

Article

Evaluating Intelligent Methods for Detecting COVID-19 Fake News on Social Media Platforms

Hosam Alhakami ¹, Wajdi Alhakami ², Abdullah Baz ³, Mohd Faizan ^{4,*}, Mohd Waris Khan ⁴ and Alka Agrawal ⁵

¹ Department of Computer Science, College of Computer and Information Systems, Umm Al-Qura University, Makkah 715, Saudi Arabia; hhhakam@uqu.edu.sa

² Department of Information Technology, College of Computers and Information Technology, Taif University, Taif 26571, Saudi Arabia; whakami@tu.edu.sa

³ Department of Computer Engineering, College of Computer and Information Systems, Umm Al-Qura University, Makkah 715, Saudi Arabia; aobaz01@uqu.edu.sa

⁴ Department of Computer Application, Integral University, Lucknow 226026, India; waris.khan070@gmail.com

⁵ Department of Information Technology, Babasaheb Bhimrao Ambedkar University, Lucknow 226025, India; alka_csjmu@yahoo.co.in

* Correspondence: faizanmhd943@gmail.com

Abstract: The advent of Internet-based technology has made daily life much easy than earlier days. The exponential rise in the popularity of social media platforms has not only connected people from faraway places, but has also increased communication among humans. However, in several instances, social media platforms have also been utilized for unethical and criminal activities. The propagation of fake news on social media during the ongoing COVID-19 pandemic has deteriorated the mental and physical health of people. Therefore, to control the flow of fake news regarding the novel coronavirus, several studies have been undertaken to automatically detect the fake news about COVID-19 using various intelligent techniques. However, different studies have shown different results on the performance of the predicting models. In this paper, we have evaluated several machine learning and deep learning models for the automatic detection of fake news regarding COVID-19. The experiments were carried out on two publicly available datasets, and the results were assessed using several evaluation metrics. The traditional machine learning models produced better results than the deep learning models in predicting fake news.

Keywords: fake news; COVID-19; social media; machine learning; deep learning



Citation: Alhakami, H.; Alhakami, W.; Baz, A.; Faizan, M.; Khan, M.W.; Agrawal, A. Evaluating Intelligent Methods for Detecting COVID-19 Fake News on Social Media Platforms. *Electronics* **2022**, *11*, 2417. <https://doi.org/10.3390/electronics11152417>

Academic Editor: Guillermo L. Taboada

Received: 29 June 2022

Accepted: 30 July 2022

Published: 3 August 2022

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



Copyright: © 2022 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

The Internet is now an essential part of modern society and has surpassed the conventional ways of gathering information and knowledge from television and newspapers. Social media platforms have significantly changed the ways in which people receive news and current affairs around the globe. With social networking sites like Facebook and Twitter experiencing unprecedented growth in their user base, society is largely dependent on such platforms for acquiring news and information.

The ease of accessing social media platforms helps people to instantly share ideas and news without checking the legitimacy of the news source. Consequently, social media is being used for spreading misinformation and fake news [1]. Social media is capable of propagating unverified content, creating an ecosystem of misinformation that may increase the likelihood of changing people's perspectives of reality through the spread of fake news.

Fake news is information with false content, spread intentionally to manipulate public perspective on a specific agenda for political, social or economic gains or simply for fun. A concerning aspect of fake news is that it draws more public attention than legitimate sources of information [2]. Moreover, fake news propagates much faster and penetrates deep into

people's minds to influence their perception than legitimate news does [1]. Consequently, people usually accept and forward such news content without inspecting the authenticity of the source of news. This result in an exponential spread of fake news, either intentionally or knowingly, by many people. The repercussions of the spread of fake news are varied, with grave consequences such as compromised decision-making, cyberbullying, social hostility and violence.

The alarming impact of the spread of false news has come into the limelight in the current COVID-19 pandemic. The ongoing pandemic is considered one of the biggest current public health emergencies, affecting millions of humans in a number of ways [3]. Throughout the pandemic, fake news has been circulating on digital platforms that has created panic among the public. Governments have had to step in to verify the authenticity of news, in order to alleviate panic and fear in the population and to maintain law and order. The pandemic has already disrupted human lives in unprecedented ways. The misinformation regarding COVID-19 has prompted many individuals to commit suicide after being diagnosed with the deadly virus [4]. As COVID-19 is still a pandemic across many countries of the world, it would be necessary to work on controlling fake news about COVID-19 from being spread on social media platforms. Therefore, researchers are striving to control this menace by developing intelligent and automated techniques for identifying fake news content. To achieve this task, machine learning techniques are being applied to predict fake news content.

Machine learning (ML) techniques enable a computer system to automatically learn patterns and relationships from input data, using various algorithms to solve problems without being specifically programmed. Deep learning (DL) is a specialized type of ML employing artificial neural networks that imitates the human brain having multiple layers of neurons to solve complex problems. DL models are known to outperform ML models in solving several complex problems [5].

The labeled data can be categorized using a supervised machine learning technique called classification [6]. The classification technique can be used to classify the information as either true or false given a labeled dataset [7]. Many researchers have made efforts to improve the classification accuracy on several datasets, however, more analysis is required to evaluate the effectiveness of the classification models in detecting the fake news content. This problem can be expressed as a text classification problem [8,9].

The identification of COVID-19 fake news is a binary classification problem. The present work intends to search the best classification model among several candidates for predicting fake news on COVID-19. The conventional ML models and recent DL models for text classification are evaluated for this task.

The main contributions are: The evaluation of six ML classifiers and two DL classifiers for the identification of fake news on two publicly available datasets. Secondly, comparing the best performing ML models with the DL models.

The rest of the paper is organized as follows: Section 2 explore the related work on fake news prediction. Section 3 describes the various classifiers used in this work. Section 4 discusses the experimental setup. Section 5 contains the results and discussion. Finally, Section 6 contains conclusion and future work.

2. Related Work

This section will discuss several studies on the prediction of fake news. A study was specifically undertaken to discuss intelligent techniques for the identification of clickbait. The authors discussed the relevance of combining textual features like the semantics of news content, and non-textual features like graphics and user behavior for detecting fake news content [10]. Another study on clickbait detection utilizes the clickbait headlines. The researchers have performed the experiments on a public dataset of fake news, and obtained accuracy of 89.59 percent on the logistic regression learning model [11]. The credibility of news was also checked using text analysis techniques, where different features (such as user profiles) and content-specific attributes (such as comments, replies, external links etc.) were

used. The authors employed Naïve Bayes, Random Forest and decision tree classifiers to implement their techniques [12]. Similarly, another work on fake news detection used four types of features: textual, user profile-based, content-specific and message propagation. These features were extracted from a Twitter dataset, and the models were trained to achieve a precision of around 70 to 80 percent [13].

The stylometry features such as writing style and grammar used in the news content were utilized to label the information as true or false [14]. The experimental results on several datasets have shown that the use of such features could provide better results than using simple textual features. Linguistic features have been extensively used in several works for the identification of misleading content. Sixteen linguistic features divided into four categories were used in a study, and the decision tree classifier achieved 60 percent accuracy on the selected features [15]. Similarly, several classifiers were used on various linguistic features from the Facebook dataset, and more than 90 percent accuracy was obtained by the linear regression model [16].

In another attempt to classify Twitter posts as either fake or real, the authors created four categories of features that contain 45 different features [17]. The selected features represent different attributes of the Twitter posts, such as the length and polarity of the posts, user characteristics and other content-based features. The models were trained on the PHEME and CREDBANK datasets, while testing was performed on the BuzzFeed dataset. The model achieved 65.29 percent of accuracy on the test dataset. Similarly, in another study, the authors used 29 features for training the logistic regression classifier that obtained an accuracy of around 60 percent [18]. Zhou et al. [19] have proposed nine classes containing 20 features for identifying fake news content. Gravanis et al. [20] have introduced a new dataset called UNBiased, which contained 3004 instances by taking samples from four publicly available datasets. They achieved 95 percent accuracy on their dataset using the support vector machine classifier. Another study [21] introduced a deep learning model for the identification of false information. The Kaggle dataset was utilized for the experiment, and GloVe vectors were used for embedding textual features. The author proposed a hybrid model based on convolutional and recurrent neural networks.

In a recent study, the authors explored several embedding models for detection of Arabic fake news. The authors developed transformer-based classifiers for the task. They constructed a dataset of fake news in the Arabic language for evaluating their proposed models, and achieved an accuracy score of more than 98 percent [22].

The existing work discussed in the preceding section indicates that no single methodology or classification model can provide the best results. No single model can claim to always perform better for every problem. Therefore, in this work, we have tried to figure out the performance of several models and their parameter tuning for the prediction of news containing false information about COVID-19. The proposed work is different from the study [22] discussed above in the sense that it evaluates eight DL classifiers for detecting fake news in the Arabic language. Whereas, our work evaluates six ML and two DL classifiers for the same task in the English language.

3. Description of Classifiers

Various ML and DL classification algorithms that shall be used are discussed below.

3.1. Naïve Bayes

Naïve Bayes (NB) applies the Bayes theorem from the field of probability. This classifier assumes that a change in one feature does not have an affect on the other feature i.e., all the features are conditionally independent [23]. It calculates the probability for an event to occur by obtaining joint probability to the happening of another event. The NB classifier is easy to implement and can be economically trained with good classification results.

3.2. Logistic Regression

Logistic Regression is a linear classification algorithm borrowed from the field of statistics. It passes the input data through the sigmoid function, which returns a probability value that can be used to classify a particular data point to its corresponding class [23]. A decision boundary or a threshold is computed for mapping the data points to their respective classes, depending on the value of the sigmoid function. It is based upon the assumption that the likelihood of mapping a sample to a particular class can be obtained from the linear combination of the features of that sample.

3.3. Support Vector Machines

A Support Vector Machine (SVM) [23] performs a classification by computing a decision hyperplane that demarcates samples from various classes. The hyperplane is positioned in the hyperspace such that the distance between the different classes is maximized. As the SVM does not require the training examples to be transformed into different spaces, it can handle a very large feature subset. Since the SVM is one of the best-performing classifiers with a good classification performance [24], it has been used in several existing studies.

3.4. Decision Tree

A Decision Tree (DT) is among the most popular ML models for text classification [23]. A decision tree classifier is represented as a tree data structure. The leaf node of the tree represents the label of a target class, and the remaining internal nodes indicate the test to be applied on a single feature, with a sub tree and a branch for every result of the test. The sample of the dataset is classified by moving through the tree, with the root as the initial point down to the leaf node that specifies the target class of the sample.

3.5. Random Forests

Random Forests (RF) is an ensemble of multiple decision trees that can be used for both classification and regression [23]. It applies the voting method to produce the outcome. It gives the class label to the sample as having the maximum number of votes. A random subset of features is used to split the leaf node for each of the decision trees. This procedure is replicated to generate the K number of DT. The class of the instance is specified that it has a majority of the vote from the decision trees.

3.6. K-Nearest Neighbor

The k-nearest neighbor (KNN) is one of the simplest approaches for classification problems [23]. The KNN model finds the class of a sample by comparing it with the training data closest to it. Each sample of the dataset is represented as a point in an n-dimensional space. All of the training data resides in this space. The class of the test data point is obtained by looking for the k training samples that are neighbors of the test data point in the space. These closest training data points are called the k-nearest neighbors. The nearest neighbors are identified by calculating the distance between the given test data and training data in the space. The class label of the nearest neighbors is specified as the class of the given test data point.

3.7. Convolutional Neural Network (CNN or ConvNet)

CNN is a discriminative supervised DL model that automatically extracts features from the input data [25]. The CNN is an enhancement to the conventional artificial neural networks. Each of the layers of the CNN considers optimal hyper parameter settings to produce acceptable results, while reducing the complexity of the model. The CNN also supports a mechanism to handle the overfitting of the model which may not be found in traditional neural networks [26]. CNNs have been extensively applied in the area of pattern recognition, speech processing and natural language processing.

3.8. Long Short-Term Memory (LSTM)

LSTM [27] is a type of recurrent neural network that can specifically recall patterns for a long time. The recursive cell state in LSTM is long-term memory that can store previous information in them. The cell states are modified by the forget gates placed below them. The output of the forget gate indicates whether the information is to be kept or forgotten. The information is kept if the output of the forget gate is one. The entry of information into the cells is managed by the input gate. The output gate decides the information that needs to be transferred to the next state. Unlike other traditional ML models, LSTM can use multiword strings to determine the class labels, which would be helpful in the text classification task.

4. Experiments

4.1. Description of the Datasets

A dataset on fake news during the pandemic period called the COVID-19 Fake News Dataset [28], is used for analyzing the performance of various classifiers. The dataset was published by Abhishek Koirala [29] and is publicly available on the Mendeley. The dataset consists of 3119 social media posts related to the COVID-19 pandemic, which were collected approximately seven months after the pandemic has started. The posts in the English language were collected across several social media platforms worldwide using Webhose.io. The keywords used for filtering the posts were: corona, coronavirus and COVID-19. The posts were manually labeled as either fake news or true news. There were 2061 posts in the true news category and 1058 in the fake news category.

The second dataset that we used was the Constraint@AAAI 2021 COVID-19 Fake News detection dataset [30]. The dataset consisted of social media posts and their corresponding labels specifying whether the post is real or fake. The dataset contained 10,700 samples, of which 5600 samples were real and the remaining 5100 samples were fake. The authenticity of real posts was manually verified from several fact-checking websites.

4.2. Data Preprocessing

The data preprocessing was performed in three stages, as described below:

- **Tokenization:** Tokenization of the textual data is performed to break long sentences into individual words or numbers called tokens, and delimited by spaces.
- **Noise Removal:** Noise is unwanted data like punctuation marks, special symbols and hyperlinks that need to be removed from the text, as they do not carry any meaning to the model.
- **Removal of Stop words:** The most frequently used words in any language that do not contribute much information to the data are called stop words. The stop words in English are *is, am, are, of, the*, etc. The removal of stop words from the text decreases the dimensionality of the feature space and may also help in enhancing the performance of the classification model.

After applying the three steps of preprocessing, all the text was converted into lower case to maintain uniformity in the dataset.

4.3. Feature Representation

The machine learning models are unable to work on plain text, therefore the textual data had to be transformed into an appropriate format that can be easily understood by the models. For the ML models, we employed the Bag of Words (BoW) model [31] to represent the textual dataset into numerical vectors. The BoW was used for its simple and effective representation of textual data for classical ML algorithms [32]. The Term Frequency-Inverse Document Frequency (TF-IDF) [31] was used to allocate weights to each of the features in the BoW model. The TF-IDF captures the relevance of a feature in the dataset as well as in the individual documents. The TF-IDF allots a high weight to a feature if it appears

frequently in the document, but a lower weight is assigned if the feature appears in several documents of the corpus.

The existing studies have found DL models to be performing better as compared to the traditional ML models [33–35]. The enhancement in performance is achieved by the application of text representation techniques that preserve the context and semantic meaning of the textual data [36]. Such techniques are known as word embeddings, that retain the semantics of the data in numerical vectors and are used with DL models [37]. Hence, the BoW is generally not used with the DL models [9]. There are many word embedding techniques such as the Continuous Bag of Words, Skip-Gram, GloVe etc. [38].

For the DL classifiers, we have applied the GloVe [39] to represent the textual features by transforming them into high-dimensional vectors. GloVe is a word embedding technique where words that share common semantics are placed closer in the vector space based on the training data. In GloVe, a co-occurrence matrix is constructed from the training dataset, followed by the factorization of the matrix to obtain the global vectors. GloVe was used in this work because it has been shown to produce a comparatively better performance than other embedding techniques, and can work on small as well as large corpus sizes [40,41].

4.4. Evaluation Metrics

The evaluation metrics are derived from the four parameters given in Table 1, based on the predicted class label and actual class label. We used seven evaluation metrics to evaluate the output of the ML and DL classification models. A brief detail of the evaluation metrics is discussed in Table 1.

Table 1. Detail of the Evaluation Metrics.

Parameter	Actual Class Label	Predicted Class Label
True Positive (TP)	True	True
True Negative (TN)	Fake	Fake
False Positive (FP)	Fake	True
False Negative (FN)	True	Fake

4.4.1. Accuracy

Accuracy is the fraction of the correct predictions made by the classifier. It is given by Equation (1).

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

4.4.2. Precision

Accuracy may not be a good measure of performance when we are working on an imbalanced dataset. In such scenarios, precision is used to evaluate the effectiveness of the classifiers. It is given by Equation (2).

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

4.4.3. Recall

Recall, also known as sensitivity/TP rate, is the ratio between the TP predictions to the total number of TP and FN predictions made by the classifier. It is given by Equation (3).

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

4.4.4. False Positive Rate (FPR)

The FPR is the ratio between the FP and the total of FP and TN predictions made by the classifiers. It is given by Equation (4).

$$FPR = \frac{FP}{FP + TN} \quad (4)$$

4.4.5. Misclassification Rate

The misclassification rate is the ratio between the incorrect predictions to the total number of predictions made by the classifier. It is given by Equation (5).

$$Misclassification\ Rate = \frac{FP + FN}{TP + TN + FP + FN} \quad (5)$$

4.4.6. F-Score

F-score is the harmonic mean of precision and recall; thus, it is a combination of two metrics in a single measure. F-score is also known as F-measure indicates the performance of the classifiers in the case of imbalanced datasets. It is given by Equation (6).

$$F - score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (6)$$

4.4.7. Receiver Operating Characteristic (ROC) Curve

The ROC curve is the graph obtained by plotting the FPR against the TPR [42]. It shows the performance of the classifier in discriminating between the two classes in binary classification. The area under the ROC (AUC) curve measures the accuracy of the classification algorithm.

4.5. Parameter Settings

The hyper parameters of the ML algorithms were fine tuned using the grid search technique [43]. The value of the parameters is given in Table 2. The NB does not have any hyper parameter to be tuned [43].

Table 2. Parameter values of different classifiers.

Model	Hyperparameter	Value
SVM	C	1
	gamma kernel	1 rbf
KNN DT	n_neighbors criterion	8 gini
	max_depth	8
LR RF	C criterion	1 gini
	max_depth	8
	max_features n_estimators	sqrt 200
	CNN	Conv1D Layers
	Filters	32
	Kernel size	2,3,4
	global avgpool layer	3
	activation function	relu
	optimizer	Adam
	loss function	binary cross entropy

Table 2. *Cont.*

Model	Hyperparameter	Value
LSTM	hidden layers	6
	learning rate	0.001
	batch size	64
	gate activation function	sigmoid
	state activation function	tanh

After preprocessing the dataset as discussed in the preceding paragraphs, the dataset samples were split into train and test partitions with 70 percent and 30 percent of samples, respectively. All of the experiments were carried out on machine with Intel i5 microprocessor onboard and 4 GB of RAM on the Windows 10 operating system.

5. Results and Discussion

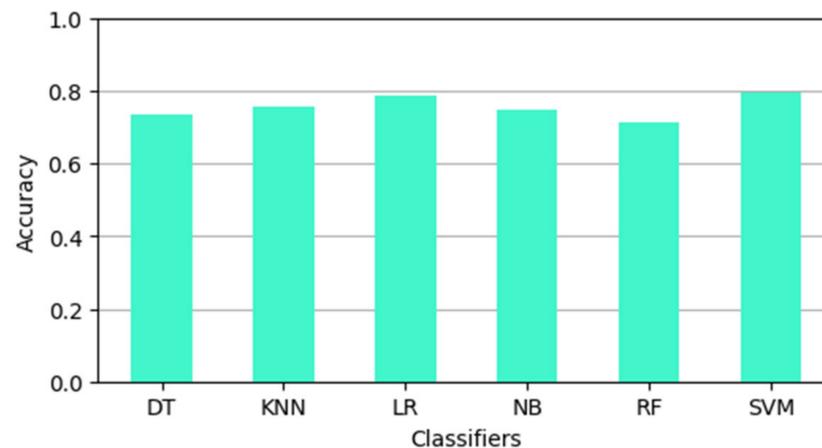
This section discusses the results obtained on the two datasets for the several ML and DL classifiers.

5.1. COVID-19 Fake News Dataset

Table 3 shows the summary of the results obtained from different classifiers for the various evaluation metrics on the COVID-19 Fake News dataset. Figure 1 shows that the SVM and LR classifiers achieved nearly the same classification accuracy, with the SVM achieving 80 percent accuracy followed by the LR and KNN. However, the SVM takes a bit more time for training than the LR model. Among all the models, the tree-based classifiers, i.e., the DT and RF, achieved the lowest accuracy with the RF being the worst performing model at 71 percent. The NB obtains the average accuracy score of 75 percent among all the classifiers.

Table 3. Classification performance of ML classifiers on COVID-19 Fake News dataset.

Classifier	DT	KNN	LR	NB	RF	SVM
Accuracy	0.74	0.76	0.79	0.75	0.71	0.80
Precision	0.78	0.77	0.79	0.82	0.70	0.81
Recall	0.82	0.90	0.91	0.79	0.97	0.90
FPR	0.43	0.51	0.45	0.33	0.78	0.40
Rate of Misclassification	0.26	0.24	0.21	0.25	0.29	0.20
F-score	0.79	0.83	0.85	0.80	0.81	0.85

**Figure 1.** Accuracy of ML classifiers on COVID-19 Fake News dataset.

A good F-score indicates the better performance of the classifiers. The SVM and LR both obtained 85 percent of the F-score value, which indicates a relatively better performance than other classifiers. The DT classifier obtained the lowest F-score, whereas the KNN and NB produced average results among all the classifiers. The RF was marginally better than the DT in terms of the F-score.

The FPR is an important metric, as the fake news being tagged as true by the model could have grave consequences on public health. Therefore, a low FPR is preferable in the fake news detection task. The NB classifier, despite having average accuracy among all the classifiers, has the lowest FPR of 33 percent. This means that if a news article is labeled as true by the model, then there is a 33 percent chance that it was a fake news article. Since the RF and DT are both tree-based models, RF has the highest FPR among all models, whereas DT has comparatively much less FPR. The SVM, despite having the highest accuracy value, does not have the lowest FPR. Other remaining classifiers have an FPR value between 40 to 70 percent, which is quite high. The ROC plot shown in Figure 2 indicates that the SVM, LR and RF are far better in their predictions than the remaining models.

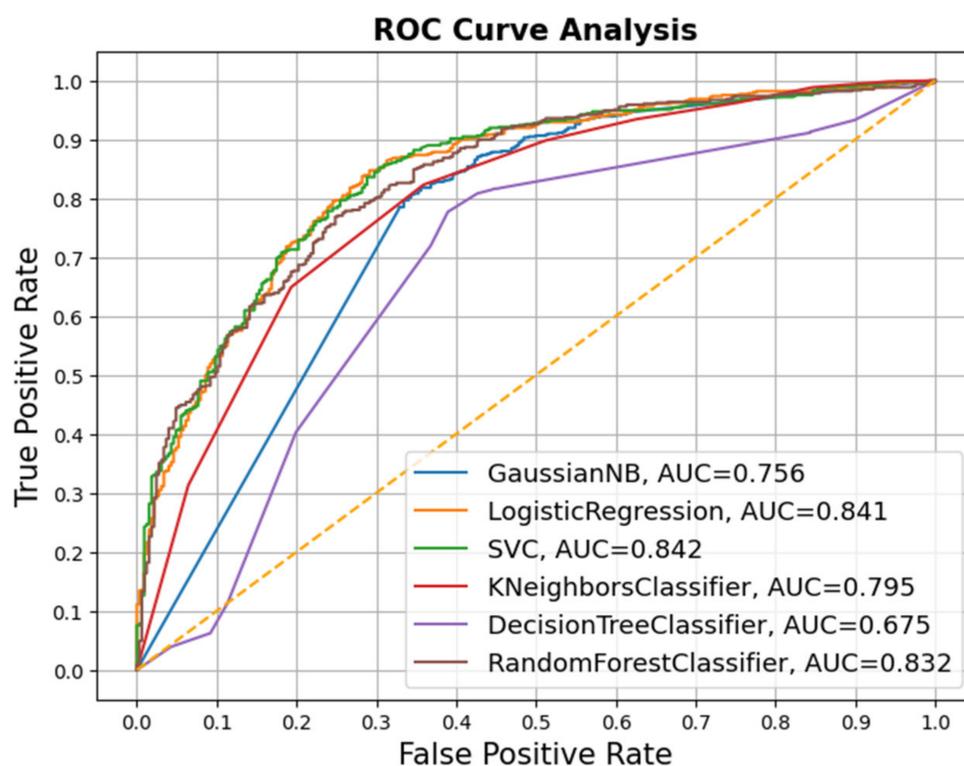


Figure 2. ROC curves of ML classifiers on COVID-19 Fake News dataset.

Now we discuss the performance of the DL models. The DL model does not have any significant performance improvements over the conventional ML model. Table 4 compares the performance of the CNN and LSTM models. Figures 3 and 4 show the ROC curves of CNN and LSTM, respectively. Both of the models achieved almost the same performance for each of the evaluation metrics. The DL models could not outperform the ML models in all evaluation metrics. It indicates that the application of the word embedding technique for feature representation does not have any effect on the classification performance of our dataset. Moreover, training the DL models is also a computationally expensive task compared to the machine learning models in terms of time and memory usage. Therefore, in our evaluation of the intelligent techniques for predicting fake news on COVID-19, the machine learning models would be a better choice than the deep learning models.

Table 4. Classification performance of DL models on COVID-19 Fake News dataset.

Classifier	CNN	LSTM
Accuracy	0.76	0.75
Precision	0.77	0.77
Recall	0.89	0.88
FPR	0.50	0.50
Rate of Misclassification	0.24	0.25
F-score	0.83	0.82

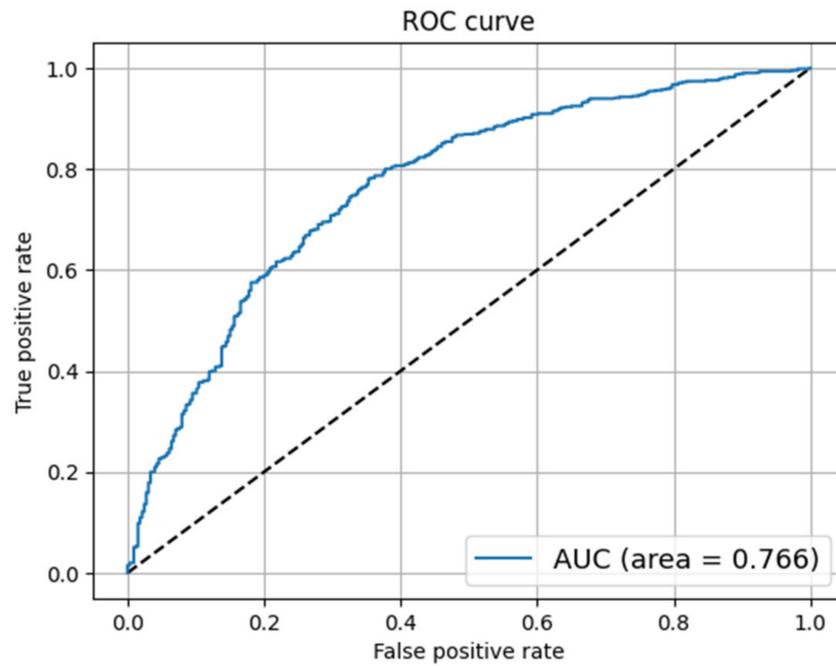


Figure 3. ROC curve of CNN on COVID-19 Fake News dataset.

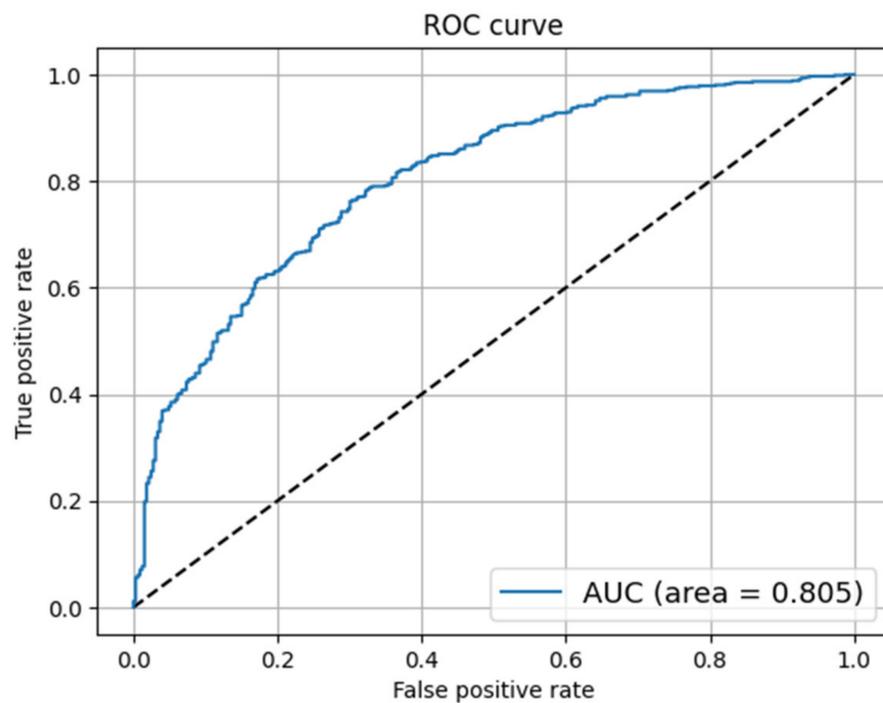


Figure 4. ROC curve of LSTM on COVID-19 Fake News dataset.

5.2. Constraint@AAAI 2021 Dataset

The performance of the ML classifiers on the Constraint@AAAI 2021 dataset is shown in Table 5. The LR classifier consistently produced the highest classification accuracy, followed by the tree-based classifiers i.e., DT and RF. Contrary, to the values in Table 3, the SVM gets the lowest accuracy among all the models. The NB again produces the relatively average results among all the models. The KNN obtained similar accuracy, however, it was relatively much less compared to the highest accuracy obtained by the LR. The same pattern can be seen to be followed by the classifiers in terms of the F-score. The LR achieves the highest F-score, followed by the RF and DT. Again, the SVM obtained the lowest F-score. The bad performance of the SVM may be due to the relatively larger size of the Constraint@AAAI 2021 dataset than the COVID-19 Fake News dataset, as SVMs are not suitable for large datasets [44]. Moreover, we did not perform any feature selection on the dataset, which resulted in a substantial set of training features to be handled by the SVM.

Table 5. Classification performance of ML classifiers on Constraint@AAAI 2021 dataset.

Classifier	DT	KNN	LR	NB	RF	SVM
Accuracy	0.87	0.76	0.91	0.81	0.88	0.65
Precision	0.91	0.96	0.93	0.95	0.88	0.60
Recall	0.84	0.58	0.90	0.68	0.88	0.99
FPR	0.09	0.03	0.08	0.04	0.13	0.72
Rate of Misclassification	0.13	0.24	0.09	0.19	0.12	0.35
F-score	0.87	0.72	0.92	0.80	0.88	0.75

The KNN classifier obtained the lowest FPR at 3 percent, followed by the NB at 4 percent. The NB classifier consistently obtained lower FPR values on both of the datasets. The SVM also produced bad results in terms of FPR values. The ROC plot shown in Figure 5 indicates that the LR and RF are far better in their predictions than the remaining models.

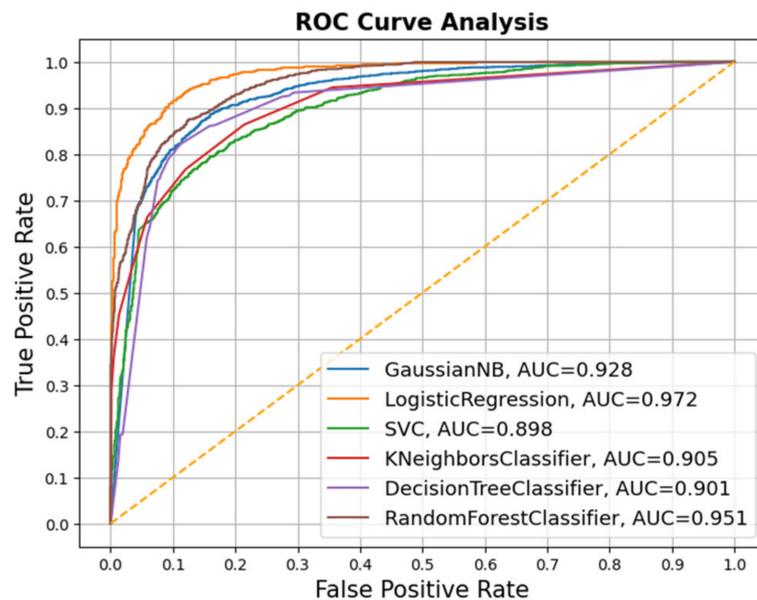


Figure 5. ROC curves of ML classifiers on Constraint@AAAI 2021 dataset.

Table 6 shows the performance of the DL model on the Constraint@AAAI 2021 dataset. Figures 6 and 7 show the ROC curves of the CNN and LSTM, respectively. The DL models once again could not surpass the performance of the ML models on any of the evaluation metrics. The LSTM was shown to give similar results as the CNN for detecting fake news [21], but in our results the LSTM performed marginally better than the CNN. However, the LSTM achieved a respectable FPR value of 7 percent, which is comparable

with the ML models. Contrary to the results in [20], the application of the computationally expensive word embedding technique could not produce better results for the deep learning classifiers.

Table 6. Classification performance of DL models on Constraint@AAAI 2021 dataset.

Classifier	CNN	LSTM
Accuracy	0.81	0.84
Precision	0.83	0.93
Recall	0.77	0.72
FPR	0.17	0.07
Rate of Misclassification	0.19	0.16
F-score	0.80	0.81

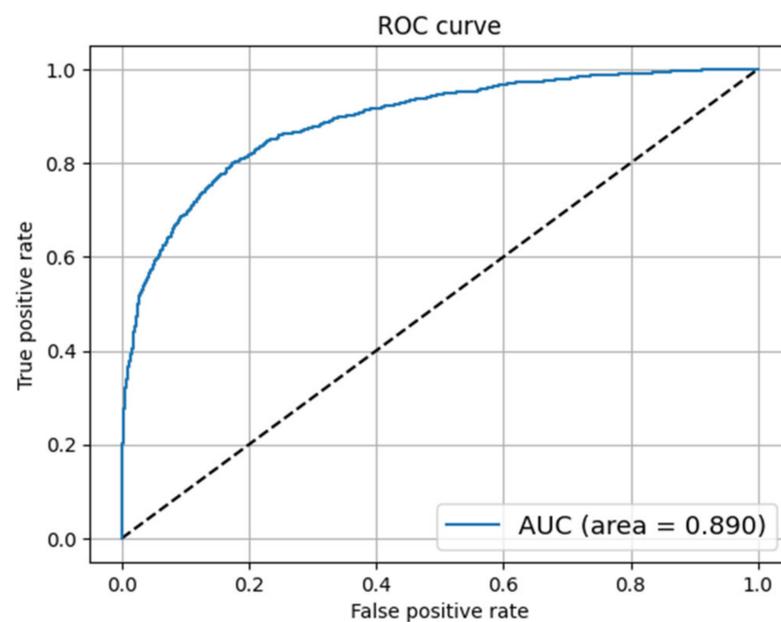


Figure 6. ROC curve of CNN on Constraint@AAAI 2021 dataset.

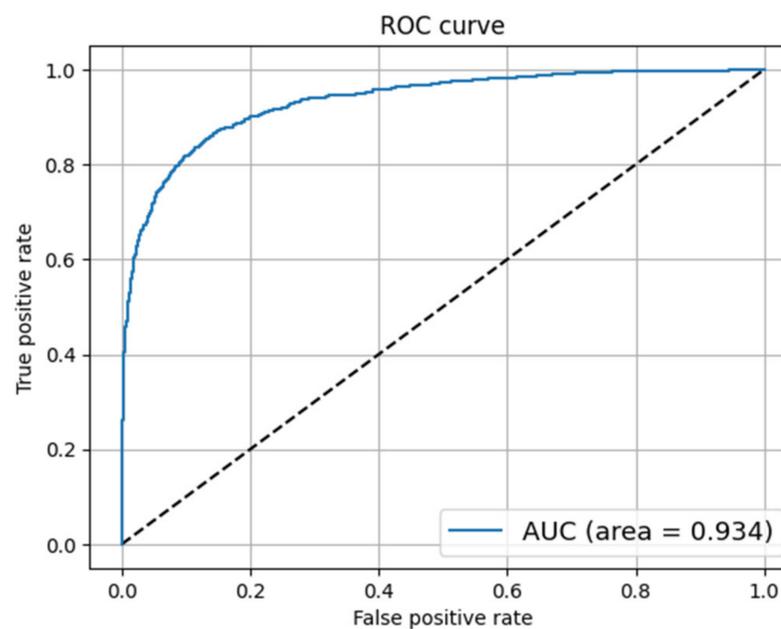


Figure 7. ROC curve of LSTM on Constraint@AAAI 2021 dataset.

Overall, the NB and LR classifiers showed a consistent performance on both datasets in terms of FPR and accuracy score, respectively. In fact, the existing study [11,16] also confirmed that the LR classifier obtained good accuracy for prediction of fake news. Table 7 shows the comparison of the DL classifiers with the NB and LR classifiers in terms of FPR and classification accuracy on the two datasets. As the NB and LR classifiers are both simple, they can be used for the task of predicting fake news. Whereas the DL classifiers could not provide any significant improvement in performance, and took much computational time for the training phase compared to the LR and NB classifiers. Since the authors of fake news tend to frequently use similar words and phrases [45], the content-based representation of text (BoW) in the machine learning models produced better results than the context-based representation in the deep learning models.

Table 7. Comparison of DL and ML classifiers.

Classifiers	COVID-19 Fake News Dataset [28]		Constraint@AAAI 2021 Dataset [30]	
	Accuracy	FPR	Accuracy	FPR
CNN	0.76	0.50	0.81	0.17
LSTM	0.75	0.50	0.84	0.07
LR	0.79	0.45	0.91	0.08
NB	0.75	0.33	0.81	0.04

6. Conclusions

Fake news has become a challenging task for government agencies while handling the COVID-19 epidemic. The spread of unverified news regarding the novel virus has aggravated the pandemic situation around the world. Therefore, curbing this type of misinformation has become inevitable during these challenging times. In this work, we have analyzed several ML and DL models for detecting fake news content. The experiments were performed on two publicly available datasets containing fake news articles on COVID-19. The classification models were evaluated on different evaluation metrics. The LR and NB models were the best performing among all models in terms of accuracy and FPR, respectively. As discussed in the related work section, the LR classifier has shown a good classification performance. Therefore, the LR model could be taken up to develop advanced models, supplemented with feature engineering techniques for enhanced results. However, the DL models did not show any significant performance improvement. It was also observed that adding a layer of word embedding in the DL models did not show any significant performance improvement over the ML models.

There are some limitations to this work which can be taken into account in future works. In this work, we have not applied any feature selection technique. The reduced feature set may further improve the classification performance. Secondly, we have not explored the transformer-based deep learning models that are known for their excellent performance. The above limitations could be addressed to evaluate their effects on the overall performance of the models. In future works, the results obtained from these experiments could be utilized to build classification models with a greater improvement in performance. Moreover, the ensemble model could also be applied to further enhance the results.

Author Contributions: Methodology, M.F.; Resources, H.A., W.A. and A.B.; Supervision, A.A.; Validation, M.F.; Writing—original draft, M.F.; Writing—review & editing, M.W.K. and A.A. All authors have read and agreed to the published version of the manuscript.

Funding: The study was funded by the Deanship of Scientific Research at Umm Al-Qura University, Makkah, Saudi Arabia (Grant Code: 20UQU0066DSR); and Researchers Supporting Project (TURSP-2020/107), Taif University, Taif, Saudi Arabia.

Acknowledgments: The authors would like to thank the Deanship of Scientific Research at Umm Al-Qura University for supporting this work by Grant Code: (20UQU0066DSR); and Researchers Supporting Project number (TURSP-2020/107), Taif University, Taif, Saudi Arabia.

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

References

1. Vosoughi, S.; Roy, D.; Aral, S. The spread of true and false news online. *Science* **2018**, *359*, 1146–1151. [CrossRef] [PubMed]
2. Zaryan, S. Truth and Trust: How Audiences are Making Sense of Fake News. Master's Thesis, Lund University, Stockholm, Sweden, June 2017.
3. Nicola, M.; Alsafi, Z.; Sohrabi, C.; Kerwan, A.; Al-Jabir, A.; Iosifidis, C.; Agha, M.; Agha, R. The socio-economic implications of the coronavirus pandemic (COVID-19): A review. *Int. J. Surg.* **2020**, *78*, 185–193. [CrossRef] [PubMed]
4. Fake News in India—Wikipedia. Available online: <https://en.wikipedia.org/wiki/FakenewsIndia> (accessed on 12 May 2022).
5. Janiesch, C.; Zschech, P.; Heinrich, K. Machine learning and deep learning. *Electron. Mark.* **2021**, *31*, 685–695. [CrossRef]
6. Aggarwal, C.C. *Data Classification: Algorithms and Applications*; CRC Press: Boca Raton, FL, USA, 2014.
7. Lai, C.-M.; Chen, M.-H.; Kristiani, E.; Verma, V.K.; Yang, C.-T. Fake News Classification Based on Content Level Features. *Appl. Sci.* **2022**, *12*, 1116. [CrossRef]
8. Oshikawa, R.; Qian, J.; Wang, W.Y. A survey on natural language processing for fake news detection. *arXiv* **2018**, arXiv:1811.00770.
9. Minaee, S.; Minaee, S.; Kalchbrenner, N.; Cambria, E.; Nikzad, N.; Chenaghlu, M.; Gao, J. Deep Learning—Based Text Classification: A Comprehensive Review. *ACM Comput. Surv.* **2021**, *54*, 1–40. [CrossRef]
10. Chen, Y.; Conroy, N.J.; Rubin, V.L. Misleading online content: Recognizing clickbait as ‘false news’. In Proceedings of the ACM Workshop on Multimodal Deception Detection, Seattle, WA, USA, 13 November 2015.
11. Bourgonje, P.; Schneider, J.M.; Rehm, G. From clickbait to fake news detection: An approach based on detecting the stance of headlines to articles. In *Proceedings of the EMNLP Workshop, Natural Language Processing Meets Journalism*; Association for Computational Linguistics: Copenhagen, Denmark, 2017.
12. Alrubaian, M.; Al-Qurishi, M.; Mehedi Hassan, M.; Alamri, A. A credibility analysis system for assessing information on Twitter. *IEEE Trans. Depend. Sec. Comput.* **2018**, *15*, 661–674. [CrossRef]
13. Castillo, C.; Mendoza, M.; Poblete, B. Information credibility on Twitter. In *Proceedings of the 20th International Conference on World Wide Web (WWW)*; Association for Computing Machinery: New York, NY, USA, 2011.
14. Kaliyar, R.K.; Goswami, A.; Narang, P.; Sinha, S. FNDNet—A deep convolutional neural network for fake news detection. *Cognit. Syst. Res.* **2020**, *61*, 32–44. [CrossRef]
15. Burgoon, J.K.; Blair, J.; Qin, T.; Nunamaker, J. Detecting deception through linguistic analysis. In Proceedings of the 1st NSF/NIJ Conference on Intelligence and Security Informatics, Berlin, Germany, 2–3 June 2003.
16. Vicario, M.D.; Quattrociocchi, W.; Scala, A.; Zollo, F. Polarization and fake news: Early warning of potential misinformation targets. *ACM Trans. Web.* **2019**, *13*, 1–22. [CrossRef]
17. Buntain, C.; Golbeck, J. Automatically identifying fake news in popular Twitter threads. In Proceedings of the 2017 IEEE International Conference on Smart Cloud (SmartCloud), New York, NY, USA, 3–5 November 2017.
18. Newman, M.L.; Pennebaker, J.W.; Berry, D.S.; Richards, J.M. Lying words: Predicting deception from linguistic styles. *Personal. Soc. Psychol. Bull.* **2003**, *29*, 665–675. [CrossRef]
19. Zhou, X.; Jain, A.; Phoha, V.V.; Zafarani, R. Fake news early detection: A theory-driven model. *Digit. Threat. Res. Pract.* **2020**, *1*, 1–25. [CrossRef]
20. Gravanis, G.; Vakali, A.; Diamantaras, K.; Karadais, P. Behind the cues: A benchmarking study for fake news detection. *Expert Syst. Appl.* **2019**, *128*, 201–213. [CrossRef]
21. Krešňáková, V.; Sarnovský, M.; Butka, P. Deep learning methods for Fake News detection. In Proceedings of the IEEE 19th International Symposium on Computational Intelligence and Informatics and 7th IEEE International Conference on Recent Achievements in Mechatronics, Automation, Computer Sciences and Robotics, Szeged, Hungary, 14–16 November 2019.
22. Nassif, A.B.; Elnagar, A.; Elgendy, O.; Afadar, Y. Arabic fake news detection based on deep contextualized embedding models. *Neural Comput. Appl.* **2022**, in press. [CrossRef]
23. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning*; Springer: New York, NY, USA, 2009.
24. Joachims, T. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In *European Conference on Machine Learning*; Springer: Berlin/Heidelberg, Germany, 1998.
25. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
26. Géron, A. *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*; O'Reilly Media: Sebastopol, CA, USA, 2019.
27. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
28. Koirala, A. COVID-19 Fake News Dataset; Mendeley Data, V1. 2021. Available online: <https://data.mendeley.com/datasets/zwfdmp5syg/1> (accessed on 28 June 2022).

29. Koirala, A. COVID-19 Fake News Classification with Deep Learning. 2020. Available online: https://www.researchgate.net/profile/Abhishek-Koirala/publication/344966237_COVID-19_Fake_News_Classification_with_Deep_Learning/links/5f9b6ba5299bf1b53e5130b8/COVID-19-Fake-News-Classification-with-Deep-Learning.pdf(accessed on 28 June 2022).
30. Patwa, P. Fighting an infodemic: COVID-19 fake news dataset. *arXiv* **2020**, arXiv:2011.03327.
31. Sebastiani, F. Machine learning in automated text categorization. *ACM Comput. Surv.* **2002**, *34*, 1–47. [[CrossRef](#)]
32. Zhao, R.; Kezhi, M. Fuzzy bag-of-words model for document representation. *IEEE Trans. Fuzzy Syst.* **2017**, *26*, 794–804. [[CrossRef](#)]
33. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)]
34. Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet classification with deep convolutional neural networks. In *Proceedings of the Advances in Neural Information Processing Systems*; Association for Computing Machinery: New York, NY, USA, 2012.
35. Mikolov, T.; Deoras, A.; Povey, D.; Burget, L.; Cernocky, J. Strategies for training large scale neural network language models. In *Proceedings of the Automatic Speech Recognition and Understanding, Waikoloa, HI, USA, 11–15 December 2011*.
36. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*; Springer: Cham, Germany, 2014.
37. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*; Curran Associates Inc.: Red Hook, NY, USA, 2013.
38. Naseem, U.; Razzak, I.; Khan, S.K.; Prasad, M. A Comprehensive Survey on Word Representation Models: From Classical to State-Of-The-Art Word Representation Language Models. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2021**, *20*, 1–35. [[CrossRef](#)]
39. Pennington, J.; Socher, R.; Manning, C. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2014.
40. Bhoir, S.; Ghorpade, T.; Mane, V. Comparative analysis of different word embedding models. In *Proceedings of the International Conference on Advances in Computing, Communication and Control (ICAC3)*, Mumbai, India, 1–2 December 2017.
41. Shapol, M.; Jacksi, K.; Zeebaree, S. A state-of-the-art survey on semantic similarity for document clustering using GloVe and density-based algorithms. *Indones. J. Electr. Eng. Comput. Sci.* **2021**, *22*, 552–562.
42. Hanley, J.A.; McNeil, B.J. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology* **1982**, *143*, 29–36. [[CrossRef](#)]
43. Yang, L.; Shami, A. On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing* **2020**, *415*, 295–316. [[CrossRef](#)]
44. Cervantes, J.; Li, X.; Yu, W.; Li, K. Support vector machine classification for large data sets via minimum enclosing ball clustering. *Neurocomputing* **2008**, *71*, 611–619. [[CrossRef](#)]
45. Fabio, J.; Bezerra, R. Content-based fake news classification through modified voting ensemble. *J. Inf. Telecommun.* **2021**, *5*, 499–513.