IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 4, 2021, accepted January 28, 2021, date of publication February 1, 2021, date of current version February 9, 2021.

*Digital Object Identifier 10.1109/ACCESS.2021.3056079*

# A Novel Stacking Approach for Accurate Detection of Fake News

**TAO JIANG[1], JIAN PING LI[1], AMIN UL HAQ[1], ABDUS SABOOR[1], AND AMJAD ALI[2]**

[1]School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China
[2]Department of Computer Science and Software Technology, University of Swat, Mingora 19200, Pakistan

Corresponding authors: Tao Jiang (tao1024@yahoo.com), Jian ping li (jpli2222@uestc.edu.cn), and Amin Ul Haq (khan.amin50@yahoo.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61370073, in part by the National High Technology Research and Development Program of China under Grant 2007AA01Z423, and in part by the project of the Science and Technology Department of Sichuan Province.

**ABSTRACT** With the increasing popularity of social media, people has changed the way they access news. News online has become the major source of information for people. However, much information appearing on the Internet is dubious and even intended to mislead. Some fake news are so similar to the real ones that it is difficult for human to identify them. Therefore, automated fake news detection tools like machine learning and deep learning models have become an essential requirement. In this paper, we evaluated the performance of five machine learning models and three deep learning models on two fake and real news datasets of different size with hold out cross validation. We also used term frequency, term frequency-inverse document frequency and embedding techniques to obtain text representation for machine learning and deep learning models respectively. To evaluate models' performance, we used accuracy, precision, recall and F1-score as the evaluation metrics and a corrected version of McNemar's test to determine if models' performance is significantly different. Then, we proposed our novel stacking model which achieved testing accuracy of 99.94% and 96.05 % respectively on the ISOT dataset and KDnugget dataset. Furthermore, the performance of our proposed method is high as compared to baseline methods. Thus, we highly recommend it for fake news detection.

**INDEX TERMS** Deception detection, deep learning, fake news, machine learning, McNemar's test, performance evaluation, stacking.

## I. INTRODUCTION

With the rapid development of the Internet, social media has become an perfect hotbed for spreading fake news, distorted information, fake reviews, rumors, satires. Many people think the 2016 U.S. presidential election campaign has been influenced by fake news. Subsequent to this election, the term has entered the mainstream vernacular [45].

Nowadays fake news has become a major concern for both industry and academia, one of the solutions for this problem is human fact-checking. However, the real-time nature of fake news on social media makes identify online fake news even more difficult [45]. The expert fact-checking may have very limited help because of its low efficiency. In addition, fact-checking by human is very laborious and expensive. Thus, we need to use Machine Learning (ML) and Deep Learning Models (DL) to automate this process. Various hierarchical classification methods such as [29] and [1] can be used for fake news detection.

In recent years, to identify fake news from real news, a lot researchers have been working on establishing effective and automatic frameworks for online fake news detection. A lot of researchers proposed their models based on machine learning and deep learning techniques. However, these proposed methods have some limitations in terms of accuracy. To tackle these issues and effectively detect fake news, a new method is necessary.

In this paper, we evaluated different classification algorithms such as logistic regression (LR) [44], supports vector machine (SVM) [10], k-nearest neighbor (k-NN) [44], decision tree (DT) [41], random forest (RF), convolutional neural network (CNN), gated recurrent network (GRU) [9], long

The associate editor coordinating the review of this manuscript and approving it for publication was Liang-Bi Chen.

short-term memory (LSTM) [19] for the detection of Fake news. Then we used stacking method to improve the individual model performance. Our paper involved two datasets: ISOT dataset.[1] We used techniques like term frequency (TF), term frequency-inverse document frequency (TF-IDF) and embedding to tokenize the title and text feature of these two datasets. Grid Search technique has been used for tuning the hyperparamters and model selection. Various performance evaluation metrics have been used such as accuracy, recall, f1-score, precision and training time. The experimental results of the proposed stacking method have been compared with the state of the art results in the published literature. Furthermore, all experimental results have been tabulated in various tables and graphically shown in various figures for better understanding.

The paper have the following contributions:

- Firstly, five machine learning models and three deep learning models have been trained to compare the performance difference between individual models.
- Secondly, we used two datasets of different size to test models' robustness on datasets of different size.
- Thirdly we employed a corrected version of McNemar's statistical test to decide if there really are significantly differences between two model's performance and choose the best individual model for fake news detection.
- Lastly, our proposed stacking model outperformed the state of the art methods.

The rest of this paper is organized as follows. In section 2 literature review have been presented. In section 3, we have discussed the details of data sets and classification models used in this paper. The experimental results have been presented in section 4. The conclusion and future work direction has been discussed in last section 5.

## II. LITERATURE REVIEW

To detect the fake news numerous machine learning and deep Learning techniques have been recommended by various scholars. In this research study, we have presented some of the baselines fake news detection techniques. The major objectives of the literature review is to identify the problems in the baseline methods and provide a reliable solution.

Some researchers evaluated a lot machine learning models on different datasets to choose the best individual model. Ozbay *et al.* [31] implemented twenty-three supervised artificial intelligence algorithms in three datasets including BayesNet, JRip, OneR, Decision Stump, ZeroR, Stochastic Gradient Descent (SGD), Logistic Model Tree (LMT), etc. According to their experimental results, the decision tree algorithm outperformed all other intelligent classification algorithms in all evaluation metrics except recall. Kaliyar *et al.* [21] used Random Forest, Multinomial Naïve Bayes, Gradient Boosting, Decision Tree, Logistic Regression, Linear-SVM for fake news detection. They found that gradient boosting provides state-of-the-art results and achieved an accuracy of 86% on Fake News Challenge dataset. Gilda *et al.* [14] used TF-IDF of bi-grams and probabilistic context free grammar (PCFG) features to classify news from reliable sources (labeled as 0) and unreliable sources (labeled as 1). Then, they evaluated SVM, SGD, Gradient Boosting, Decision Trees and Random Forests trained on TF-IDF only features, PCFG only features and TF-IDF and PCFG combining features. Finally, they concluded that SGD model trained on TF-IDF feature set only presented the best performance in ROC AUC measure.

There are some researchers designing extraordinary neural networks for fake news detection. Umer *et al.* [40] proposed a model that combines neural network architecture including CNN and LSTM with dimensionality reduction methods, PCA and Chi-Square, to determine if news articles' headline agree with text body. Then they observed that the proposed model resulted in the highest accuracy, 97.8%, with much shorter time. Kaliyar *et al.* [22] created a CNN-based deep neural network called FNDNet and achieved state-of-the-art results with an accuracy of 98.36% on Kaggle fake news dataset. Kumar *et al.* [24] performed a CNN + BiLSTM ensemble model with attention mechanism on their own datasets and FakeNewsNet dataset and achieved the highest accuracy of 88.78%. Ajao *et al.* [4] used a hybrid of CNN and RNN to classify fake news messages from Twitter posts. They compared the performance of plain LSTM model, LSTM with dropout regularization and LSTM-CNN hybrid model on a dataset containing approximately 5,800 tweets centered on five rumor stories. Then they concluded that the plain LSTM model has the best performance while LSTM method with dropout regularization suffers from underfitting and LSTM-CNN hybrid model suffers from limited data. Roy *et al.* [36] utilized CNN to identify hidden features and RNN to capture temporal sequence and fed obtained representation into MLP for classification. They used pre-trained 300-dimensional Google News Vectors to get feature embeddings and fed them into separate convolutional layers and separate Bi-LSTM layers. Their models were tested on Liar Dataset with an accuracy of 44.87% which outperforms the state-of-the-art model by 3%.

Some researchers try to solve the spreading of fake news on social network platform. Ma *et al.* [26] used a deep learning model to detect rumors on twitter and weibo. Their RNN-based model allow for early detection and achieves significant improvement over state-of-the-art algorithms. Monti *et al.* [28] presented a propagation-based approach for fake news detection on Twitter social network. They used a four-layer Graph CNN with two convolutional layers and two dense layers to predict and achieved great performance with 92.7% ROC AUC. Ruchansky *et al.* [37] build a CSI model that utilized the text, the user response and source characteristics at once for fake news detection. The Capture module in CSI was constructed with LSTM to exploit the temporal pattern of user response and text. The Score module of

[1] https://www.uvic.ca/engineering/ece/isot/datasets/fake-news/index.php and KDnugget dataset[2]

CSI used a neural network and user graph to assign a score to each user. The score, then, can be used to identify suspicious users. The Integrate module combined the information from first two module and classify each article as fake or not fake.

To improve the results, some researchers also used other different news features. Reis *et al.* [35] evaluated the discriminative features extracted from news content, news source, environment using several classifiers such as k-NN, NB, RF, SVM with kernel RBF, and XGBoost w.r.t. the area under the ROC curve and the Macro F1 score measurements. Della Vedova *et al.* [11] proposed a novel machine learning fake news detection method which, by combining news content and social context features increases accuracy by up to 4.8%. Then they validated it with a real-world dataset, obtaining a fake news detection accuracy of 81.7%. Shabani *et al.* [38] selected 5 different machine learning models: Logistic Regression, SVM, Random Forest, Neural Networks, and Gradient Boosting Classifier for Fake news and Satire classification. Making use of TF-IDF + paralinguistic features + sentiment related features and text similarity features extracted by querying Google, the Neural Network model achieved the highest accuracy of 81.64% which was better than the baseline results by 2.54%.

Some researchers evaluated how different feature extraction methods affect the results. Ahmed *et al.* [3] compared 2 different features extraction techniques namely, term frequency (TF) and term frequency-inverted document frequency (TF-IDF) and 6 n-gram machine learning classification models including SGD, SVM, LSVM, LR, KNN, and DT on two datasets. They saw that an increase in the n-gram size would cause a decrease in the accuracy. Agudelo *et al.* [2] used Naive Bayes Model for the identification of false news in public data sets. Their results showed that it was more effective to use CountVectorizer than TfidfVectorizer to preprocess the data, since CountVectorizer method correctly classified 89.3% of the news.

Some researchers proposed their own deep learning framework and achieved great accuracy on their datasets. Zhang *et al.* [45] presented a detailed comparison of thirteen existing fact-checking resources and seven public datasets. Then, they illustrated the overall categorizations of the current researches on online fake news detection. Finally, they proposed a three layers ecosystem including alert layer, detection layer(fact-checking, fake news detection), and intervention layer. Singhania *et al.* [39] constructed a three level hierarchical attention network(3HAN) based on a proposed HAN. 3HAN has three levels, one each for words, sentences, and the headline and provided an understandable output to enable further manual fact checking. They also used headlines to perform a supervised pre-training of the initial layers of 3HAN. Long *et al.* [25] performed an attention-based LSTM model on LIAR dataset which incorporates speaker profiles such as speaker name, title, party affiliation, current job, location of speech and credit history. Their experimental results show that speaker profile information can improve CNN and LSTM models significantly.

To detect fake news, researchers did a lot novel work. Rasool *et al.* [33] proposed a novel method of multilevel multiclass fake news detection based on relabeling of the dataset and learning iteratively. The method is tested using different supervised machine learning algorithms like SVM and decision tree with hold-out, test dataset and cross validation approaches. Their method outperformed the benchmark with an accuracy of 39.5% on LIAR dataset. Oshikawa *et al.* [30] compared and discussed nine benchmark datasets and experimental results of different methods. Then, they suggested that meta-data and additional information can be utilized to improve the robustness and performance. Jain *et al.* [20] proposed a mix of Naïve Bayes classifier, SVM, and natural language processing techniques on a fake news dataset and their model accuracy is up to 93.6% which was better than the baseline results by 6.85%. Reis *et al.* [34] provided a great understanding of how features are used in the decisions taken by models. They performed an unbiased search for XGB models. Each of them was composed of a set of randomly chosen features. Then they used SHAP to explain why news are classified as fake or real by representative models of each model cluster.

Fake news detection is a global problem, different fake news in different countries is written in different languages. A lot researchers try to find a solution for multiple language fake news detection by constructing a new dataset or training models on different language datasets. Faustini *et al.* [13] trained Naïve Bayes, K-Nearest Neighbors, SVM and Random Forest from five datasets in three languages. They compared the results obtained through a custom set of features, Document-class Distance, bag-of-words and Word2Vec in accuracy and F1-Score measures. Eventually, they concluded that SVM and RandomForest outperformed other algorithms and bag-of-words achieved the best results in general. Wang *et al.* [42] presented a English fake news dataset. They also designed a hybrid CNN model to integrate metadata with text and proved that this hybrid approach can improve a text-only model.

In [32], the researchers compiled a new Spanish language corpus of news from January to July of 2018. The corpus is annotated with two labels (real and fake) and true news and fake news are pairs of events. Statistics of the corpus like vocabulary overlap of the different news topics and labels are also mentioned in their paper. To detect fake news, they performed SVM, LR, RF, and boosting on bag of words (BOW), POS tags, and n-grams features sets of their datasets and find character 4-grams without removing the stop words with the Boosting algorithm has the best performance in accuracy. Amjad *et al.* [5] proposed a new Urdu language corpus: "Bend The Truth" for fake news detection which contains 900 news articles, 500 annotated as real and 400 labeled as fake. Their text representation feature sets include the combination of word n-grams, character n-grams, functional word n-grams (n ranging from 1 to 6) with a variety of feature weighting schemes including binary values, normalized frequency, log-entropy

**TABLE 1.** Summary of the published fake news papers.

| Ref | Year | Contributions | Data set | Models |
|---|---|---|---|---|
| [13] | 2020 | Trained four models from five datasets in three languages | Btvlifestyle, FakeOrRealNews, FakeNewsData1, FakeBrCorpus, TwitterBR | Naïve bayes (NB), KNN, SVM, RF |
| [31] | 2020 | Implemented twenty-three supervised artificial intelligence algorithms in three datasets | ISOT, BuzzFeed Political News Data set, Random Political News Data set | ZeroR, SGD, CV Parameter Selection, Randomizable Filtered Classifier, Logistic Model Tree, Locally Weighted Learning, Classification via Clustering, Weighted Instances Handler Wrapper, Ridor, MLP, Ordinal Learning Model, Simple Cart, Attribute Selected Classifier, J48, Sequential Minimal Optimization (SMO), Bagging, DT, Kernel-Logistic Regression, IBk |
| [40] | 2020 | Proposed a novel model that combines CNN and LSTM with PCA and Chi-Square | - | CNN+LSTM with PCA |
| [22] | 2020 | Achieved state-of-the-art results with an accuracy of 98.36% | Kaggel fake news dataset | DT, RF, CNN, LSTM, KNN, Multinomial Naïve Bayes |
| [24] | 2020 | Proposed CNN + BiLSTM ensembled model with attention mechanism | 1356 news instances on Twitter and other media sources | CNN+BiLSTM |
| [5] | 2020 | 900 news articles, 500 annotated as real and 400, as fake | Multinomial Naive Bayes (MNB), Bernoulli Naive Bayes (BNB), SVM, LR, RF DT, and AdaBoost | Compiled a new Urdu language corpus |
| [35] | 2019 | Evaluated features extracted from news content, news source, environment | 2282 BuzzFeed news articles related to the 2016 U.S. election | KNN,NB,RF,SVM,XGBoost |
| [33] | 2019 | Proposed a novelty binary classification method for multilabel fake news dataset | Liar dataset | SVM |
| [21] | 2019 | Evaluated different machine learning models for fake news detection | Kaggle fake news dataset | RF, Multinomial Naive Bayes, GB, DT, LR, Linear-SVM |
| [20] | 2019 | Include machine learning and deep learning models in one system | - | NB,SVM |
| [34] | 2019 | Explained how features are used in the decisions taken by models | - | XGBoost |
| [28] | 2019 | Presented a propagation-based approach | - | Graph CNN |
| [32] | 2019 | 491 ture, 480 false news | SVM, LR, RF and Boosting | Compiled a new Spanish language corpus |
| [3] | 2018 | Compared different feature extraction techniques on different machine learning models | 12 600 fake news articles from kaggle and 12 600 truthful political articles | LR, SGD, DT, KNN, LSVM, SVM |
| [11] | 2018 | Combined news content and social context features for fake news detection | - | LR |
| [2] | 2018 | Compared two different feature extraction techniques: TF and TF-IDF | - | SVM |
| [38] | 2018 | Used TF-IDF features, paralinguistic features, sentiment related features and text similarity features | - | LR, SVM, RF, GB, Neural Networks |
| [4] | 2018 | Used a hybrid of CNN and RNN | 5,800 tweets centered on five rumor stories | LSTM+CNN |
| [36] | 2018 | Utilized CNN and RNN to obtain text representation and fed obtained representation into MLP for classification | Liar dataset | CNN+Bilstm |
| [14] | 2017 | Evaluated different models on TF-IDF features and probabilistic context free grammar (PCFG) features | - | DT, Gradient Boosting, RF, SGD, SVM |
| [39] | 2017 | Constructed a three level hierarchical attention network(3HAN) | - | 3HAN |
| [42] | 2017 | Presented a new public benchmark dataset (LIAR) | Liar dataset | SVM, LR, BiLSTM, CNN |
| [25] | 2017 | Compared models' performance with and without speaker profile information | Liar dataset | Attention based LSTM model |
| [37] | 2017 | utilized features like the text, the user response and source characteristics | Twitter and Weibo | CSI |
| [26] | 2016 | Proposed a RNN based model for early detection | Twitter and Weibo microblog datasets | SVM-TS, SVM-RBF,DTC, RF, tanh-RNN, LSTM, GRU |

weighting, raw frequency, relative frequency, and TF-IDF. Then they evaluated classifiers like Multinomial Naive Bayes (MNB), Bernoulli Naive Bayes (BNB), SVM, LR, RF, DT, and AdaBoost in balanced accuracy, $F1_{Real}$ score, $F1_{Fake}$ score, and ROC-AUC with 10-fold cross-validation. Finally they found AdaBoost with the combination of character 2-grams and word 1-grams has the best performance in $F1_{Fake}$ score.

The proposed methods in literature have been summarized in Table 1. In Table 1, we presented proposed models,

**TABLE 2.** Description of ISOT dataset.

| Title | Text | Subject | Date | Label |
|---|---|---|---|---|
| UK transport police leading in... | LONDON (Reuters) - British counter-terrorism. | worldnews | September 15, 2017 | REAL |
| Pacific nations crack down on... | WELLINGTON (Reuters) - South Pacific island nation. | worldnews | September 15, 2017 | REAL |
| Three suspected al Qaeda... | ADEN, Yemen (Reuters) - Three suspected al Qaeda... | worldnews | September 15, 2017 | REAL |
| Chinese academics prod Beijing... | BEIJING (Reuters) - Chinese academics are publicly. | worldnews | September 15, 2017 | REAL |
| Classic! Kid Rock Hits Back At... | Not much to say after this classic response from. | politics | Sep 2, 2017 | FAKE |
| 'My Pillow' CEO Mike Lindell... | Who hasn t seen his commercials over and over and... | politics | Sep 1, 2017 | FAKE |
| Bitter John McCain Calls Trump... | What the heck! Senator John McCain just admitted.. | politics | Sep 1, 2017 | FAKE |
| Muslim Activist Caught Sending... | This woman has no shame1 Muslim activist Linda... | politics | Sep 1, 2017 | FAKE |

**TABLE 3.** Description of KDnugget dataset.

| Id | Title | Text | Label |
|---|---|---|---|
| 7614 | Globalization Expressway to... | If humans were largely moral and ethical beings... | FAKE |
| 10294 | Watch The Exact Moment Paul... | Google Pinterest Digg Linkedin Reddit... | FAKE |
| 7060 | Now Malaysia Dumps US for... | Now Malaysia Dumps US for Chinese Naval Vessels... | REAL |
| 10142 | Bernie supporters on Twitter... | Kaydee King (@KaydeeKing) November 9, 2016 The... | FAKE |
| 875 | The Battle of New York: Why... | It's primary day in New York and front-runners... | REAL |
| 6903 | Tehran, USA | I'm not an immigrant, but my grandparents are... | FAKE |
| 7341 | Girl Horrified At What She... | Share This Baylee Luciani (left), Screenshot of... | FAKE |
| 95 | 'Britain's Schindler' Dies at... | A Czech stockbroker who saved more than 650 Jewish... | REAL |

contributions, data sets of the related work for better understanding about fake news detection.

## III. MATERIALS AND METHODS

The fundamental concepts of proposed models have been discussed in below sections.

### A. DATA-SET

Our paper involved two datasets, ISOT fake news dataset and KDnugget dataset. ISOT dataset was entirely collected from real-world sources [31]. The real news was collected by crawling articles from Reuters.com and fake news were collected from unreliable websites that were flagged by Politifact and Wikipedia. The articles were mostly released from 2016 to 2017. This dataset includes 44898 data in total, 21417 are real news (labeled as 1) and 23481 are fake news (labeled as 0). The features include title, text (news body), subject, date and label. The news subjects has different categories like 'politicsNews', 'worldnews', 'News', 'politics', 'Government News', 'left-news', 'US_News', 'Middle-east'. We selected features like title and text in ISOT dataset to train our models.

KDnugget dataset was made public by KDnuggets (a data website) [13]. There are 6335 fake and real news in KDnugget dataset including 3171 real ones (labeled as 1) and 3164 fake news (labeled as 0). It is publicly available. The real news came from media organizations such as the New York Times, WSJ, Bloomberg, NPR, and the Guardian and were published in 2015 or 2016. The fake news is randomly selected from the kaggle fake news dataset. It has three useful features including title, text (news body) and label. To obtain more information, we selected features like title and text in KDnugget dataset to train our models.

Obviously, these two datasets have different size. The KDnugget data set is only one-seventh of ISOT data set. A little section of ISOT dataset and KDnugget dataset had been described in Table 2 and Table 3 respectively.

### B. PRE-PROCESSING

Before the data were fed into machine learning and deep learning models, the text data need to be preprocessed using methods like stop word removal, tokenization, sentence segmentation, and punctuation removal. These operations can significantly help us select the most relevant terms and increase model performance.

Both our datasets come from real word news articles, so there are a lot meaningless urls which carry none information. So we cleaned our data first by removing these urls. Stop word removal is our next preprocessing step. Stop words frequently used in English sentences to complete the sentence structure but they are insignificant in expressing individual's thoughts. So in all our experiments, we removed them in case they crate too much noise. After the text data was cleaned, we tokenized them by using TF-IDF and embedding techniques.

- Term frequency (TF)
  TF is a common tokenization technique that calculate the similarity between documents by using the counts of words in the documents. By utilizing TF technique, each document will be represented by a vector that contains the word counts. Then each vector will be normalized and the sum of its elements will be one which makes the word counts convert into probabilities.
  Let $D$ denote a corpus and let $d$ denote a document. Suppose $w$ is the word in $d$ and $n_w(d)$ is number of times the word $w$ appears. Thus the size of d can be represented as $|d| = \sum_{w \in d} n_w(d)$. The normalized TF for word w in document d can be defined as follows:
  $$TF(w)_d = \frac{n_w(d)}{|d|} \tag{1}$$
- Term frequency-inverted document frequency (TF-IDF)
  In our machine learning experiments, we also used TF-IDF to transform the data into vectors. TF-IDF is a weighting metric commonly used in text classification problem. It is used to assign a score which shows the importance of the term to every term in the document. In this method, a term's significance increases with the frequency of the term in the dataset. Let D denote a corpus, namely, set of news articles. Let d denote an article which consists of a set of words w. The inverse document frequency (IDF) can be computed mathematically using following equation.
  $$IDF(w)_D = \left(1 + \log(\frac{|D|}{\{|d : D|w \in d|\}})\right) \tag{2}$$
  TF-IDF (term frequency-inverted document frequency) for the word w with respect to document d and corpus D is calculated as follows:
  $$TF - IDF(w)_{d,D} = TF(w)_d \times IDF(w)_D \tag{3}$$
- Embedding
  In our deep learning experiments, we used pretrained word embedding technique to obtain text representation. In word embedding space, the geometrical distance between word vectors represents the semantic relationship. Word embedding can project the real human language grammar into a vector space. In an ideal embedding space, words sharing the same semantic meaning will be embedded into a similar vectors. The geometrical distance between two word vectors is highly associated with their linguistic meaning. To reduce trainable parameters and increase time efficiency, we used a file of GloVe word embeddings called 'glove.twitter.27B.100d.txt' and loaded it in our model.

### C. MACHINE LEARNING CLASSIFICATION MODELS

In this study, for classifying fake news and real news, five machine learning models have been used. These models have been discussed in detail.

#### 1) LOGISTIC REGRESSION (LR)

Logistic regression is a common machine learning classification algorithm. In a binary classification problem, to predict the values of predictive variable $y$, where $y \in [0, 1]$. The negative class is denoted by 0 while positive class is by 1.

In order to classify two classes 0 and 1, a hypothesis $h(\theta) = \theta^T X$ will be designed and the threshold of classifier's output is when $h\theta(x) = 0.5$. If the value of hypothesis $h\theta(x) \geq 0.5$, it will predict $y = 1$ which means that this news is real and if the value of $h\theta(x) < 0.5$, then it predicts $y = 0$ which shows that this news is fake.

Hence, the prediction of logistic regression under the condition $0 \leq h\theta(x) \leq 1$ is done. Logistic regression sigmoid function can be written in equation 4 as follows:

$$h\theta(x) = g\left((\theta^T X)\right) \tag{4}$$

where $g(z) = 1/(1 + x^{-z})$ and $h\theta(x) = 1/(1 + x^{-\theta^T X})$

Similarly, the logistic regression cost function can be written in equation 5 as follows:

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} cost(h\theta(x^{(i)}), y^{(i)}) \tag{5}$$

#### 2) DECISION TREE (DT)

DT is an important supervised learning algorithm. Researchers tend to use tree-based ensemble models like Random Forest or Gradient Boosting on all kinds of tasks. The basic idea of DT is that it develops a model to predict the value of a dependent factor by learning various decision rules inferred from the whole data. Decision Tree has a top-down structure and shapes like a tree in which a node can only be a leaf node which is binding with a label class or a decision node which are responsible for making decisions. Decision Tree is easily understandable about the process of making the decisions and predictions. However, it is a weak learner which means it may have bad performance on small datasets.

The key learning process in DT is to select the best attribute. To solve this problem, various trees have different metrics such as information gain used in ID3 algorithm, gain_ratio used in C4.5 algorithm. Suppose discrete attribute $A$ has n different values and $D_i$ is the set which contains all samples that has a value of i in training dataset D. The gain ratio and information gain for attribute $A$ can be calculated as follows:

$$Gain(A, D) = Entropy(D) - \sum_{i=1}^{n} \frac{|D_i|}{|D|} Entropy(D_i) \tag{6}$$

$$GainRatio(A, D) = \frac{Gain(D, A)}{IV(A)} \tag{7}$$

where intrinsic value of attribute A can be calculated as:

$$IV(A) = -\sum_{i=1}^{n} \frac{|D_i|}{|D|} log2 \frac{|D_i|}{|D|} \tag{8}$$

### 3) K-NEAREST NEIGHBOR (KNN)

K-NN is a well known algorithm in machine learning. The K-NN procedures are very simple. Given a test sample, it first finds out k nearest neighbors to this sample based on a distance measure. Then it predicts class label of the test instance with major vote strategy. Sometimes classification performance of K-NN is not high mostly because of curse of dimensionality. K-NN also is a lazy learning algorithm and it can spend a lot time on classification. The main procedures of K-NN algorithm are given in algorithm 1.

**Algorithm 1** KNN Algorithm

```
for all unlabeled data u do
    for all labeled data v do
        compute the distance between u and v
        find k smallest distances and locate the corresponding labeled instances $v_1, \ldots v_k$
        assign unlabeled data u to the label appearing most frequently in the located labeled instances
    end for
end for
End
```

### 4) RANDOM FOREST (RF)

Random Forest is an ensemble consisting of a bagging of unpruned decision trees with a randomized selection of features at each split. Each individual tree in the random forest produces a prediction and the prediction with the most votes are the final prediction. According to No Free Lunch theorem: There is no algorithm that is always the most accurate, thus RF is more accurate and robust than the individual classifiers.

The random forest algorithm can be expressed as

$$F(x) = \arg\max_{l} \left\{ \sum_{i=1}^{z} T(A(B, \theta_k)) \right\} \tag{9}$$

where F(x) is the random forest model, j is the target category variable and F is the characteristic function. To ensure the diversity of the decision tree, the sample selection of random forest and the candidate attributes of node splitting is randomness. Pseudocode of the random forest algorithm is described in algorithm 2.

**Algorithm 2** Random Forest Algorithm

```
Require: Training set ($m$ is the number of training set, $f$ is the feature set)
Ensure: Random forest with $m_{sub}$ CART trees
1: Draw Bootstrap sample sets $m_{sub}$ with replacement
2: Choose a sample set as the root node and train in a completely split way
3: Select $f_{sub}$ randomly from $f$ and choose the best feature to split the node by using minimum principle of Gini impurities
4: Let the nodes grow to the maximum extent. Label the nodes with a minimum impurity as leaf node
5: Repeat steps 2-4 until all nodes have been trained or labeled as leaf nodes.
6: Repeat steps 2-5 until all CART has been trained
7: Output the random forest with $m_{sub}$ CART trees
```

### 5) SUPPORT VECTOR MACHINE (SVM)

For binary and multi-classification related problems, SVM is one of the most popular models [7], [8], [10], [16], and [17]. It is a supervised machine learning classifier and many researchers adopted it for binary and mutli-classification related problems [7]. The instances are separated with a hyper plane in binary classification problem in such a way $w^T x + b = 0$, where $w$ is a dimensional coefficient weight vector which is normal to the hyper-plane. The bias term $b$, which is the offset values from the origin, and data points are represented by $x$. Determining the values of $w$ and $b$ is the main task in SVM. In linear case, $w$ can be solved using Lagrangian function. On the maximum border, the data points are called support vectors. As an outcome, the solution of $w$ can be expressed mathematically as in equation 6.

$$w = \sum_{i=1}^{n} \alpha_i Y_i X_i \tag{10}$$

In equation 6, $n$ denotes support vectors, and target class label is $Y_i$ which is corresponding to samples $x$. The term bias $b$ can be computed by $y_i \left( w^T x_i + b \right) - 1 = 0$. In nonlinear case, kernel trick and decision function of n $w$ and $b$ are expressed as in equation 7 as follows:

$$f(x) = sgn \left( \sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b \right) \tag{11}$$

The positive semi definite functions, which follows the Mercer 's condition like kernel functions [44]: the polynomial kernel can be written in equation 8 as:

$$K(x, x_i) = ((x^T x_i) + 1)^d \tag{12}$$

The Gaussian kernel is expressed in equation 10 as:

$$K(x, x_i) = \exp(-\gamma ||x - x_i||^2) \tag{13}$$

Here,$C$ and $\gamma$ are two parameters required to be defined by SVM.

## D. DEEP LEARNING MODELS DESCRIPTION

In this study, for classifying fake news and real news, three deep learning models have been used. These models have been discussed in detail.

### 1) CONVOLUTIONAL NEURAL NETWORKS (CNNs)

The CNN model structure has been shown in Table 4.

### 2) LONG SHORT-TERM MEMORY (LSTM)

To deal with vanishing gradient problem which means when layers increase the neural network will become untrainable,

**TABLE 4.** **CNN, LSTM and GRU Models' Structures.**

| Model name | Layer(type) | Output shape | Param# |
|---|---|---|---|
| CNN | embedding_3(Embedding) | (None,300,100) | 1000000 |
| | dropout_1(Dropout) | (None,300,100) | 0 |
| | conv1d_1(Conv1D) | (None,297,128) | 51328 |
| | global_max_pooling1d_1 | (None,128) | 0 |
| | dropout_2(Dropout) | (None,128) | 0 |
| | dense_5(Dense) | (None,128) | 16512 |
| | dense_6(Dense) | (None,1) | 129 |
| LSTM | embedding_1(Embedding) | (None,300,100) | 1000000 |
| | lstm_1(LSTM) | (None,300,64) | 42240 |
| | lstm_2(LSTM) | (None,128) | 98816 |
| | dense_1(Dense) | (None,32) | 4128 |
| | dense_2(Dense) | (None,1) | 33 |
| GRU | embedding_5(Embedding) | (None,300,100) | 1000000 |
| | gru_1(GRU) | (None,128) | 87936 |
| | dropout_5(Dropout) | (None,128) | 0 |
| | dense_9(Dense) | (None,1) | 129 |

Hochreiter *et al.* [19] developed LSTM algorithm. In practice, long short-term memory (LSTM) has become one of the common recurrent layers used to train time series and sequence data. The LSTM cell structure and LSTM model structure have been shown in Figure 1 and Table 4.

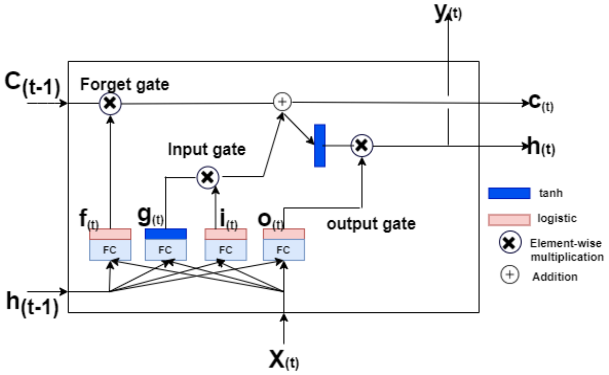


**FIGURE 1.** **LSTM cells.**

As we can see from Figure 1, long-term state $c_{(t-1)}$ first processed by forget gate, dropping some memories, and plus some new memories selected by the input gate. Then $c_{(t-1)}$ become the result $c_{(t)}$. Besides that, $c_{(t-1)}$ is copied and passed through the tanh function and filtered by the output gate to compute the short-term state $h_{(t)}$ which is also the cell's output at t time step, $y_t$. Normally in a basic RNN cell, there is only one fully connected layer that outputs $g_t$. However in LSTM cell, there are three more gate controllers layers. Due to logistic activation function, their computation results range from 0 to 1. By using element-wise multiplication, if they output zeros, the according gate will be closed and if they output ones the gate will be open. The forget gate determine what to delete in long-term state. The input gate determine what should be added to the long-term state. The output gate determine what parts of the long-term state should be read and output at current time step. Let x be the input sequence vector representation and W be the weights associated with each matrix element. The involved computation in LSTM cell can be summarized as follow:

$$i_{(t)} = \sigma(W_{xi}^T x_{(t)} + W_{hi}^T h_{(t-1)} + b_i) \tag{14}$$

$$f_{(t)} = \sigma(W_{xf}^T x_{(t)} + W_{hf}^T h_{(t-1)} + b_f) \tag{15}$$

$$o_{(t)} = \sigma(W_{xo}^T x_{(t)} + W_{ho}^T h_{(t-1)} + b_o) \tag{16}$$

$$g_{(t)} = tanh(W_{xg}^T x_{(t)} + W_{hg}^T h_{(t-1)} + b_g) \tag{17}$$

$$c_{(t)} = f_{(t)} \otimes c_{(t-1)} + i_{(t)} \otimes g_{(t)} \tag{18}$$

$$y_{(t)} = h_t = o_{(t)} \otimes tanh(c_{(t)}) \tag{19}$$

### 3) GATED RECURRENT UNIT (GRU)

GRU cell [9] is a simple version of LSTM cell, but sometimes its performance is even better than LSTM. This conclusion can also be proven in our experiments which shows that their performance is almost the same but the time GRU neural network spent on training is much shorter. In GRU cell, the long short term state are merged into a single vector $h_{(t)}$. And there are only two gate controller $z_{(t)}$ and $r_{(t)}$. When $z_{(t)}$ outputs 1, the forget gate is open and the input gate is closed. when $z_{(t)}$ outputs 0, it will close the forget gate and open input gate. At every time step, there always will be a output unlike in LSTM cell where exists an output gate. And $r_{(t)}$ controls what content in the previous state should be sent to the main layer $g_{(t)}$. The GRU cell and model structure have been shown in Figure 2 and Table 4. The all computations involved in GRU cell can be summarized as follows:

$$z_{(t)} = \sigma(W_{xz}^T x_{(t)} + W_{hz}^T h_{(t-1)} + b_z) \tag{20}$$

$$r_{(t)} = \sigma(W_{xr}^T x_{(t)} + W_{hr}^T h_{(t-1)} + b_r) \tag{21}$$

$$g_{(t)} = tanh(W_{xg}^T x_{(t)} + W_{hg}^T (r_{(t)} \otimes h_{(t-1)}) + b_g) \tag{22}$$

$$h_{(t)} = z_{(t)} \otimes h_{(t-1)} + (1 - z_{(t)}) \otimes g_{(t)} \tag{23}$$

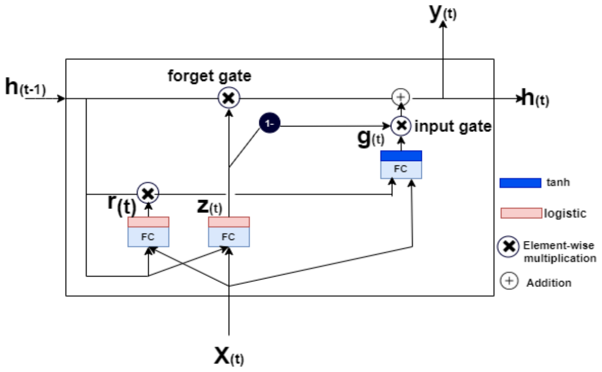


**FIGURE 2.** **GRU cells.**

## E. HOLD OUT CROSS VALIDATION METHOD

For best model selection, we have adopted hold out cross-validation method. In hold out method, the data set is divided into two parts for training and testing. In our experiments, 80% of the instances are used to train the classifiers and the remaining 20% of the datasets are used for testing the classifiers.

### F. MODEL EVALUATION CRITERIA

We employed accuracy, precision, recall, and f1-score [18], for model evaluation and in equation 24, 25, 26, and 27, we expressed these evaluation metrics as follows:

$$Accuracy\ (Acc) = \frac{TP+TN}{TP+TN+FP+FN}100\% \tag{24}$$

$$Recall\ (Re) = \frac{TP}{TP+FN}100\% \tag{25}$$

$$Precision\ (Pre) = \frac{TN}{TN+FP}100\% \tag{26}$$

$$F1-score = 2\frac{(precision)(recall)}{precision+recall} \tag{27}$$

### G. McNemar's STATISTICAL TEST

McNemar's test [27] is a nonparametric statistical test for paired nominal data. We can use McNemar's test to compare the predictive accuracy of two machine learning and deep learning models' performance. McNemar's test is based on a 2 times 2 contingency table of the two model's predictions on the test dataset. In the Table 5 the contingency table has been given. The total number of samples in the test set are n and $n = n_{00} + n_{01} + n_{10} + n_{11}$. The McNemars' test statistics ("chi-squared") can be computed in equation 28 as follows:

$$\chi^2 = \frac{(n_{01} - n_{10})^2}{n_{01} + n_{10}} \tag{28}$$

**TABLE 5.** Contingency Table.

| | Model2 correct | Model2 wrong |
|---|---|---|
| Model1 correct | $n_{11}$ | $n_{10}$ |
| Model1 wrong | $n_{01}$ | $n_{00}$ |

Approximately 1 year after McNemar [27] published the McNemar Test, Edwards [12] proposed a different continuity corrected version, which is the more commonly used variant today. Then McNemar's formula, corrected for continuity, may be written in equation 29 as follows:

$$\chi^2 = \frac{(|n_{01} - n_{10}| - 1)^2}{n_{01} + n_{10}} \tag{29}$$

We will use the continuity corrected version in our paper.

In McNemar's test, if the sum of cell n10 and n01 is sufficiently large, the $\chi^2$ value follows a chi-squared distribution with one degree of freedom. After setting a significance threshold $\alpha$, in our case $\alpha = 0.5$, we will compute the p-value, the p-value is the probability of observing this empirical (or a larger) chi-squared value. If the p-value is lower than our chosen significance level, we can reject the null hypothesis because the two model's performances are different. If the p-value is greater than our chosen significance level, we will accept the null hypothesis that the models' performances are equal. Mathematically we can summarize it as: If $p < \alpha$: then hypothesis $H_0$ are rejected, the model performances are not equal, If $p > \alpha$: then $H_0$ is accepted and the models have the same performance when trained on the specific training dataset.

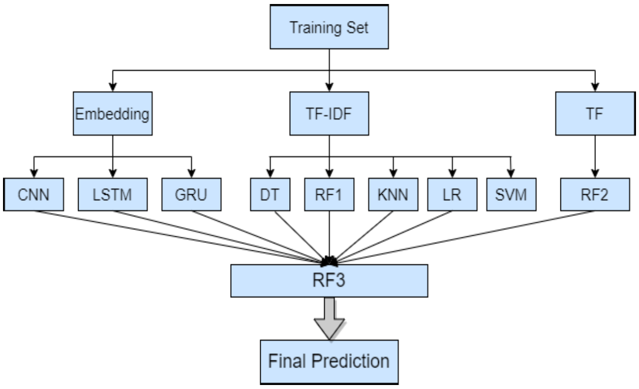


**FIGURE 3.** Proposed stacking method for fake news detection.

### H. PROPOSED STACKING MECHANISM

Stacking is one of the ensemble methods that connects multiple models of different types through a meta classifier to achieve better results. It can be seen as a more sophisticated version of cross-validation [43]. When we utilize stacking mechanism, we should ensure that each base learners must perform better than random guess and these base learners must be diverse. Otherwise, the stacking method may not be working.

In our work, we used the complete training set to train the eight base learners. To increase the diversity, we used machine learning models like SVM, LR, DT,KNN and RF and deep learning models like CNN, LSTM and GRU. We also used three different tokenization methods such as embedding, TF-IDF and TF. Then, the meta classifier, RF is fitted by using the prediction of each individual base models. Our proposed staking method is shown in 3.

## IV. EXPERIMENTAL RESULTS AND DISCUSSION

In this research work, we first removed the stopwords and urls from our dataset. Then, we used tokenization methods like TF, TF-IDF and embedding to obtain the text representation. After that, we trained individual models including five machine learning models such as LR, DT, KNN, RF and SVM and three deep learning models like LSTM,GRU, CNN on these text representation features. To choose the best individual model, we used a corrected version of McNemar' test to determine if the model with the highest accuracy has a significant difference with other models on both dataset. Finally, to improve the individual model performance, we proposed our stacking method of training another RF model based on the prediction results of all individual models. The experimentation on both ISOT and KDnugget datasets was performed by using Google Colab, a free cloud service supported by Google. The programming language is Python 3.7 and all experiments were performed by using different python libraries like tensorflow (for deep learning experiments) and

**TABLE 6.** Classifier performance on ISOT dataset. ∗ = p-value ≤ 0.05, NS = Not Significant.

| Classifiers | Model Performance evaluation metrics | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Parameters | tokenization methods | Acc (%) | Pre (%) | Recall (%) | F1-score(%) | Training time(s) | $\chi^2$ | p-value |
| SVM | C=1 | TF-IDF | 99.63 | 100 | 100 | 100 | 20 | 16.0 | * |
| LR | C=1 | TF-IDF | 99.63 | 100 | 99 | 100 | 2 | 16.0 | * |
| DT | max_depth=5 | TF-IDF | 99.60 | 100 | 100 | 100 | 86 | 13.22 | * |
| K-NN | K=9 | TF-IDF | 68.65 | 94 | 37 | 53 | 337 | 2789.06 | * |
| RF1 | n_estimators=400,max_depth=40 | TF-IDF | **99.87** | 100 | 100 | 100 | 225 | - | - |
| RF2 | n_estimators=300,max_depth=40 | TF | 99.84 | 100 | 100 | 100 | 160 | 0.57 | NS |
| LSTM | - | embedding | 99.74 | 100 | 100 | 100 | 1500 | 3.22 | NS |
| CNN | - | embedding | 99.52 | 100 | 100 | 100 | 45 | 20 | * |
| GRU | - | embedding | 99.69 | 100 | 100 | 100 | 14 | 6.61 | * |

scikit-learn (for machine learning experiments). In this work, for both ISOT and KDnugget datasets, 80% of the instances are used to train the classifiers and the remaining 20% of the datasets are used for testing the classifiers.

### A. MODEL CLASSIFICATION PERFORMANCE ON ISOT DATASET

The performance of machine learning and deep learning models have been checked on the bigger ISOT dataset in order to check if one model's performance is higher as compared to other models. All models had 100 percent performance in precison, recall, F1-score on this dataset, except K-NN.

According to Table 7, the classification performance of RF with TF-IDF (RF1) is highest among all machine learning and deep learning models with an accuracy of 99.87%.

Three deep learning models have only slightly lower accuracy than RF. Among these three deep learning models, LSTM has the highest accuracy. However, the low computation time is also an important criteria for best model selection. Considering that, the running time of GRU model, however, is only one hundredth of LSTM. So maybe GRU is a better option than LSTM.

When it comes to machine learning models, to find the best parameters, we used Grid Search method to train these models with different hyperparameters. According to Table 6, among six machine learning experiments, SVM with hyperparameters $C = 1$ and LR with essential hyperparameters $C = 1$ had the same high accuracy in all machine learning models. Their accuracy is only slightly lower than RF with TF-IDF. The KNN model with $k = 9$ has the highest performance as compared to other $k$ values. Due to this reason, we only reported the KNN performance when $k = 9$. However the KNN model's performance when $k = 9$ are still very low in all metrics as compared to other models due to curse of dimensionality and the running time is also much bigger. So on large dataset, we may should avoid using it. The classification performance of all models have been graphically reported in Figure 4 for better understanding.

**TABLE 7.** Performance Comparison on ISOT dataset.

| Metrics | Performance comparison |
|---|---|
| Accuracy | RF1>RF2>LSTM>GRU>SVM=LR>CNN>DT>KNN |
| Precision | RF1=RF2=LSTM=GRU=SVM=LR=CNN=DT=100>KNN |
| Recall | RF1=RF2=LSTM=GRU=SVM=CNN=DT=100>LR>KNN |
| F1-score | RF1=RF2=LSTM=GRU=SVM=LR=CNN=DT=100>KNN |

### B. MODEL CLASSIFICATION PERFORMANCE ON KDnugget DATASET

Both machine learning and deep learning models had also been checked on KDnugget dataset to choose the best model for fake news detection. As we mentioned before, KDnugget dataset has much less instances than ISOT dataset. On this small dataset, all machine learning and deep learning models had worse performance in all metrics as compared to their almost perfect performance on the big ISOT dataset. According to Table 8, LR with essential hyperparameters C=1 has the highest performance in all metrics among all machine learning and deep learning models.

Deep learning models usually need to train a lot of parameters and need more data which explains their bad performance on this small dataset. According to Table 9, LSTM model has only seventh best accuracy now. GRU model has the best performance in all metrics among three deep learning models but still was no better than both random forest models.

Machine learning models also had performed very differently on this small dataset. Support Vector Machine's accuracy is almost as high as Logistic Regression. Besides that, the performance of KNN is higher than Decision Tree on KDnugget dataset in all metrics now. We all know that K-NN algorithm has limitations that it can suffer from curse of dimensionality. As we expected, KNN model has much higher performance on small KDnugget dataset than on big ISOT dataset. On the other hand, the performance of Decision Tree decrease significantly on this small dataset. This is because Decision Tree is a weak leaner and it needs big data to make decisions. Therefore, on small datasets, we may should avoid using Decision Tree. The classification performance

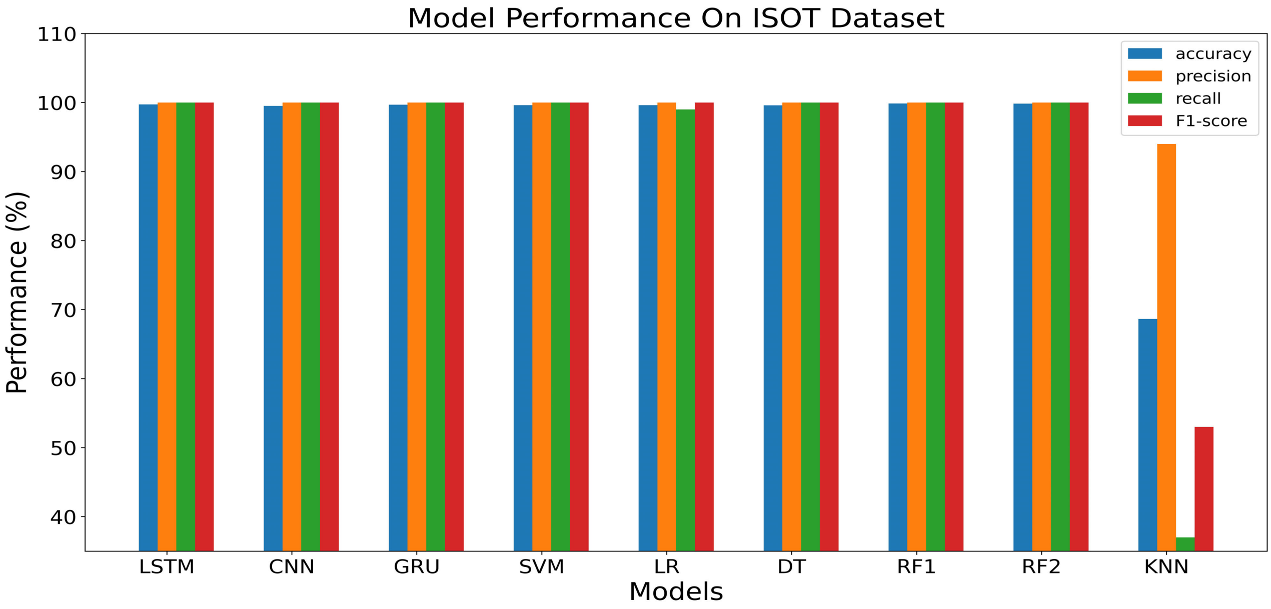


**FIGURE 4.** Performance of models on ISOT dataset.

**TABLE 8.** Classifier performance on KDnugget Dataset. $*$ = p-value $\leq$ 0.05, NS = Not Significant.

| Classifiers | Model performance evaluation metrics | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | parameters | tokenization methods | Acc (%) | Pre (%) | Recall (%) | F1-score(%) | Trainiing time(s) | $\chi^2$ | p-value |
| SVM | C=1 | TF-IDF | 92.42 | 93 | 93 | 93 | 11 | 0.94 | NS |
| LR | C=10 | TF-IDF | **92.82** | 93 | 93 | 93 | 1 | - | - |
| DT | max_depth=5 | TF-IDF | 79.87 | 83 | 76 | 80 | 25 | 108.89 | * |
| K-NN | K=9 | TF-IDF | 82.56 | 77 | 94 | 85 | 12 | 80.78 | * |
| RF1 | n_estimators=150 | TF-IDF | 91.63 | 91 | 93 | 92 | 12 | 0.70 | NS |
| RF2 | n_estimators=200 | TF | 91.48 | 91 | 93 | 92 | 16 | 13.24 | * |
| LSTM | - | embedding | 88.95 | 89 | 89 | 89 | 450 | 17.32 | * |
| CNN | - | embedding | 89.50 | 90 | 89 | 89 | 0.0033 | 11.67 | * |
| GRU | - | embedding | 91.32 | 91 | 91 | 91 | 2 | 2.59 | NS |

of all models have been graphically reported in Figure 5 for better understanding.

### *C. McNemar's STATISTICAL TEST FOR MODELS PERFORMANCE COMPARISON*

Model's accuracy is a common metrics that researchers used to select the models. As we can see from Figure 6, there are a lot models that had very similar accuracy. Thus, we employed a corrected version of McNemar's test to determine if models' performance has significant difference in accuracy. As we can see from Table 6 and Table 8, RF1 has the highest accuracy in ISOT dataset and LR has the highest accuracy in KDnugget dataset. Therefore, we computed $\chi^2$ and p-value between RF1 and other models and put the results into Table 6. Since LR has the highest accuracy on KDnugget dataset, we computed $\chi^2$ and p-value between LR and other models and put the results into Table 8.

**TABLE 9.** Performance Comparison on KDnugget dataset.

| Metrics | Performance comparison |
|---|---|
| Accuracy | LR>SVM>RF1>RF2>GRU>CNN>LSTM>KNN>DT |
| Precision | LR=SVM>RF1=RF2=GRU>CNN>LSTM>DT>KNN |
| Recall | KNN>LR=SVM=RF1=RF2>GRU>CNN=LSTM>DT |
| F1-score | LR=SVM>RF1=RF2>GRU>LSTM=CNN>KNN>DT |

In our experiments, the value of significance level alpha is 0.05, and the confidence level is 0.95. Based on p-value and alpha, we accept or reject the null hypothesis. If p-value < $\alpha$: then $H_0$ is rejected, the models has significantly different performance. If p-value >= $\alpha$: then $H_0$ is accepted and the models have the same performance.

In Table 6, SVM, LR, DT, KNN, CNN and GRU have a p-value less than 0.5, so their performance is significantly different from LSTM model at the 0.05 significance level

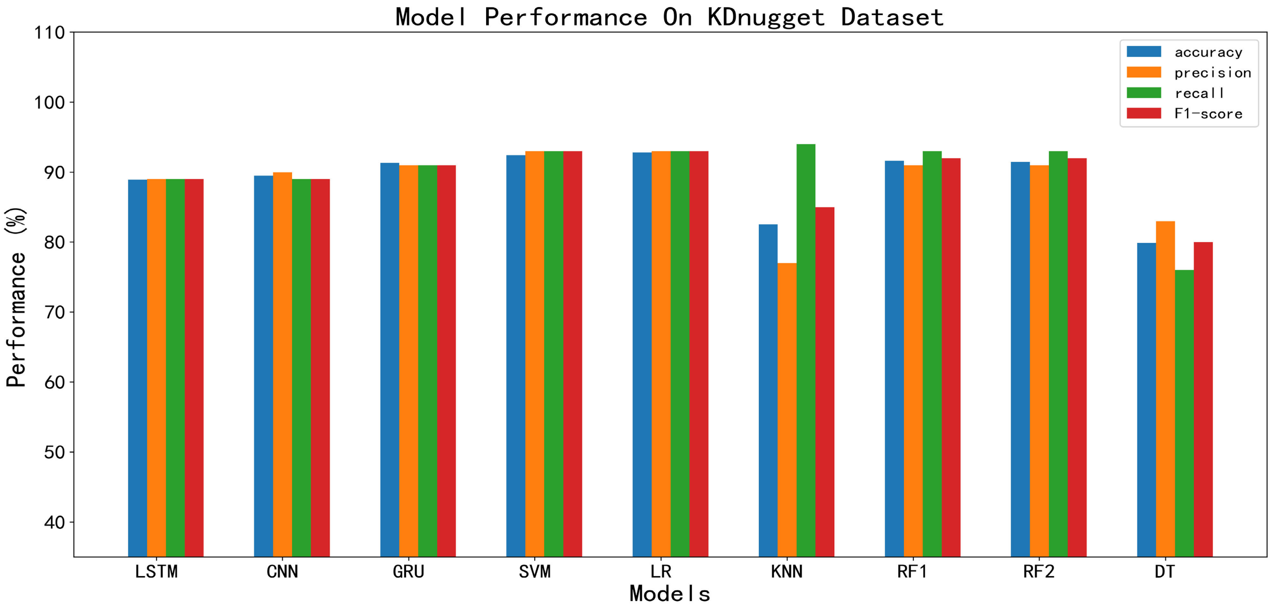


**FIGURE 5.** Performance of models on KDnugget dataset.

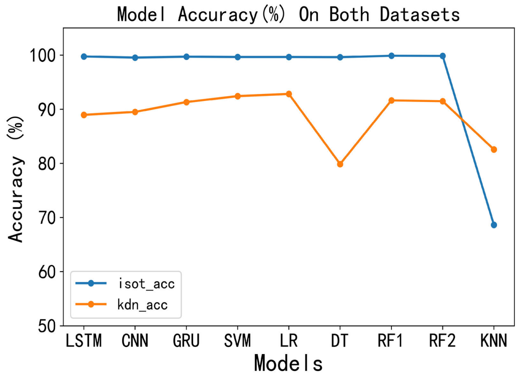


**FIGURE 6.** Individual model performance comparison on both datasets.

**TABLE 10.** The performance of our proposed stacking model.

| Datasets | Acc (%) | Pre (%) | Recall (%) | F1-score (%) |
|---|---|---|---|---|
| IOST | 99.94 | 100 | 100 | 100 |
| KDnugget | 96.05 | 97 | 96 | 96 |

which means their performances are worse. On the other hand, RF2 and LSTM have a p-value that are greater than 0.05. So their performance is as good as RF1's. According to the p-value from Table 8, only SVM, RF1 and GRU have a p-value that are less than 0.05. Therefore we can conclude that their performance is just as good as LR's performance and other models have worse performance on this dataset.

In a nutshell, RF1, RF2 and LSTM have the best performance on the bigger dataset ISOT when the significance level is 0.05. LR, SVM, RF1 and GRU have the best performance on the smaller KDnugget dataset. Thus, the best individual model on these two datasets is Random Forest with TF-IDF tokenization method.

**TABLE 11.** Performance comparison of the proposed method with the baselines methods.

| Reference | dataset | Accuracy (%) |
|---|---|---|
| [3] | ISOT | 92 |
| [31] | ISOT | 96.8 |
| [15] | ISOT | 99.8 |
| [23] | ISOT | 99.86 |
| **proposed model** | ISOT | 99.94 |
| [6] | KDnugget | 92.7 |
| [13] | KDnugget | 94 |
| **proposed model** | KDnugget | 96.05 |

### *D. RESULTS OF THE PROPOSED STACKING MECHANISM*

To further improve the results, we used the predictions of all individual models as the training data. Since RF is the best individual model among all machine learning and deep learning models, we used it to train the prediction data. According to Table 10, our proposed model has much better performance than all individual models on both datasets in all evaluation metrics. So we highly recommend it for fake news detection.

### *E. PERFORMANCE COMPARISON WITH BASELINE METHODS*

The proposed method for fake news detection have been compared with state of the art methods in Table 11. According to Table 11, in terms of accuracy our method achieved 99.94% accuracy on ISOT dataset and 96.05% on KDnugget dataset which is so much better than existing methods. Therefore, the proposed method is highly recommended for detection of fake news and it could be easily employed in real environment.

**TABLE 12.** Mathematical symbols and notations used in this paper.

| Symbol | Description |
|---|---|
| $c_{(t)}$ | long term state at t time step |
| $h_{(t)}$ | short term state at t time step |
| $y_{(t)}$ | output at t time step |
| $i_{(t)}$ | input gate controller |
| $f_{(t)}$ | forget gate controller |
| $o_{(t)}$ | output gate controller |
| $n$ | total number of instances in dataset, support vector |
| $x$ | Input |
| $y$ | classe label |
| $b$ | bais, offset value from the origin |
| $w$ | d-dimensional coefficient vector |
| $x_i$ | ith instance of dataset sample X |
| $y_i$ | target labels to $x_i$ |
| p-value | probability |
| $\alpha$ | significance level |
| $K$ | kernel |
| $I$ | image |
| $M_{m,n}$ | matrix of m rows and n columns |

## V. CONCLUSION AND FUTURE WORK

In this paper, we evaluated five machine learning models and three deep learning models on two fake news datasets of different size in terms of accuracy, precision, recall, F1-score. According to our experiments, some models like K-Nearest Neighbors had better performance on small dataset and other models like Decision Tree, Support Vector Machine, Logistic Regression, CNN, GRU, LSTM had a lot worse performance on small datasets. To select the best model, we used a corrected version of McNemar's test to determine if models' performance is significantly different. According to our final experiments, among all individual models, Random Forest with TF-IDF has the highest accuracy on the ISOT dataset and Logistic Regression with TF-IDF has the highest accuracy on the KDnugget dataset.

The experimental results of these two best models demonstrated that our proposed stacking method achieved 99.94% accuracy on the ISOT dataset and 96.05% accuracy on the KDnugget dataset was very high as compared to individual models. We also compared our results to other existing work and concluded that our stacking model is much better. Due to the high performance of our proposed stacking methods, we recommend it for the detection of fake news.

The major innovation of this research work as follows: Firstly, five machine learning and three deep learning models have been trained in order to compare the performance difference between machine learning and deep learning models. Secondly, we used two datasets of different size for evaluation of the proposed method to test models' robustness on datasets of different size. Thirdly we employed a corrected version of McNemar's statistical test to decide if there really are significant differences between two model's performance and determine the best individual model for fake news detection. Lastly, we proposed a stacking model to improve the individual model performance.

In future, we will perform more experiments on other data sets in different languages. We will also try to use more different machine learning and deep learning models for fake news detection. We will also collect more fake and real news data in different language to detect fake news in different countries.

## CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

## APPENDIX

The mathematical notations used in paper are given in Table 12.

## REFERENCES

[1] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapè, "MIMETIC: Mobile encrypted traffic classification using multimodal deep learning," *Comput. Netw.*, vol. 165, Dec. 2019, Art. no. 106944.

[2] G. E. R. Agudelo, O. J. S. Parra, and J. B. Velandia, "Raising a model for fake news detection using machine learning in Python," in *Proc. Conf. e-Bus., e-Services e-Soc.* Cham, Switzerland: Springer, 2018, pp. 596–604.

[3] H. Ahmed, I. Traore, and S. Saad, "Detecting opinion spams and fake news using text classification," *Secur. Privacy*, vol. 1, no. 1, p. e9, Jan. 2018.

[4] O. Ajao, D. Bhowmik, and S. Zargari, "Fake news identification on Twitter with hybrid CNN and RNN models," in *Proc. 9th Int. Conf. Social Media Soc.*, Jul. 2018, pp. 226–230.

[5] M. Amjad, G. Sidorov, A. Zhila, H. Gómez-Adorno, I. Voronkov, and A. Gelbukh, "'Bend the truth': Benchmark dataset for fake news detection in urdu language and its evaluation," *J. Intell. Fuzzy Syst.*, vol. 39, no. 2, pp. 2457–2469, 2020.

[6] S. D. Bhattacharjee, A. Talukder, and B. V. Balantrapu, "Active learning based news veracity detection with feature weighting and deep-shallow fusion," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 556–565.

[7] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, p. 27, 2011.

[8] H.-L. Chen, B. Yang, J. Liu, and D.-Y. Liu, "A support vector machine classifier with rough set-based feature selection for breast cancer diagnosis," *Expert Syst. Appl.*, vol. 38, no. 7, pp. 9014–9022, Jul. 2011.

[9] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, *arXiv:1406.1078*. [Online]. Available: http://arxiv.org/abs/1406.1078

[10] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernelbased Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[11] M. L. Della Vedova, E. Tacchini, S. Moret, G. Ballarin, M. DiPierro, and L. de Alfaro, "Automatic online fake news detection combining content and social signals," in *Proc. 22nd Conf. Open Innov. Assoc. (FRUCT)*, May 2018, pp. 272–279.

[12] A. L. Edwards, "Note on the 'correction for continuity' in testing the significance of the difference between correlated proportions," *Psychometrika*, vol. 13, no. 3, pp. 185–187, 1948.

[13] P. H. A. Faustini and T. F. Covões, "Fake news detection in multiple platforms and languages," *Expert Syst. Appl.*, vol. 158, Nov. 2020, Art. no. 113503.

[14] S. Gilda, "Evaluating machine learning algorithms for fake news detection," in *Proc. IEEE 15th Student Conf. Res. Develop. (SCOReD)*, Dec. 2017, pp. 110–115.

[15] M. H. Goldani, S. Momtazi, and R. Safabakhsh, "Detecting fake news with capsule neural networks," 2020, *arXiv:2002.01030*. [Online]. Available: http://arxiv.org/abs/2002.01030

[16] A. U. Haq, J. Li, M. H. Memon, J. Khan, and S. U. Din, "A novel integrated diagnosis method for breast cancer detection," *J. Intell. Fuzzy Syst.*, vol. 38, no. 2, pp. 2383–2398, Feb. 2020.

[17] A. U. Haq, J. Li, M. H. Memon, J. Khan, S. U. Din, I. Ahad, R. Sun, and Z. Lai, "Comparative analysis of the classification performance of machine learning classifiers and deep neural network classifier for prediction of Parkinson disease," in *Proc. 15th Int. Comput. Conf. Wavelet Act. Media Technol. Inf. Process. (ICCWAMTIP)*, Dec. 2018, pp. 101–106.

[18] A. U. Haq, J. P. Li, M. H. Memon, J. Khan, A. Malik, T. Ahmad, A. Ali, S. Nazir, I. Ahad, and M. Shahid, "Feature selection based on L1-norm support vector machine and effective recognition system for Parkinson's disease using voice recordings," *IEEE Access*, vol. 7, pp. 37718–37734, 2019.

[19] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[20] A. Jain, A. Shakya, H. Khatter, and A. K. Gupta, "A smart system for fake news detection using machine learning," in *Proc. Int. Conf. Issues Challenges Intell. Comput. Techn. (ICICT)*, vol. 1, Sep. 2019, pp. 1–4.

[21] R. K. Kaliyar, A. Goswami, and P. Narang, "Multiclass fake news detection using ensemble machine learning," in *Proc. IEEE 9th Int. Conf. Adv. Comput. (IACC)*, Dec. 2019, pp. 103–107.

[22] R. K. Kaliyar, A. Goswami, P. Narang, and S. Sinha, "FNDNet—A deep convolutional neural network for fake news detection," *Cognit. Syst. Res.*, vol. 61, pp. 32–44, Jun. 2020.

[23] S. Kula, M. Choraś, R. Kozik, P. Ksieniewicz, and M. Woźniak, "Sentiment analysis for fake news detection by means of neural networks," in *Proc. Int. Conf. Comput. Sci.* Cham, Switzerland: Springer, 2020, pp. 653–666.

[24] S. Kumar, R. Asthana, S. Upadhyay, N. Upreti, and M. Akbar, "Fake news detection using deep learning models: A novel approach," *Trans. Emerg. Telecommun. Technol.*, vol. 31, no. 2, p. e3767, Feb. 2020.

[25] Y. Long, "Fake news detection through multi-perspective speaker profiles, version I17-2043," Assoc. Comput. Linguistics, Stroudsburg, PA, USA, Tech. Rep., Nov. 2017, vol. 2. [Online]. Available: http://repository.essex.ac.uk/27757/

[26] J. Ma, W. Gao, P. Mitra, S. Kwon, B. J. Jansen, K.-F. Wong, and M. Cha, "Detecting rumors from microblogs with recurrent neural networks," in *Proc. 25th Int. Joint Conf. Artif. Intell.*, 2016, pp. 3818–3824.

[27] Q. McNemar, "Note on the sampling error of the difference between correlated proportions or percentages," *Psychometrika*, vol. 12, no. 2, pp. 153–157, Jun. 1947.

[28] F. Monti, F. Frasca, D. Eynard, D. Mannion, and M. M. Bronstein, "Fake news detection on social media using geometric deep learning," 2019, *arXiv:1902.06673*. [Online]. Available: http://arxiv.org/abs/1902.06673

[29] A. Montieri, D. Ciuonzo, G. Bovenzi, V. Persico, and A. Pescape, "A dive into the dark Web: Hierarchical traffic classification of anonymity tools," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 3, pp. 1043–1054, Jul. 2020.

[30] R. Oshikawa, J. Qian, and W. Y. Wang, "A survey on natural language processing for fake news detection," 2018, *arXiv:1811.00770*. [Online]. Available: http://arxiv.org/abs/1811.00770

[31] F. A. Ozbay and B. Alatas, "Fake news detection within online social media using supervised artificial intelligence algorithms," *Phys. A, Stat. Mech. Appl.*, vol. 540, Feb. 2020, Art. no. 123174.

[32] J.-P. Posadas-Durán, H. Gómez-Adorno, G. Sidorov, and J. J. M. Escobar, "Detection of fake news in a new corpus for the spanish language," *J. Intell. Fuzzy Syst.*, vol. 36, no. 5, pp. 4869–4876, May 2019.

[33] T. Rasool, W. H. Butt, A. Shaukat, and M. U. Akram, "Multi-label fake news detection using multi-layered supervised learning," in *Proc. 11th Int. Conf. Comput. Automat. Eng. (ICCAE)*, 2019, pp. 73–77.

[34] J. C. S. Reis, A. Correia, F. Murai, A. Veloso, and F. Benevenuto, "Explainable machine learning for fake news detection," in *Proc. 10th ACM Conf. Web Sci. (WebSci)*, 2019, pp. 17–26.

[35] J. C. S. Reis, A. Correia, F. Murai, A. Veloso, F. Benevenuto, and E. Cambria, "Supervised learning for fake news detection," *IEEE Intell. Syst.*, vol. 34, no. 2, pp. 76–81, Mar. 2019.

[36] A. Roy, K. Basak, A. Ekbal, and P. Bhattacharyya, "A deep ensemble framework for fake news detection and classification," 2018, *arXiv:1811.04670*. [Online]. Available: http://arxiv.org/abs/1811.04670

[37] N. Ruchansky, S. Seo, and Y. Liu, "CSI: A hybrid deep model for fake news detection," in *Proc. ACM Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 797–806.

[38] S. Shabani and M. Sokhn, "Hybrid machine-crowd approach for fake news detection," in *Proc. IEEE 4th Int. Conf. Collaboration Internet Comput. (CIC)*, Oct. 2018, pp. 299–306.

[39] S. Singhania, N. Fernandez, and S. Rao, "3HAN: A deep neural network for fake news detection," in *Proc. Int. Conf. Neural Inf. Process.* Cham, Switzerland: Springer, 2017, pp. 572–581.

[40] M. Umer, Z. Imtiaz, S. Ullah, A. Mehmood, G. S. Choi, and B.-W. On, "Fake news stance detection using deep learning architecture (CNN-LSTM)," *IEEE Access*, vol. 8, pp. 156695–156706, 2020.

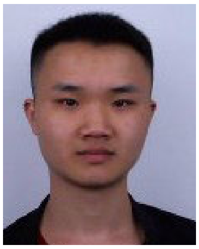
[41] P. W. Wagacha, "Induction of decision trees," *Found. Learn. Adapt. Syst.*, vol. 12, pp. 1–14, May 2003.

[42] W. Y. Wang, "'Liar, liar pants on fire': A new benchmark dataset for fake news detection," 2017, *arXiv:1705.00648*. [Online]. Available: http://arxiv.org/abs/1705.00648

[43] D. H. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, no. 2, pp. 241–259, Jan. 1992.

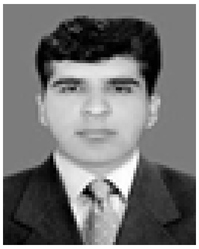
[44] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg, "Top 10 algorithms in data mining," *Knowl. Inf. Syst.*, vol. 14, no. 1, pp. 1–37, 2008.

[45] X. Zhang and A. A. Ghorbani, "An overview of online fake news: Characterization, detection, and discussion," *Inf. Process. Manage.*, vol. 57, no. 2, Mar. 2020, Art. no. 102025.



**TAO JIANG** is currently pursuing the M.S. degree with the School of Computer Science and Engineering, UESTC, China. His research interests include machine learning, deep learning, big data analysis, IoT, E-Health, and concerned technologies and algorithms.

**JIAN PING LI** is currently the Chairman of Computer Science and Engineering College and the Model Software College, University of Electronic Science and Technology of China. He is also the Director of the International Centre for Wavelet Analysis and its Applications. He is the Chief Editor of International Progress on Wavelet Active Media Technology and Information Processing. He is also an Associate Editor of *International Journal of Wavelet, Multiresolution and Information Processing*. He served in the National Science and Technology Award Evaluation Committee, the National Natural Science Foundation Committee of China, and the Ministry of Public Security of the People's Republic of China, such as a Technical Advisor and a dozen academic and social positions.



**AMIN UL HAQ** is currently working as a Postdoctoral Scientific Research Fellow with the University of Electronic Science and Technology of China (UESTC), China. He is also a Lecturer with Agricultural University Peshawar, Pakistan. He is also associated with the Wavelets Active Media Technology and Big Data Laboratory, as a Postdoctoral Scientific Research Fellow. He has a vast academic, technical and professional experience in Pakistan. He has been published high level research articles in good journals. His research interests include machine learning, deep learning, medical big data, IoT, E-Health and telemedicine, and concerned technologies and algorithms. He is an Invited Reviewer for numerous world-leading high-impact journals (reviewed more than 40 journal articles to date).

**ABDUS SABOOR** is currently pursuing the M.S. degree with the School of Computer Science and Engineering, UESTC, China. He is currently a Lecturer with Government University Peshawar, Pakistan. His research interests include machine learning, medical big data, IoT, E-Health and telemedicine,and concerned technologies and algorithms.

**AMJAD ALI** received the Ph.D. degree in real time systems from Gyeongsang National University, South Korea. He is currently an Assistant Professor and the Chairman of the Department of Computer Science and Software Technology with University of Swat. He published several research papers in international journals and conferences.