing*

The pre-processing stages include filtering, tokenisation, and Arabic stemming.

#### 3.2.1. Tokenisation: Or 1.1 (Tokenisation)

Tokenisation involves the breakdown of a naturally occurring input string of orthographic symbols into elementary symbols named "tokens" for subsequent processing. Since there is no single correct tokenisation, a tokenisation system is a purely analytical tool [62].

Tokenisation systems segment the text of a document into a sequence of tokens. There are numerous options for specifying the splitting points. The default setup involves the use of all non-letter characters. The outcome generates tokens that consist of one single word, which is the most suitable option for sentiment analysis [61].

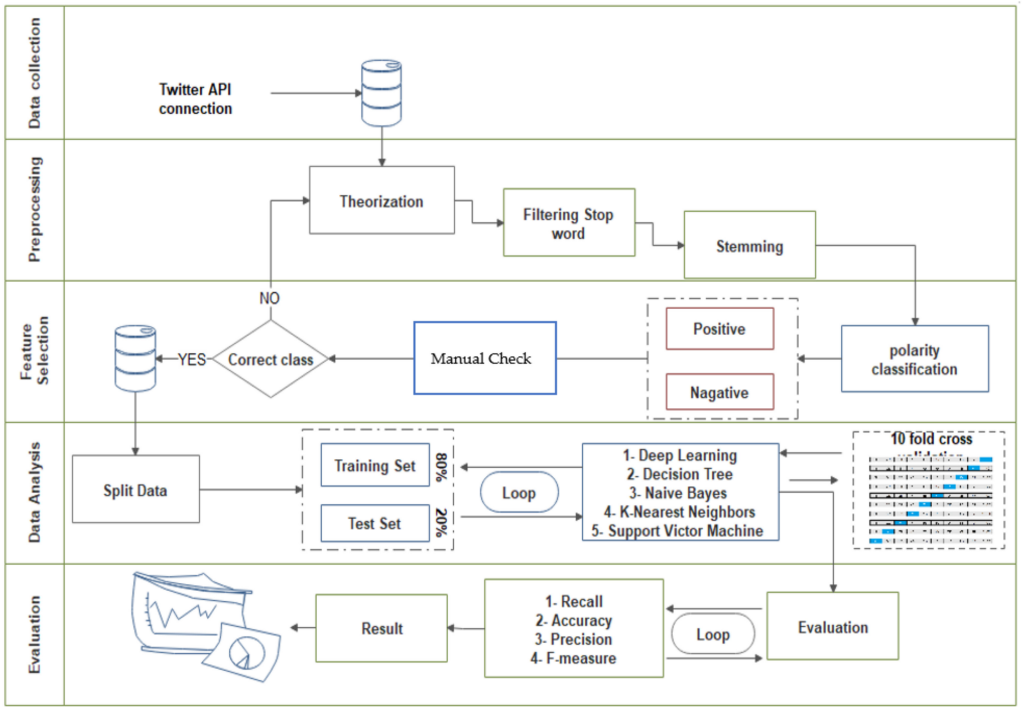


**Figure 1.** Overview of the five-stage method to evaluate SA algorithm for Arabic.

#### 3.2.2. Filtering Step Words

The next pre-processing stage excludes Arabic stopwords with the help of "arabicStemR", a package for stemming Arabic in the R programming language. Frequent words with less than three letters are removed. In general, stop words usually only convey syntactic meanings. These include the words "or" (أو, ( فثً*an*)), "in" ("إٍ ح), "were" ("على), "on" ("لمن), "whose" (كثث), "all" ( إٍ غ), and "beyond" ("مثثن)". The omission of stop words makes the documents shorter in length, which leads to quicker processing and increases the effectiveness of term indexing [63].

#### 3.2.3. Stemming

A light stemmer algorithm was used to remove the most frequently occurring prefixes and suffixes. This algorithm alters the form of the words in certain cases [64]. The initial "و)" and definite articles "لل","وال","ال","كال","فال","بال" and suffixes "ها","ان","ات","ون","ين","يه","ية","ه","ة","ي" are removed by the light stemmer and light10 algorithms. These algorithms allowed recurrent suffixes and prefixes to be removed from strings.

### *3.3. Feature Selection*

The tweets were labelled as positive or negative. In addition, the result was transferred to two Arabic language experts at the University of Khartoum to validate the labels.

#### Manual Annotations

An Arabic language expert assessed the accuracy of a random sample of the RapidMiner polarity labels [65]. More than 5000 mixed datasets were annotated manually. Moreover, the must set the polarity of the sentence based on three classes, these being

neutral, negative, or positive. While this process, the inter-agreement between them was measured using the Kappa coefficient.

$$k = \frac{P_{agree} - P_{chance}}{1 - P_{chance}} \quad (1)$$

$P_{agree}$ = Proportion of trials in which judges agree.
$P_{chance}$ = Proportion of trials in which agreement would be expected due to chance.

The tweets and their analogous annotations are saved in a database. Each tweet needs to own at least three labels to have completed labelling. These human manual labels are recognised as the ideal golden truth, and the accuracy of the classifiers is calculated concerning these labels. Moreover, the following formula was used to estimate the necessary sample size for the check:

$$n = \frac{2\left(Z_a + Z_{1-\beta}\right)^{2\sigma 2,}}{\Delta^2} \quad (2)$$

where $Z$ is a constant (set by convention according to the accepted $\alpha$ error and whether it is a one-sided or two-sided effect) and $n$ is the required sample size for $Z_a$ [66].

### *3.4. Data Analysis*

#### 3.4.1. Machine Learning Algorithms

The five most common classifiers were selected for comparative analysis: decision tree, support vector machine, K-nearest neighbours, deep learning, and Naïve Bayes. These are popular and have been reasonably accurate in previous similar tasks [67–69]. In addition, more than 21 existing ML classifiers for Arabic language performance analysis and recommendation methodologies were rigorously examined. The best performing classifiers with higher frequencies were selected to represent the top five ML algorithms. In the end, we evaluate the performance of the top five ML classifiers selected.

Support Vector Machine

Support vector machines determine the maximum margin hyperplane that offers the highest separation between the classes, while the samples that are closest to this hyperplane are called the support vectors [70].

The optimal hyperplane function is expressed as

$$h(x) = wTx + b, \quad (3)$$

For any new point **z**, the class of support vector machines classifier is predicted as

$$\hat{y} = sign(h(z)) = sing(wTz + b) \quad (4)$$

where the sign "·" function returns −1 if its argument is negative, and +1 if its argument is positive [63].

Support vector machines are intrinsically two-class classifiers; a multiclass problem is formed through the construction of multiclass support vector machines, where a two-class classifier is developed over a feature vector $\Phi\left(\vec{x},y'\right)$ obtained from the pair that constitutes the class of the datum and the input characteristics.

The classifier selects the class at test time using the following expression

$$y = argmax_{y'}\ \vec{w}^T \Phi\left(\vec{x},y'\right) \quad (5)$$

In the course of training, the margin refers to the gap or difference between the value for the correct class and the other closest class, and therefore the quadratic program formulation will require the following:

Naïve Bayes

For the Naïve Bayes classifier, the probabilities of classes are ascertained with the help of a feature vector table. The class that has the maximum posterior probability is then assigned the classification. In general, the text classification is carried out through the implementation of two models of the Naïve Bayes approach: Bernoulli's multivariate and multinomial. The Naïve Bayes is basically a stochastic model applied to generate documents, which adheres to the Bayes' rule expressed as follows [71,72]:

$$\frac{P(c|x) = P(c).P(x)}{P(x)} \tag{6}$$

Above:

- $P(c \mid x)$ is the posterior probability of class (c, target) given predictor (x, attributes);
- $P(c)$ is the prior probability of class;
- $P(x \mid c)$ is the likelihood which is the probability of predictor given class;
- $P(x)$ is the prior probability of predictor.

To find the probability of the tweets class given the tweet, we adopt Equation (2), as used in [73].

$$P\left(\frac{C}{t}\right) = P(C)P\prod_{i=1}^{n} P(f_i|C) \tag{7}$$

Here, C represents the classes negative, positive, and neutral; t represents the tweet; and f represents the feature. The NLTK implementation of the Naïve Bayes algorithm was used [74].

K-Nearest Neighbours

In the k-nearest neighbours algorithm, an unknown example is compared with the $k$ training examples that are its closest neighbours. In the process of applying the k-nearest neighbours algorithm to a new example, the first step is to identify the k closest training examples. The term "closeness" is defined as a distance in the n-dimensional space, which is embodied using the $n$ attributes in the training example set. Other metrics can also be used to represent the distance between the unknown example and the training examples—for instance, the Euclidean distance. The normalisation of data before training and application of the k-nearest neighbours algorithm is recommended because the distances usually depend on absolute values. The accurate configuration is defined by the parameters of the operator and the metric used. In the second step, the unknown example is classified by the k-nearest neighbours algorithm based on the majority vote of the identified neighbours. The predicted value, in the case of a regression, is defined as the average of the values of the found or identified neighbours.

The k nearest neighbours are assigned a weight of 1/k and all others a weight of 0 by the k-nearest neighbours classifier. This generalisation can be further applied to the weighted closest or nearest neighbours classifiers. In other words, where the $i$th nearest neighbours are assigned a weight $(w_{ni})$ with $\sum_{i=0}^{n} w_{ni} = 1$, an analogous result is applied to the strong and robust consistency of weighted nearest neighbours classifiers [74].

$C_n^{wnn}$ signifies the weighted nearest classifier with weights $\{w_{ni}\}_{i=1}^{n}$. The excess risk has the following asymptotic expansion subject to the regularity conditions applicable for class distributions [75].

$$R_R C_n^{wnn} - R_R\left(C^{Bayes}\right) = B_1 s_n^2 + B_2 t_n^2)\{1 + (1)\}, \tag{8}$$

For constants $B_1$ and $B_2$ where

$$s_n^2 = \sum_{i=1}^{n} w_{ni}^2 and\ t_n = n^{-2/d} \sum_{i=1}^{n} w_{ni}\left\{i^{1+2/d} - (i-1)^{1+2/d}\right\} \tag{9}$$

The following expressions represent the optimal weighting scheme $\{w_{ni}^*\}_{i=1}^n$ that balances the two terms in the display above set $w_{ni}^* = \frac{1}{k^*}\left[1 + \frac{d}{2} - \frac{d}{2k^{*2/d}}\left\{i^{1+2/d} - (i-1)^{1+2/d}\right\}\right]$ for $i = 1, 2, \ldots, k^*$ and $w_{ni}^* = 0$ for $i = k^* + 1, \ldots, n$. The dominant term in the asymptotic expansion of the excess risk, after the consideration of the optimal weights, is $\mathcal{O}\left(n^{-\frac{4}{d+4}}\right)$.

Decision Tree

In a decision tree, a node indicates the test of an attribute value, and a branch signifies the result of the test. An ensemble of classifiers is created in a decision tree through the formation of various decision trees at the training stage that uses arbitrary feature selection and bagging methodology. The decision tree creates two kinds of nodes: the interior node linked with a feature and the leaf node labelled as a class [49].

Mathematical Formulation

With training vectors $x_i \in R^n$, $i = 1, \ldots, I$ and a label vector, $y \in R^l$, a decision tree recurrently partitions and separates the space so that the samples with the same labels are classified together.

In the following expressions, $Q$ represents the data at node m. Every candidate split $\theta = (j, t_m)$ comprises a threshold $t_m$ and a feature, and the data are partitioned into $Q_{left}(\theta)$ and $Q_{right}(\theta)$ subsets.

$$Q_{left}(\theta) = (x, y)|x_j <= t_m \tag{10}$$

$Q_{right}(\theta) = Q \backslash Q_{left}(\theta)$ the impurity at $m$ is calculated with the help of an impurity function $H(\mathrm{n})$, the selection of which is based on the task being solved (regression or classification).

$$G(Q, \theta) = \frac{n_{left}}{N_m} H\left(Q_{left}(\theta)\right) + \frac{n_{right}}{N_m} H\left(Q_{right}(\theta)\right) \tag{11}$$

The parameters that minimise the impurity is selected

$$\theta^* = argmin_\theta G(Q, \theta) \tag{12}$$

Recourse for subsets $Q_{left}(\theta^*)$ and $Q_{right}(\theta^*)$ until the maximum permissible depth is attained, $N_m < min_{samples}$ *or* $N_m = 1$.

Deep Learning

Deep artificial neural networks are a group of algorithms reaching new levels of accuracy for several important issues, such as image recognition, sound recognition, recommender systems [76–78]. A deep neural network is a type of neural network. It consists of three parts: the input layer (X), the multi-layer hidden layer (Hn), and the output layer (Y). Figure 2 shows the multi-layer structure of a deep learning algorithm, which can express complex non-linear relationships. It uses neural networks to simulate human decision-making. Deep learning can be computationally expensive and needs large datasets for training. This is because there are many parameters to learn. For example, a deep learning algorithm might be trained to "learn" what a cat looks like. It would take an enormous dataset of pictures for it to understand the minor details that distinguish a cat from a wolf or a fox. Deep learning works by:

- Using a cascade of multiple layers of non-linear processing units for feature extraction and transformation. Every successive layer uses the output from the previous layer as input;
- Learning multiple levels of representation that correspond to different levels of abstraction within a hierarchy of concepts [76].

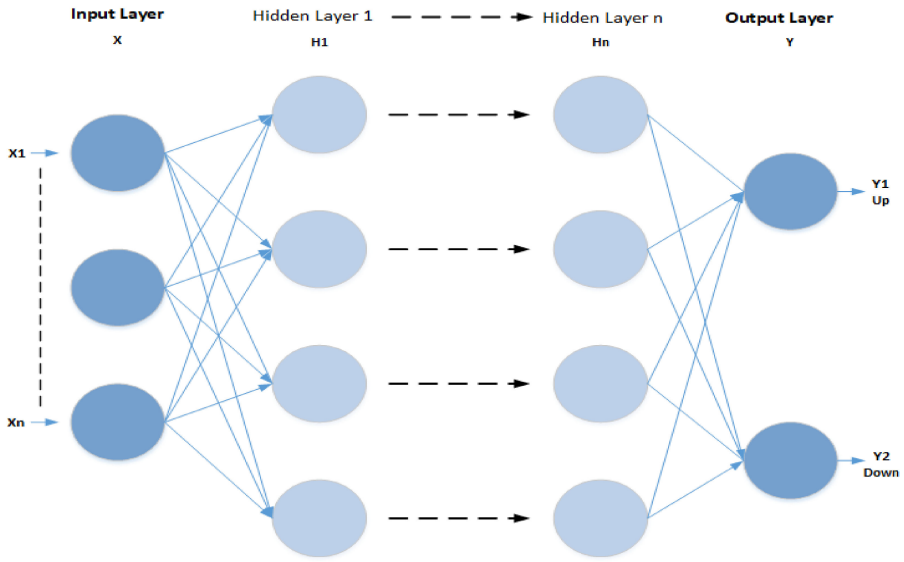


**Figure 2.** The Multi-layer Structure of a Deep Learning Algorithm.

#### 3.4.2. Advantages and Disadvantages of the Prominent ML Algorithms for Arabic SA

The top five ML algorithms' key advantages and disadvantages were discussed in Table 2, which is a cover classifier that is used for a different type of Arabic dataset collected from various domains such as book reviews, hotels, and restaurants.

**Table 2.** Key Advantages and Disadvantages of five Machine Learning Algorithms.

| Algorithm | Advantage | Disadvantage |
| --- | --- | --- |
| Support Vector Machines | Non-linear decision boundaries can be modelled by support vector machines. There are different kernels to make the selection. They are also equitably robust against overfitting, specifically in high-dimensional space [49]. | Support vector machines are memory intensive. They are difficult to scale up to massive datasets and are complex to tune because the weights have given to the selection of the right kernel. In the industry at present, random forests are generally preferred over support vector machines [79]. |
| Decision Tree | Decision trees are comparatively robust to outliers and are capable of learning non-linear relationships. When it comes to practice, ensembles perform exceptionally well, winning several classical (that is, non-deep-learning) machine learning competitions. | Individual trees that are unconstrained are inclined towards overfitting as they can continue branching until they store the training data in memory. Ensembles can improve or alleviate this [80]. |
| Naïve Bayes | Despite the fact that the assumption of conditional independence seldom holds, Naïve Bayes models perform well. The scale and are easy to implement. | Naïve Bayes models, as a result of their absolute simplicity, are frequently outclassed by models that are suitably trained and tuned with the help of the earlier listed algorithms. |
| Deep Learning | When it comes to the classification of text, audio, or image data, deep learning performs extremely well [81]. | Deep neural networks are similar to regression and require huge amounts of data for training, and therefore they are not regarded as a general-purpose algorithm. |
| Nearest Neighbours | Nearest neighbours algorithms are "instance-based," which implies that every training observation is considered. They then predict new observations through the search for the most similar or parallel training observations and combining their values. | As these algorithms are highly memory-intensive, their performance is weak for high-dimensional data. A meaningful distance function is required by them to evaluate and calculate similarity. However, in practice, tree ensembles or training regularised regression nearly always optimises the use of time. |

### *3.5. Evaluation Metrics*

To evaluate ML algorithms, we performed 10-fold cross-validation with 90% as a test set. We assessed their performance using Accuracy, Precision, Recall, and F-measure, which were proposed for performance evaluation in multi-label learning [82], as defined below.

The Accuracy provided by Equation (13) measures the most intuitive performance measure, and it is merely a ratio of correctly predicted observation to the total observations [83].

$$\text{Accuracy} = \frac{\sum true\ positive + \sum True\ negative}{\sum Total\ population} \tag{13}$$

On the other hand, Precision provided by Equation (14) measures the number of True Positives divided by the number of True Positives and False Positives [83].

$$\text{Precision} = \frac{true\ positive}{true\ positive + false\ positive} \tag{14}$$

In this context, Recall measures expressed by Equation (15) are also referred to as the true positive sensitivity or rate, and precision is also known as the positive predictive value (PPV). Accuracy and true negative rate are the other relevant measures that are used in classification. Specificity is another term for the true negative rate [83].

$$Recall = \frac{true\ positive}{true\ positive + false\ negative} \tag{15}$$

F-measure, as expressed in Equation (16), is a measure that represents the combined precision, while recall indicates the harmonic mean of precision and recall, and the balanced F-score or the traditional F-measure [83].

$$F - measure = \frac{2.Precision.\ Recall}{(Precision + Recall)} \tag{16}$$

In addition to this, another two evaluation measures were included, which is significant for the assessment of the performance. The first measure is the area under the curve (AUC) [84].

$$F_{abs} = \left(\frac{AUC_{non-IV}}{AUC_{IV}}\right) \times \left(\frac{Dose_{IV}}{Dose_{non-IV}}\right) \tag{17}$$

Given that an algorithm's runtime may vary based on different inputs of the same size, the send one is a runtime (times complexity). The worst-case time complexity refers to the maximum amount of time needed for inputs of a certain volume. On the other hand, the average-case complexity, which is relatively less common and generally stated explicitly, denotes the average of the time taken for inputs of a given size. The time complexity is usually represented as a function of the size of the input in both cases [85].

## 4. Results and Discussion

The results are discussed separately for each evaluation criterion. Moreover, to ensure the performance of the classifiers, we combined various domains to test the accuracy of five machine learning algorithms using Arabic dialect features.

### *4.1. Accuracy*

Figure 3 illustrates the performance of Naïve Bayes, decision tree, SVM, K-NN, and deep learning classifiers. It indicates that deep learning shows better accuracy (85.25%) compared to a decision tree, Naïve Bayes, k-NN, and SVMs. However, SVM also has shown slightly better accuracy, which is 82.30% better than decision tree, k-NN, and Naïve Bayes, 79.31%, 61.41%, and 57.56%, respectively.

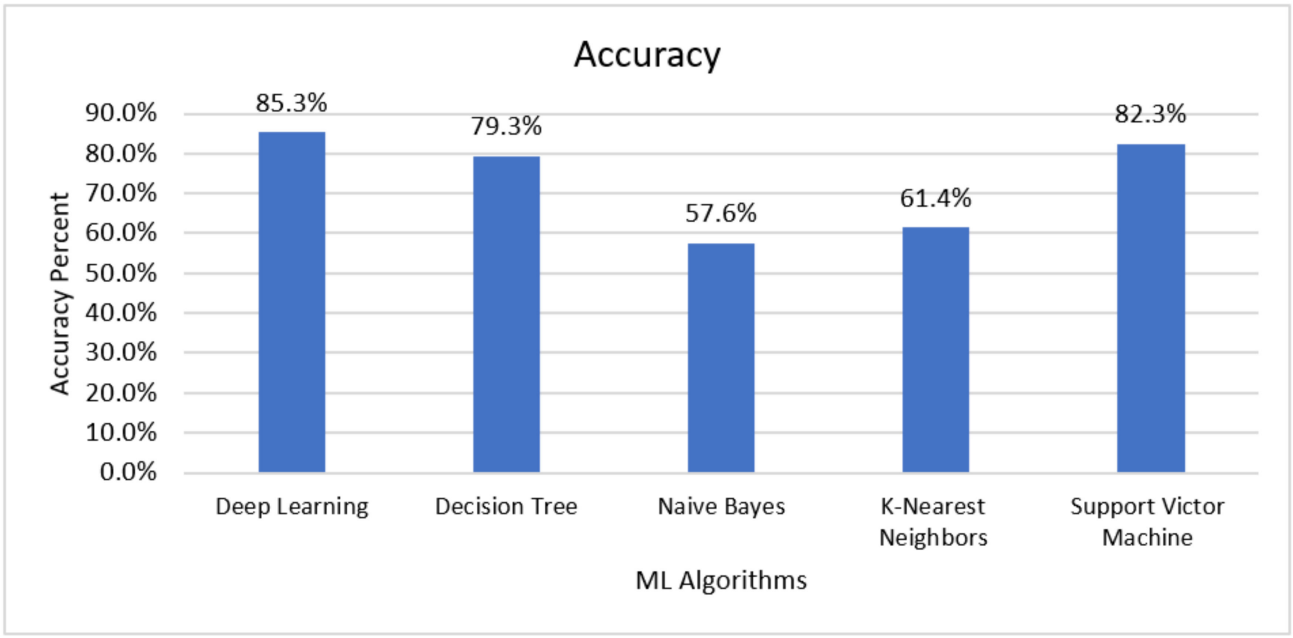


**Figure 3.** Accuracy of Top Five Machine Learning Algorithms.

### *4.2. Precision*

Figure 4 shows the various result of precision for deep learning, decision tree, Naïve Bayes, K-NN, and SVM classifiers. It appears that the decision tree is core (96.67%) compared to deep learning, Naïve Bayes, k-NN, and SVMs. However, deep learning and SVM also showed a slightly better result, which is 85.30%, 83.87%, respectively, better than Naïve Bayes, and K-NN.

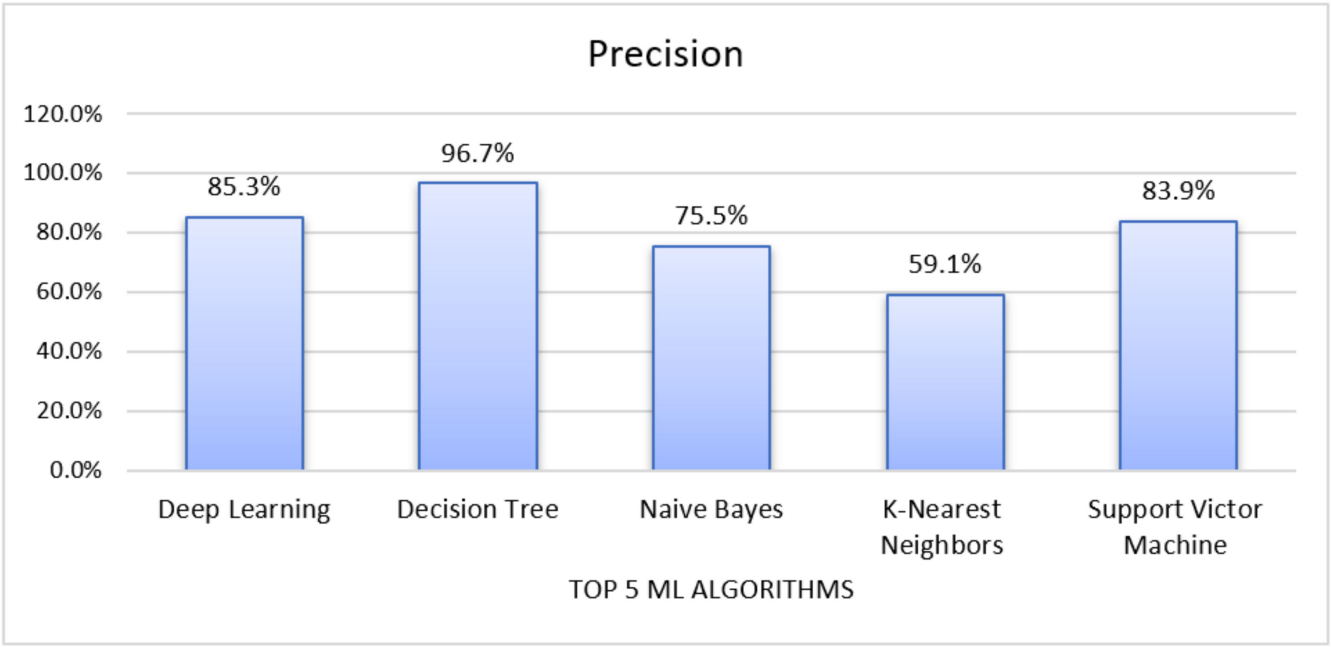


**Figure 4.** Precision Result for Five Machine Learning Algorithms.

### *4.3. Recall*

Figure 5 illustrates the various result of deep learning, decision tree, Naïve Bayes, K-NN, and SVM classifiers. It indicates that deep learning shows better accuracy (85.25%)

compared to a decision tree, Naïve Bayes, k-NN, and SVMs. However, SVM also has shown slightly better accuracy, which is 82.30% better than decision tree, k-NN, and Naïve Bayes, 79.31%, 61.41%, and 57.56%, respectively.

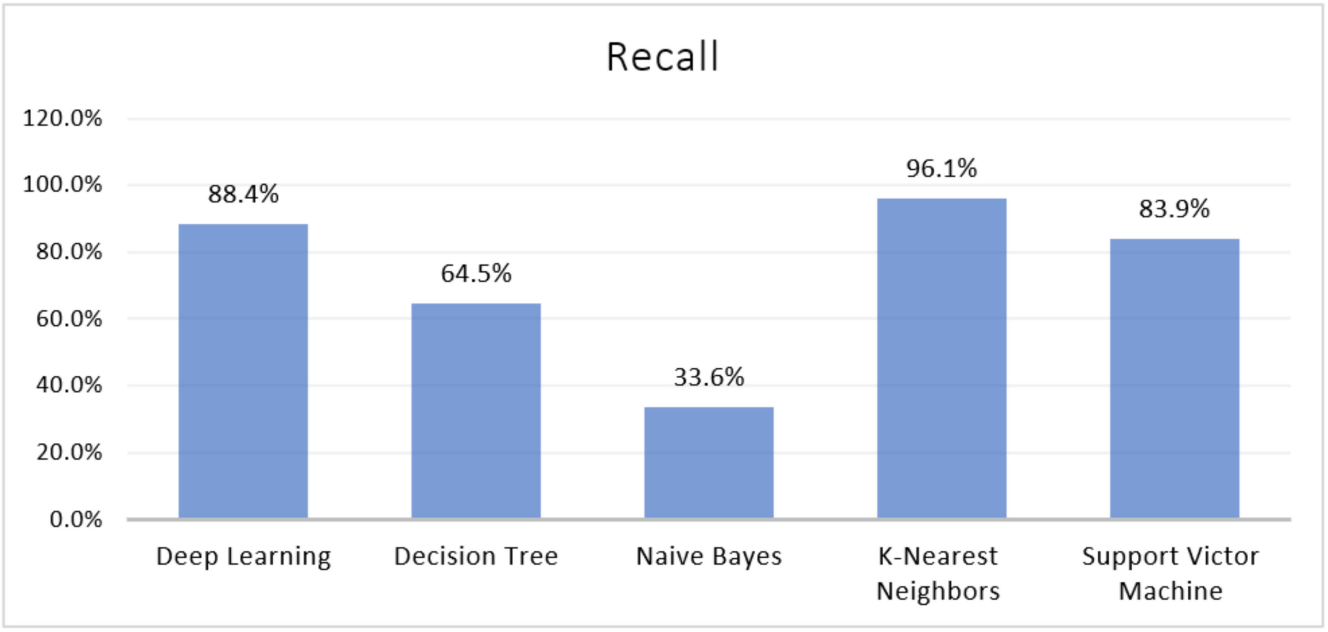


**Figure 5.** Recall Result for Five Machine Learning Algorithms.

### *4.4. F-Measure*

Figure 6 illustrates various accuracies of deep learning, decision tree, Naïve bays, K-NN, and SVM classifiers. It appears that deep learning shows the better result of F-measure is core (86.81%) compared to a decision tree, Naïve Bayes, K-NN, and SVMs. However, SVM also showed a slightly better result, which is 83.87% better than decision tree (77.36%), K-NN (73.32%), and Naïve Bayes (46.681%).

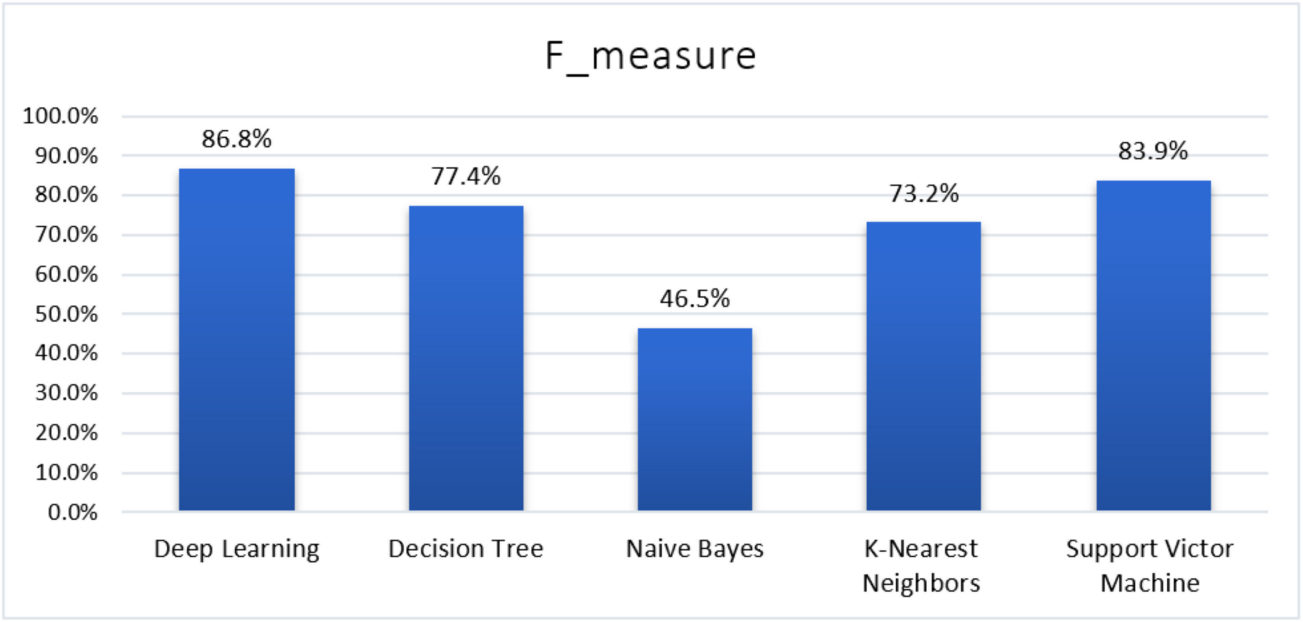


**Figure 6.** F-Measure Result for Five Machine Learning Algorithms.

### *4.5. Correct and Wrong Detection*

Table 3 illustrates the correct detection of deep learning, decision tree, Naïve Bays, K-NN, and SVM classifiers. It indicates that deep learning shows the highest accurate detections (9929) and lowest wrong detections (1718) compared to a decision tree, Naïve Bayes, K-NN, and SVMs. The SVM also has shown slightly higher accurate detection (9585) and low wrong detections (2062). However, the result is better than the decision tree's correct detections (9237) and low incorrect detections (2410), K-NN's accurate detections (7153) and low wrong detections (4494), and Naïve Bayes' accurate detections (6704) and low wrong detections (4943).

**Table 3.** Correct and Wrong Detection.

| | DL | DT | NB | K-NN | SVM |
|---|---|---|---|---|---|
| Correct Detection | 9929 | 9237 | 6704 | 7153 | 9585 |
| Wrong Detection | 1718 | 2410 | 4943 | 4494 | 2062 |
| Total | 11647 | 11647 | 11647 | 11647 | 11647 |

### *4.6. Classification Error*

Figure 7 illustrates various Classification errors of deep learning, decision tree, Naïve Bayes, K-NN, and SVM algorithms. It indicates that deep learning shows the lowest classification error, which is 14.75%, compared to a decision tree, Naïve Bayes, K-NN, and SVM. However, SVM also has shown a slightly lower classification error, which is 17.70%. However, the result is better than the decision tree, with a classification error of 20.69%; the K-NN classification error of 38.59%; and Naïve Bayes, with a higher classification error of 42.44%.

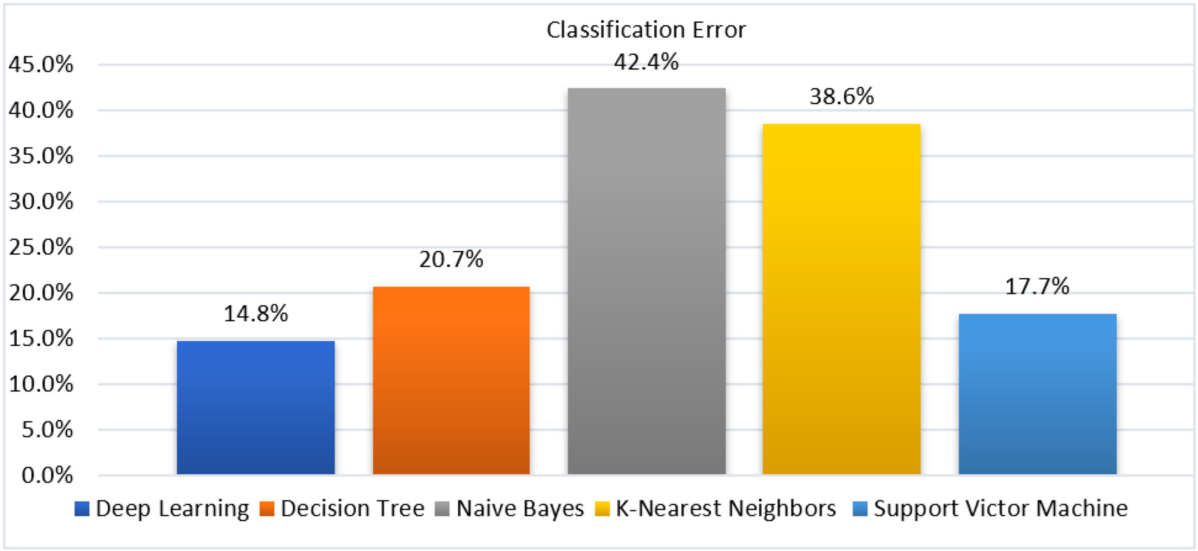


**Figure 7.** Classification Error Result for Five Machine Learning Algorithms.

### *4.7. CPU Time*

Figure 8 illustrates various time complexity of deep learning, decision tree, Naïve bays, K-NN, and SVM classifiers. It appears that SVM is faster with core 897 ms compared to K-NN, Naïve Bayes, decision tree, and deep learning. However, k-nearest neighbours showed a slightly better result, which is (1690 ms) better than decision tree (3870 ms), deep learning (8464 ms), and Naïve Bayes (1800 ms).

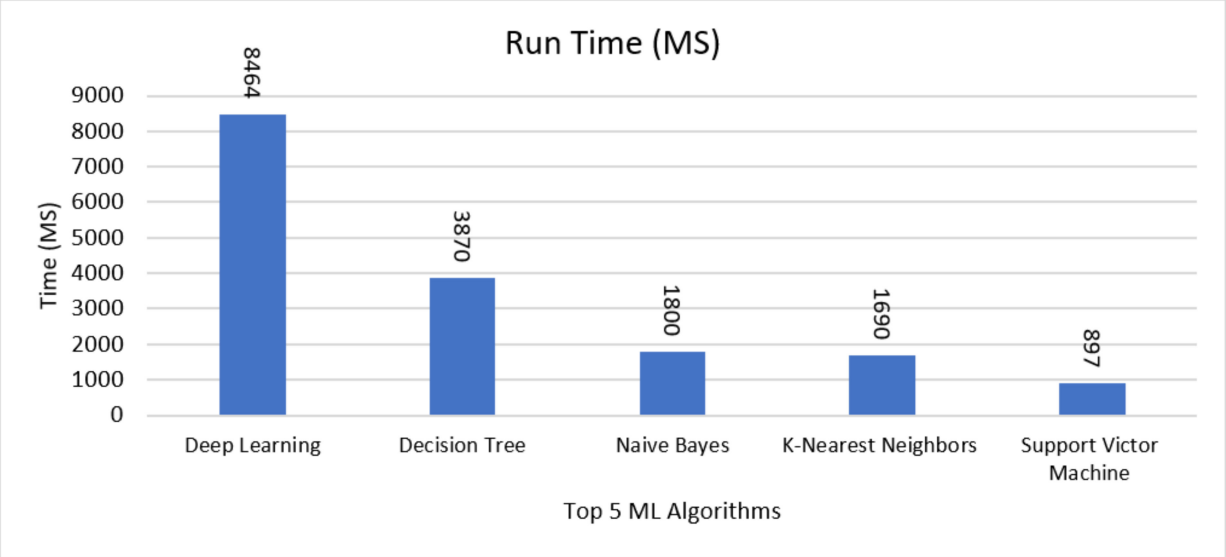


**Figure 8.** Runtime Result for Five Machine Learning Algorithms.

### *4.8. Area under the Curve (AUC)*

Figure 9 presents a graph of the various area under the curve results for the five classifiers, namely, deep learning, decision tree, Naïve Bayes, K-NN, and SVM classifiers. It appears that deep learning shows a better result of AUC as core 0.928 compared to a decision tree, Naïve Bayes, K-NN, and SVMs. However, SVM also showed a slightly better result, which is 0.900 better than the decision tree (0.878), Naïve Bayes (0.610), and K-NN (0.500).

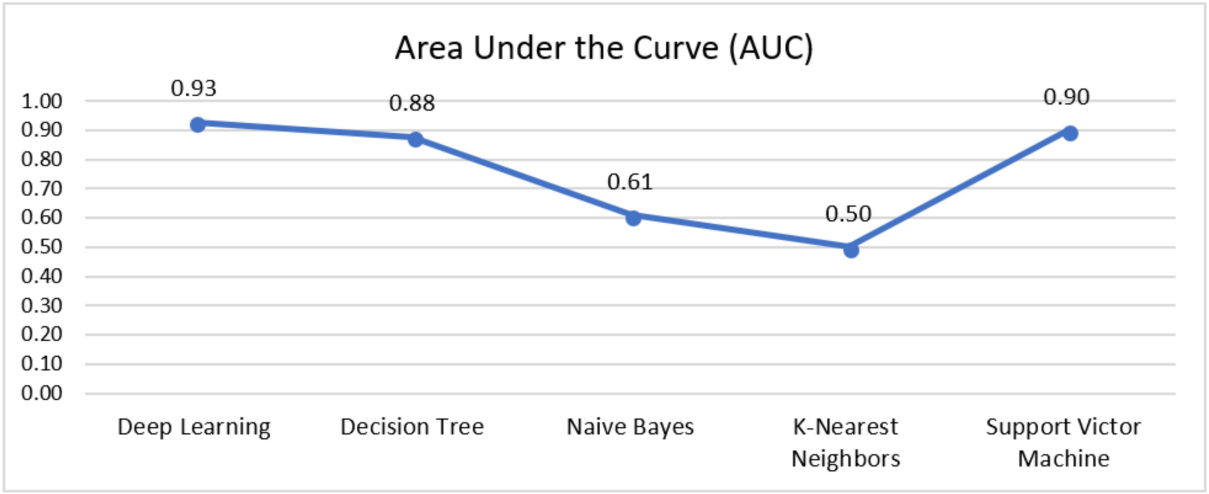


**Figure 9.** Area Under the Curve (AUC) Result for Five Machine Learning Algorithms.

### *4.9. Overall Result*

In this section, we discuss the overall performance result for the top five machine learning classifiers, such as deep learning, decision tree, Naïve Bayes, K-nearest neighbours, and SVM. Generally, Table 4 and Figure 10 consists of the dialect Arabic dataset, in which the performance of the classifiers involved are dependent on the multi-criteria evaluating strategy as well as the dataset. For example, decision tree precision reached 96.67%, and K-Nearest neighbours recall reached 96.67%. However, the other metrics are low for decision trees and k-nearest neighbours. Moreover, the classification error for k-nearest neighbours reached 38.59%, and the decision tree reached 20.69% for classification error. Moreover,

Naïve Bayes obtained the lowest result for the recall, which is 33.59%, and the higher result is 75.49% for precision, and also the classification error and the runtime is slightly high, which is near to 2000 MS compared to other classifier metrics. The accuracy for the decision tree, K-nearest neighbours, and Naïve Bayes are 79.31%, 61.41%, and 57.56%, respectively.

**Table 4.** Summary of the Overall Performance of Machine Learning Classifiers.

| Model | Acc | P | R | Fm | CE | AUC | RT |
|---|---|---|---|---|---|---|---|
| Deep Learning | 85.25% | 85.30% | 88.41% | 86.81% | 14.75% | 0.928 | 8464 |
| Decision Tree | 79.31% | 96.67% | 64.52% | 77.36% | 20.69% | 0.878 | 3870 |
| Naïve Bayes | 57.56% | 75.49% | 33.59% | 46.48% | 42.44% | 0.610 | 1800 |
| K-Nearest Neighbours | 61.41% | 59.14% | 96.14% | 73.23% | 38.59% | 0.500 | 1690 |
| Support Victor Machine | 82.30% | 83.87% | 83.89% | 83.87% | 17.70% | 0.900 | 897 |

Acc = Accuracy, P = Precision, R = Recall, Fm = F-measure, CE = Classification Error, AUC = Area Under the Curve, RT = Run Time.

On the other hand, deep learning reached a higher result for recall (88.41%) compared with other metrics: F-measure (86.81%), precision (85.30%), and accuracy (85.25%). Classification error shows better results compared to another classifier, which is 14.75%. However, the runtime is shown as 3464 MS, which needs improvement. Meanwhile, the support vector machine classifier is demonstrating a better runtime with 897 MS. Furthermore, other metrics are found within the range of 90 in the area under the curve and show a slightly higher classification error of 17.70% compared to a decision tree, K-nearest neighbours, and Naïve Bayes classifier.

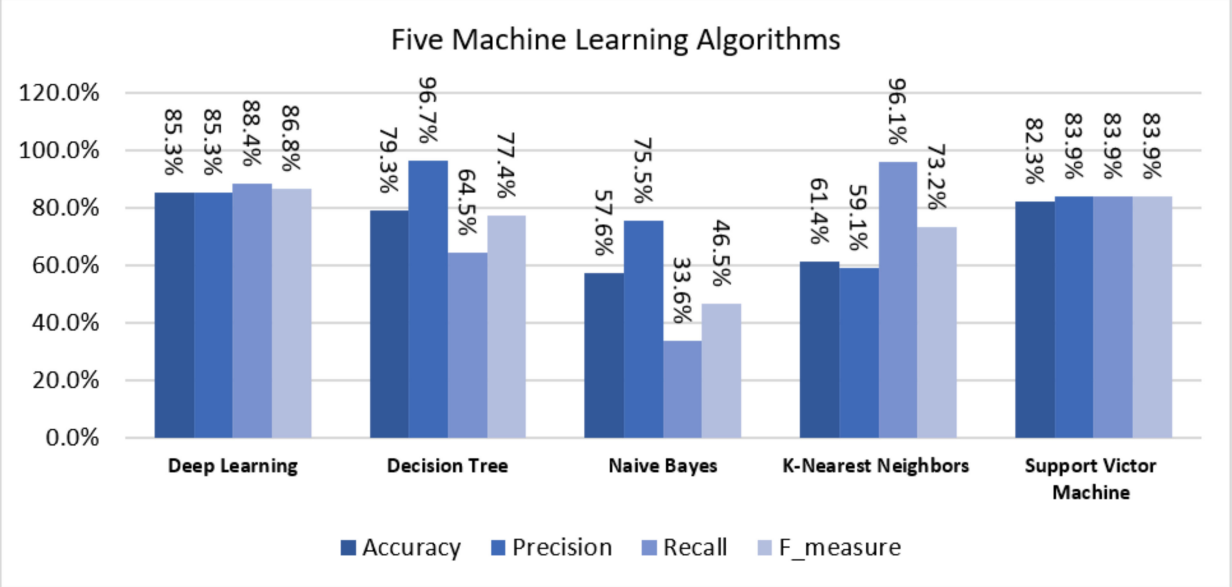


**Figure 10.** Overall Result for Five Machine Learning Algorithms.

## 5. Conclusion and Future Work

In this study, dialect Arabic data from Twitter was used. Moreover, the original dataset was used with AYLIEN tools to label as positive and negative. The dataset was pre-processed by performing data selection, data cleaning, and construction before being used for modelling. After testing the top five classifiers deep learning, decision tree, K-nearest neighbours, Naïve Bayes, and support vector machine, the result of the experiment shows that deep learning and support vector machine classifier showed better performance results in terms of accuracy, precision, recall, F-measure, and AUC compared to a decision tree, K-nearest neighbours, and Naïve Bayes classifiers. However, this does not imply that

deep learning and support vector machine classifiers would always be the best algorithms, as some algorithms appeared to be promising using different datasets; this is because the dataset differs considering the location and the population of the data that was collected. In this study, we further compared the performance of five machine learning algorithms classifier: deep learning, decision tree, Naïve Bayes, K-nearest neighbours, and support vector machine with multi-criteria using dialect Arabic dataset of tweets, to select a better classifier for dialect Arabic. This study can be used to aid novice and prominent researchers in comprehending the theories and concepts of ML algorithms, performance measures, and the sentiment analysis of dialect Arabic texts for a sustainable solution to its morphologically oriented challenges. As for future works, the study highlighted three feasible future research directions that can lead to the production of useful studies related to the Arabic dialect sentiment domain. They include: (i) Using different dialect Arabic training and testing datasets; (ii) using more than the top five machine learning classification algorithms on the same data sources; and (iii) mixing the Arabic dialect dataset with modern standard Arabic dataset to generalise the outcome.

**Author Contributions:** Data curation, M.E.M.A.; Formal analysis, M.E.M.A.; Methodology, M.E.M.A.; Resources, I.A.T.H. and J.Z.M.; Software, M.E.M.A. and U.N.; Supervision, N.I., R.M., A.Q. and S.Y.; Validation, M.E.M.A.; Visualization, M.E.M.A. and S.K.K.; Writing—original draft, M.E.M.A.; Writing—review & editing, M.E.M.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work is supported in part by Universiti Malaya Research University Grant-Faculty Program No GPF091A-2020 and in part by Universiti Brunei Darussalam under research grant UBD/RSCH/URC/RG(b)/2020/023.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

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

## Abbreviations

| | |
|---|---|
| ML | Machine Learning |
| SA | Sentiment Analysis |
| CPU | Central Processing Unit |
| AUC | Area Under the Curve |
| SVM | Support Vector Machine |
| K-NN | K-Nearest Neighbours |
| MSA | Modern Standard Arabic |
| CA | Classical Arabic |
| DA | Dialectical Arabic |
| MRL | Morphologically Rich Language |
| NB | Naïve Bayes |
| DT | Decision Tree |
| MSA | Modern Standard Arabic |
| PPV | Positive Predictive Value |

## References

1. Michalski, R.S.; Carbonell, J.G.; Mitchell, T.M. *Machine Learning: An Artificial Intelligence Approach*; Springer Science & Business Media: Berlin, Germany, 2013.
2. Ali, R.; Lee, S.; Chung, T.C. Accurate multi-criteria decision making methodology for recommending machine learning algorithm. *Expert Syst. Appl.* **2017**, *71*, 257–278. [CrossRef]
3. Zhang, B.; Xu, X.; Li, X.; Chen, X.; Ye, Y.; Wang, Z. Sentiment Analysis through Critic Learning for Optimizing Convolutional Neural Networks with Rules. *Neurocomputing* **2019**, *356*, 21–30. [CrossRef]

4. Poria, S.; Cambria, E.; Bajpai, R.; Hussain, A. A review of affective computing: From unimodal analysis to multimodal fusion. *Inf. Fusion* **2017**, *37*, 98–125. [CrossRef]
5. Goodfellow, I.; Bengio, Y.; Courville, A.; Bengio, Y. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016; Volume 1.
6. Qazi, A.; Raj, R.G.; Hardaker, G.; Standing, C. A systematic literature review on opinion types and sentiment analysis techniques: Tasks and challenges. *Internet Res.* **2017**, *27*, 608–630. [CrossRef]
7. Peng, Y.; Wang, G.; Wang, H. User preferences based software defect detection algorithms selection using MCDM. *Inf. Sci.* **2012**, *191*, 3–13. [CrossRef]
8. Mullainathan, S.; Spiess, J. Machine learning: An applied econometric approach. *J. Econ. Perspect.* **2017**, *31*, 87–106. [CrossRef]
9. Szeliski, R. *Computer Vision: Algorithms and Applications*; Springer Science & Business Media: Berlin, Germany, 2010.
10. Eiland, E.E. *A Coherent Classifier/Prediction/Diagnostic Problem Framework and Relevant Summary Statistics*; New Mexico Institute of Mining and Technology: Socorro, NM, USA, 2017.
11. Qazi, A.; Raj, R.G.; Tahir, M.; Cambria, E.; Syed, K.B.S. Enhancing business intelligence by means of suggestive reviews. *Sci. World J.* **2014**, *2014*, 879323. [CrossRef]
12. De la Paz-Marín, M.; Gutiérrez, P.A.; Hervás-Martínez, C. Classification of countries' progress toward a knowledge economy based on machine learning classification techniques. *Expert Syst. Appl.* **2015**, *42*, 562–572. [CrossRef]
13. Odeh, A.; Abu-Errub, A.; Shambour, Q.; Turab, N. Arabic text categorization algorithm using vector evaluation method. *arXiv* **2015**, arXiv:1501.01318. [CrossRef]
14. Abo, M.E.M.; Raj, R.G.; Qazi, A. A Review on Arabic Sentiment Analysis: State-of-the-Art, Taxonomy and Open Research Challenges. *IEEE Access* **2019**, *7*, 162008–162024. [CrossRef]
15. Khasawneh, R.T.; Wahsheh, H.A.; Al-Kabi, M.N.; Alsmadi, I.M. Sentiment analysis of arabic social media content: A comparative study. In Proceedings of the 8th International Conference for Internet Technology and Secured Transactions (ICITST-2013), London, UK, 9–12 December 2013; pp. 101–106.
16. Duwairi, R.M.; Alfaqeh, M.; Wardat, M.; Alrabadi, A. Sentiment analysis for Arabizi text. In Proceedings of the 2016 7th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 5–7 April 2016; pp. 127–132.
17. Han, W.; Jiang, Y.; Tu, K. Lexicalized Neural Unsupervised Dependency Parsing. *Neurocomputing* **2019**, *349*, 105–115. [CrossRef]
18. Guellil, I.; Adeel, A.; Azouaou, F.; Benali, F.; Hachani, A.-E.; Hussain, A. Arabizi sentiment analysis based on transliteration and automatic corpus annotation. In Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, Brussels, Belgium, 31 October 2018; pp. 335–341.
19. Abo, M.E.M.; Shah, N.A.K.; Balakrishnan, V.; Kamal, M.; Abdelaziz, A.; Haruna, K. SSA-SDA: Subjectivity and Sentiment Analysis of Sudanese Dialect Arabic. In Proceedings of the 2019 International Conference on Computer and Information Sciences (ICCIS), Sakaka, Saudi Arabia, 3–4 April 2019; pp. 1–5.
20. Kotthoff, L. Algorithm Selection for Combinatorial Search Problems: A Survey. In *Transactions on Petri Nets and Other Models of Concurrency XV*; Springer Science and Business Media LLC: Berlin, Germany, 2016; pp. 149–190.
21. Govindan, K.; Rajendran, S.; Sarkis, J.; Murugesan, P. Multi criteria decision making approaches for green supplier evaluation and selection: A literature review. *J. Clean. Prod.* **2015**, *98*, 66–83. [CrossRef]
22. Chandrashekar, G.; Sahin, F. A survey on feature selection methods. *Comput. Electr. Eng.* **2014**, *40*, 16–28. [CrossRef]
23. Chang, P.-L.; Chen, Y.-C. A fuzzy multi-criteria decision making method for technology transfer strategy selection in biotechnology. *Fuzzy Sets Syst.* **1994**, *63*, 131–139. [CrossRef]
24. DeFries, R.; Chan, J.C.-W. Multiple criteria for evaluating machine learning algorithms for land cover classification from satellite data. *Remote. Sens. Environ.* **2000**, *74*, 503–515. [CrossRef]
25. Kononenko, I. Machine learning for medical diagnosis: History, state of the art and perspective. *Artif. Intell. Med.* **2001**, *23*, 89–109. [CrossRef]
26. Wang, Z.; Lemmon, M. Stability analysis of weak rural electrification microgrids with droop-controlled rotational and electronic distributed generators. In Proceedings of the 2015 IEEE Power & Energy Society General Meeting, Denver, CO, USA, 26–30 July 2015; pp. 1–5.
27. Wegrzyn-Wolska, K.; Bougueroua, L.; Dziczkowski, G. Social media analysis for e-health and medical purposes. In Proceedings of the 2011 International Conference on Computational Aspects of Social Networks (CASoN), Salamanca, Spain, 19–21 October 2011; pp. 278–283.
28. Salameh, M.; Mohammad, S.; Kiritchenko, S. Sentiment after Translation: A Case-Study on Arabic Social Media Posts. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, CO, USA, 31 May–4 June 2015; pp. 767–777.
29. Sghaier, M.A.; Zrigui, M. Sentiment analysis for Arabic e-commerce websites. In Proceedings of the 2016 International Conference on Engineering & MIS (ICEMIS), Agadir, Morocco, 22–24 September 2016; pp. 1–7.
30. Alayba, A.M.; Palade, V.; England, M.; Iqbal, R. Arabic language sentiment analysis on health services. In Proceedings of the 2017 1st International Workshop on Arabic Script Analysis and Recognition (ASAR), Nancy, France, 3–5 April 2017; pp. 114–118.
31. Duwairi, R.M. Sentiment analysis for dialectical Arabic. In Proceedings of the 2015 6th International Conference on Information and Communication Systems (ICICS), Amman, Jordan, 7–9 April 2015; pp. 166–170.

32. Hathlian, N.F.B.; Hafezs, A.M. Sentiment—Subjective analysis framework for arabic social media posts. In Proceedings of the 2016 4th Saudi International Conference on Information Technology (Big Data Analysis), Riyadh, Saudi Arabia, 6–9 November 2016; pp. 1–6.
33. Abdulkareem, M.; Tiun, S. Comparative analysis of ML POS on Arabic tweets. *J. Theor. Appl. Inf. Technol.* **2017**, *95*, 403.
34. Alqarafi, A.; Adeel, A.; Hawalah, A.; Swingler, K.; Hussain, A. A Semi-supervised Corpus Annotation for Saudi Sentiment Analysis Using Twitter. In Proceedings of the Transactions on Petri Nets and Other Models of Concurrency XV, Xi'an, China, 6 October 2018; pp. 589–596.
35. Cambria, E.; Poria, S.; Hussain, A.; Liu, B. Computational Intelligence for Affective Computing and Sentiment Analysis [Guest Editorial]. *IEEE Comput. Intell. Mag.* **2019**, *14*, 16–17. [CrossRef]
36. AlHumoud, S.; Albuhairi, T.; Altuwaijri, M. Arabic Sentiment Analysis using WEKA a Hybrid Learning Approach. In Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, Lisbon, Portugal, 12–14 November 2015; pp. 402–408.
37. Abo, M.E.M.; Shah, N.A.K.; Balakrishnan, V.; Abdelaziz, A. Sentiment analysis algorithms: Evaluation performance of the Arabic and English language. In Proceedings of the 2018 International Conference on Computer Control, Electrical, and Electronics Engineering (ICCCEEE), Khartoum, Sudan, 12–14 August 2018; pp. 1–5.
38. Alabdullatif, A.; Shahzad, B.; Alwagait, E. Classification of Arabic Twitter Users: A Study Based on User Behaviour and Interests. *Mob. Inf. Syst.* **2016**, *2016*, 8315281. [CrossRef]
39. Hadi, W.E. Classification of Arabic Social Media Data. *Adv. Comput. Sci. Technol.* **2015**, *8*, 29–34.
40. Hamouda, S.B.; Akaichi, J. Social networks' text mining for sentiment classification: The case of Facebook'statuses updates in the 'Arabic Spring'era. *Int. J. Appl. Innov. Eng. Manag.* **2013**, *2*, 470–478.
41. Mountassir, A.; Benbrahim, H.; Berrada, I. Some methods to address the problem of unbalanced sentiment classification in an arabic context. In Proceedings of the 2012 Colloquium in Information Science and Technology, Fez, Morocco, 22–24 October 2012; pp. 43–48.
42. Ahmed, S.; Pasquier, M.; Qadah, G.Z. Key issues in conducting sentiment analysis on Arabic social media text. In Proceedings of the 2013 9th International Conference on Innovations in Information Technology (IIT), Al Ain, United Arab Emirates, 17–19 March 2013; pp. 72–77.
43. Abdul-Mageed, M.; Diab, M.T.; Korayem, M. Subjectivity and sentiment analysis of modern standard Arabic. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short papers-Volume 2, Portland, OR, USA, 19 June 2011; pp. 587–591.
44. Al-Kabi, M.N.; Abdulla, N.A.; Al-Ayyoub, M. An analytical study of Arabic sentiments: Maktoob case study. In Proceedings of the 8th International Conference for Internet Technology and Secured Transactions (ICITST-2013), London, UK, 9–12 December 2013; pp. 89–94.
45. Duwairi, R.M.; Marji, R.; Sha'Ban, N.; Rushaidat, S. Sentiment Analysis in Arabic tweets. In Proceedings of the 2013 IEEE International Conference on Computer Vision Workshops, Irbid, Jordan, 1–3 April 2014; pp. 1–6.
46. Abdulla, N.A.; Ahmed, N.A.; Shehab, M.A.; Al-Ayyoub, M. Arabic sentiment analysis: Lexicon-based and corpus-based. In Proceedings of the 2013 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT), Amman, Jordan, 3–5 December 2013; pp. 1–6.
47. Alqasemi, F.; Abdelwahab, A.; Abdelkader, H. An enhanced feature extraction technique for improving sentiment analysis in Arabic language. In Proceedings of the 2016 4th IEEE International Colloquium on Information Science and Technology, Tangier, Morocco, 24–26 October 2016; pp. 380–384. [CrossRef]
48. Al Sallab, A.A.; Baly, R.; Badaro, G.; Hajj, H.; El Hajj, W.; Shaban, K.B. Deep learning models for sentiment analysis in Arabic. In Proceedings of the Proceedings of the Second Workshop on Arabic Natural Language Processing, Beijing, China, 26–31 July 2015; p. 9.
49. Altawaier, M.; Tiun, S. Comparison of Machine Learning Approaches on Arabic Twitter Sentiment Analysis. *Int. J. Adv. Sci. Eng. Inf. Technol.* **2016**, *6*, 1067. [CrossRef]
50. Al-Rubaiee, H.; Qiu, R.; Li, D. Identifying Mubasher software products through sentiment analysis of Arabic tweets. In Proceedings of the 2016 International Conference on Industrial Informatics and Computer Systems (CIICS), Sharjah, United Arab Emirates, 13–15 March 2016; pp. 1–6.
51. Alotaibi, S.S.; Anderson, C.W. Extending the knowledge of the arabic sentiment classification using aforeign external lexical source. *Int. J. Nat. Lang. Comput.* **2016**, *5*, 1–11. [CrossRef]
52. Shoukry, A.; Rafea, A. Sentence-level Arabic sentiment analysis. In Proceedings of the 2012 International Conference on Collaboration Technologies and Systems (CTS), Denver, CO, USA, 21–25 May 2012; pp. 546–550.
53. Alhumoud, S.; Albuhairi, T.; Alohaideb, W. Hybrid Sentiment Analyser for Arabic Tweets using R. In Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, Lisbon, Portugal, 12–14 November 2015; pp. 417–424. [CrossRef]
54. Duwairi, R.; El-Orfali, M. A study of the effects of preprocessing strategies on sentiment analysis for Arabic text. *J. Inf. Sci.* **2014**, *40*, 501–513. [CrossRef]
55. Alotaibi, S.; Anderson, C. Word Clustering as a Feature for Arabic Sentiment Classification. *Int. J. Educ. Manag. Eng.* **2017**, *7*, 1–13. [CrossRef]

56. Al-Moslmi, T.; Albared, M.; Al-Shabi, A.; Omar, N.; Abdullah, S. Arabic senti-lexicon: Constructing publicly available language resources for Arabic sentiment analysis. *J. Inf. Sci.* **2017**, *44*, 345–362. [CrossRef]
57. Refaee, E. Sentiment Analysis for Micro-blogging Platforms in Arabic. In Proceedings of the Transactions on Petri Nets and Other Models of Concurrency XV, Vancouver, BC, Canada, 9–14 July 2017; pp. 275–294.
58. Al-Ayyoub, M.; Nuseir, A.; Kanaan, G.; Al-Shalabi, R. Hierarchical Classifiers for Multi-Way Sentiment Analysis of Arabic Reviews. *Int. J. Adv. Comput. Sci. Appl.* **2016**, *7*, 531–539. [CrossRef]
59. Tobaili, T.; He, H.; Lei, T.; Roberts, W. Arabizi Identification in Twitter Data. In Proceedings of the Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics—Student Research Workshop, Berlin, Germany, 7–12 August 2016; pp. 51–57. [CrossRef]
60. Al-Twairesh, N.; Al-Khalifa, H.; Alsalman, A.; Erk, K.; Smith, N.A. AraSenTi: Large-Scale Twitter-Specific Arabic Sentiment Lexicons. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, 7–12 August 2016; pp. 697–705.
61. Valdivia, A.; Hrabova, E.; Chaturvedi, I.; Luzón, M.V.; Troiano, L.; Cambria, E.; Herrera, F. Inconsistencies on TripAdvisor reviews: A unified index between users and Sentiment Analysis Methods. *Neurocomputing* **2019**, *353*, 3–16. [CrossRef]
62. Pasha, A.; Al-Badrashiny, M.; Diab, M.T.; El Kholy, A.; Eskander, R.; Habash, N.; Pooleery, M.; Rambow, O.; Roth, R. MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In Proceedings of the Proceedings of the 9th International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, 26–31 May 2014; pp. 1094–1101.
63. Al-Smadi, M.; Qawasmeh, O.; Al-Ayyoub, M.; Jararweh, Y.; Gupta, B. Deep Recurrent neural network vs. support vector machine for aspect-based sentiment analysis of Arabic hotels' reviews. *J. Comput. Sci.* **2018**, *27*, 386–393. [CrossRef]
64. Abuata, B.; Al-Omari, A. A rule-based stemmer for Arabic Gulf dialect. *J. King Saud Univ. -Comput. Inf. Sci.* **2015**, *27*, 104–112. [CrossRef]
65. Mostafa, M.M. More than words: Social networks' text mining for consumer brand sentiments. *Expert Syst. Appl.* **2013**, *40*, 4241–4251. [CrossRef]
66. Stephens, Z.D.; Lee, S.Y.; Faghri, F.; Campbell, R.H.; Zhai, C.; Efron, M.J.; Iyer, R.; Schatz, M.C.; Sinha, S.; Robinson, G.E. Big Data: Astronomical or Genomical? *PLoS Biol.* **2015**, *13*, e1002195. [CrossRef]
67. Acharya, U.R.; Fujita, H.; Lih, O.S.; Adam, M.; Tan, J.H.; Chua, C.K. Automated detection of coronary artery disease using different durations of ECG segments with convolutional neural network. *Knowl.-Based Syst.* **2017**, *132*, 62–71. [CrossRef]
68. Singh, A.; Ganapathysubramanian, B.; Singh, A.K.; Sarkar, S. Machine learning for high-throughput stress phenotyping in plants. *Trends Plant Sci.* **2016**, *21*, 110–124. [CrossRef] [PubMed]
69. Ching, T.; Himmelstein, D.S.; Beaulieu-Jones, B.K.; Kalinin, A.A.; Do, B.T.; Way, G.P.; Ferrero, E.; Agapow, P.-M.; Zietz, M.; Hoffman, M.M.; et al. Opportunities and obstacles for deep learning in biology and medicine. *J. R. Soc. Interface* **2018**, *15*, 20170387. [CrossRef] [PubMed]
70. Catal, C.; Nangir, M. A sentiment classification model based on multiple classifiers. *Appl. Soft Comput.* **2017**, *50*, 135–141. [CrossRef]
71. Al-Batah, M.S.; Mrayyen, S.; Alzaqebah, M. Arabic Sentiment Classification using MLP Network Hybrid with Naive Bayes Algorithm. *J. Comput. Sci.* **2018**, *14*, 1104–1114. [CrossRef]
72. Xiong, S.; Lv, H.; Zhao, W.; Ji, D. Towards Twitter sentiment classification by multi-level sentiment-enriched word embeddings. *Neurocomputing* **2018**, *275*, 2459–2466. [CrossRef]
73. Tian, Y.; Qi, Z.; Ju, X.; Shi, Y.; Liu, X. Nonparallel Support Vector Machines for Pattern Classification. *IEEE Trans. Cybern.* **2014**, *44*, 1067–1079. [CrossRef]
74. Mohammad, A.H.; Alwada'n, T.; Al-Momani, O. Arabic text categorization using support vector machine, Naïve Bayes and neural network. *GSTF J. Comput.* **2018**, *5*, 1–8. [CrossRef]
75. Salloum, S.A.; AlHamad, A.Q.; Al-Emran, M.; Shaalan, K. A Survey of Arabic Text Mining. In *Intelligent Natural Language Processing: Trends and Applications*; Humana Press: Totowa, NJ, USA, 2018; pp. 417–431.
76. Tang, D.; Wei, F.; Qin, B.; Liu, T.; Zhou, M. Coooolll: A Deep Learning System for Twitter Sentiment Classification. In Proceedings of the Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland, 23–24 August 2014; pp. 208–212.
77. Lu, H.; Jin, L.; Luo, X.; Liao, B.; Guo, D.; Xiao, L. RNN for Solving Perturbed Time-Varying Underdetermined Linear System with Double Bound Limits on Residual Errors and State Variables. *IEEE Trans. Ind. Inform.* **2019**, *15*, 5931–5942. [CrossRef]
78. Wu, D.; Luo, X.; Shang, M.; He, Y.; Wang, G.; Zhou, M. A Deep Latent Factor Model for High-Dimensional and Sparse Matrices in Recommender Systems. *IEEE Trans. Syst. Man, Cybern. Syst.* **2021**, *51*, 4285–4296. [CrossRef]
79. Qazi, A.; Tamjidyamcholo, A.; Raj, R.G.; Hardaker, G.; Standing, C. Assessing consumers' satisfaction and expectations through online opinions: Expectation and disconfirmation approach. *Comput. Hum. Behav.* **2017**, *75*, 450–460. [CrossRef]
80. Cano, J.-R.; Gutiérrez, P.A.; Krawczyk, B.; Woźniak, M.; García, S. Monotonic classification: An overview on algorithms, performance measures and data sets. *Neurocomputing* **2019**, *341*, 168–182. [CrossRef]
81. Liu, W.; Wang, Z.; Liu, X.; Zeng, N.; Liu, Y.; Alsaadi, F.E. A survey of deep neural network architectures and their applications. *Neurocomputing* **2017**, *234*, 11–26. [CrossRef]
82. Bravo-Marquez, F.; Mendoza, M.; Poblete, B. Meta-level sentiment models for big social data analysis. *Knowl.-Based Syst.* **2014**, *69*, 86–99. [CrossRef]

83. Powers, D.M. Evaluation: From precision, recall and F-measure to ROC, Informedness, Markedness and correlation. *arXiv* **2020**, arXiv:2010.16061.
84. Yamout, B.; Issa, Z.; Herlopian, A.; El Bejjani, M.; Khalifa, A.; Ghadieh, A.S.; Habib, R.H. Predictors of quality of life among multiple sclerosis patients: A comprehensive analysis. *Eur. J. Neurol.* **2013**, *20*, 756–764. [CrossRef]
85. Abooraig, R.; Alzubi, S.; Kanan, T.; Hawashin, B.; Al Ayoub, M.; Hmeidi, I. Automatic categorization of Arabic articles based on their political orientation. *Digit. Investig.* **2018**, *25*, 24–41. [CrossRef]