IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 10, 2020, accepted August 24, 2020, date of publication August 26, 2020, date of current version September 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3019735

Fake News Stance Detection Using Deep Learning Architecture (CNN-LSTM)

MUHAMMAD UMER[1], ZAINAB IMTIAZ[1], SALEEM ULLAH[1], ARIF MEHMOOD[2], GYU SANG CHOI[3], AND BYUNG-WON ON[4]

[1]Department of Computer Science, Khwaja Fareed University of Engineering and Information Technology, Rahim Yar Khan 64200, Pakistan
[2]Department of Computer Science and Information Technology, The Islamia University of Bahawalpur, Bahawalpur 63100, Pakistan
[3]Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38542, South Korea
[4]Department of Statistics and Computer Science, Kunsan National University, Gunsan 54150, South Korea

Corresponding authors: Gyu Sang Choi (castchoi@ynu.ac.kr) and Byung-Won On (on.byung.won@gmail.com)

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2019R1A2C1006159) and MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2016-0-00313) supervised by the IITP (Institute for Information & communications Technology Promotion), and in part by the Fareed Computing Research Center, Department of Computer Science, Khwaja Fareed University of Engineering and Information Technology (KFUEIT), Rahim Yar Khan, Pakistan.

ABSTRACT Society and individuals are negatively influenced both politically and socially by the widespread increase of fake news either way generated by humans or machines. In the era of social networks, the quick rotation of news makes it challenging to evaluate its reliability promptly. Therefore, automated fake news detection tools have become a crucial requirement. To address the aforementioned issue, a hybrid Neural Network architecture, that combines the capabilities of CNN and LSTM, is used with two different dimensionality reduction approaches, Principle Component Analysis (PCA) and Chi-Square. This work proposed to employ the dimensionality reduction techniques to reduce the dimensionality of the feature vectors before passing them to the classifier. To develop the reasoning, this work acquired a dataset from the Fake News Challenges (FNC) website which has four types of stances: agree, disagree, discuss, and unrelated. The nonlinear features are fed to PCA and chi-square which provides more contextual features for fake news detection. The motivation of this research is to determine the relative stance of a news article towards its headline. The proposed model improves results by $\sim$ 4% and $\sim$ 20% in terms of *Accuracy* and $F1 - score$. The experimental results show that PCA outperforms than Chi-square and state-of-the-art methods with 97.8% accuracy.

INDEX TERMS Fake news detection, text mining, deep learning, PCA, Chi-square, CNN-LSTM, word embedding.

I. INTRODUCTION

In the age of technology, a tremendous amount of data is being generated online every day. However, an unprecedented amount of the data flooded on the Internet is fake news, which is generated to attract the audience, to influence beliefs and decisions of people [1]–[3], to increase the revenue generated by clicking [4], and to affect major events such as political elections [5]. Readers are misguided by deliberately spreading false information. Obtaining and spreading information through social media platforms has become extremely trouble-free, which makes it difficult and nontrivial to detect based merely on the content of news. For example, some reports illustrate that Russia has created fake accounts and social bots to spread fake news. According to a research poll, 64% of US citizens reported that fake news has caused a ''great deal of confusion'' about the factual content of reported events [6]. On top of that, large-scale false information cascade has increasingly harmful consequences in the field of business, marketing, and stock-share. For instance, in 2013, 130 billion dollars were wiped out in stock value after a false news spread on twitter that Barack Obama was injured in an explosion [5]. In the US presidential campaign of 2016, fake news has been accused of being foremost contributing factor of the increasing political polarization and partisan conflict as well as affecting the outcome [7]–[9]. Thus, it goes without saying that fake news identification is undeniably a grave challenge for the news industry and

The associate editor coordinating the review of this manuscript and approving it for publication was Keli Xiao.

journalists and the tools for detection of fake news have become dire necessity.

As manual fact checking is a very tedious task, automatically identification of fake news has drawn considerable attention in the Natural Language Processing (NLP) community to help alleviate the burdensome and time-consuming human activity of fact checking [10], [11]. Despite that, the task of evaluating the authenticity of news remains very complex even for automated systems [12]. Identifying fake news articles by understanding what other news organizations are reporting about the same topic could be a valuable first step. This step is known as Stance detection. Stance detection has always been an important foundation for various tasks, such as analyzing online debates [13]–[15], determining the authenticity of rumors on twitter [16], [17], or understanding the argumentative structure of persuasive essays [18].

In order to encourage the development of automated fake news detection tools using AI technology and machine learning, Pomerleau and Rao (2017) organized the first Fake News Challenge (FNC-1) [19] to evaluate what a news source is saying about a particular issue. Around 50 teams participated from both industry and academia in this challenge. The purpose of the FNC-1 challenge is to determine the stance of a news article relative to a given headline. There can be four types of stances of an article. It can either agree or disagree with the headline, discuss the same topic, or it is unrelated. The information on the FNC-1 task, its rules, the dataset, and the evaluation metrics is given on their official website [19]. Table 1 shows four example documents elaborating these stances.

TABLE 1. Headline and text bodies with respective stances from FNC Dataset.

Headline	Seven girls fall pregnant after five-days school trip in Bosnia and Herzegovina.
Agree	Seven girls, aged 13 to 15, have fallen pregnant after a five-days school trip to their country's capital city and their parents are being blamed [..].
Disagree	On December 17, a site called In Serbia.info published a 400-word piece under the sensational headline: BiH: Seven Primary School Students Pregnant After Five-Days Excursion[..].
Discuss	School field trips aren't what they used to be. Just ask the furious parents of seven teenage girls who became pregnant on a five-days school trip to Sarajevo, the capital city of Bosnia[..].
Unrelated	Lebanese authorities are holding a daughter and an ex-wife of the head of the ISIS jihadist group, Abu Bakr al-Baghdadi, the interior minister said. It was initially reported that a wife and son of the self-proclaimed had been arrested in November[..].

Deep learning models such as recurrent neural networks (RNN) and its variants [20]–[22] and convolution neural networks (CNN) [23] have been used effectively in many NLP tasks that share similarities to fake news and consist of calculating semantic similarity between sentences [24], [25] and community based question answering [26], [27]. In [28], Siamese MaLSTM is used to compute the semantic similarity of question pairs. A deep neural network converts the text sequence into fixed length vector representation which is then used to measure the relevance of two textual sequence, which is the relevance of each headline-body pair in our case [9], [29]–[31].

In this paper, we propose a model that automatically classifies the news articles with stance labels of either agree, disagree, unrelated, or discuss. The classification is done based on the level of agreement between the headline and body assigned to headlines. The proposed methodology is based on the observations to find the relevance of articles, which can be found using keywords within headlines. Some keywords in the headlines are useful for identifying key sentences in the body of the article. As shown in Table 1, only the first body is related to the headline rest do not have much relevance to the headline. Keywords such as 'seven girls', 'pregnant' are used to retrieve all the bodies related to these keywords and then classify them.

In the proposed model, first the feature set is passed with and without preprocessing to the embedding layer for conversion of features to word-vectors. Another set of experiments is carried out by using PCA and Chi-square, to perform component level analysis and obtain the reduced feature set.

One of the most popular statistical techniques for feature selection is PCA [32]. The discriminative power of the classifiers can be enhanced by utilizing PCA. It has many applications in face recognition, text categorization, and image compression [32]. The crux of PCA is to transform the original variables into a subset of variables by computing the highest correlation of original variables [33]. Principal Component Analysis (PCA) is a widely used technique that uses a linear transformation to reduce the dimensions of a feature set. The resulting dataset is simplified but it retains the characteristics of the original data set [34]–[36]. The new dataset might have an equal or lesser number of features than the original dataset. The co-variance matrix is used to compute the principal components.

After obtaining the features through any of the above mentioned methods, the features set is passed to the embedding layer of the deep model. The embedding layer vectorizes all the features and then these vector are fed to a 1 dimensional $(1-D)$ Convolution Neural Network (CNN) layer that further extracts the useful features by applying 64 filters of 5 different dimensions. The extracted features are fed to the max-pooling layer to select the features with the highest importance value during the computation. Remaining useful features are passed to the LSTM layer for sequence modeling and to find the hidden relevance of keywords and bodies.

We compare the effectiveness of the feature reduction based methods used with two deep learning models i.e. CNN and LSTM. The experimental results indicate that the proposed model improves the F1-score and accuracy by 20% and 4% respectively when used with the reduced feature set, than the other techniques discussed in the related work section.

The rest of the paper is structured as follows. Section II describes state-of-the-art-works related to this work. Section III gives a summary of the dataset, preprocessing steps performed on the dataset. Section IV illustrates the brief explanation of the deep learning model, experiment details, and machine specifications used for the experiment. Section V discuss the model performance evaluation metrics. Section VI presents the results and discussion and finally section VII conclude the paper with possible future research directions.

II. RELATED WORK

Stance Detection is a well-established and well-researched task in NLP. It is defined as determining from the text whether the audience is in favor, against or neutral about the target [37]. Stance detection has become foundation for many tasks such as fake news detection [19], claim validation [38], and argument search [39]. Previous studies in fake news detection focused on target-specific stance prediction in which the stance of a text entity relating to a topic or a named entity is determined. In many researches, target-specific stance prediction is performed for tweets (where tweets are the text and target is single stance) [37], [40], [41] and online debates [13], [15], [40]. Such target-specific approaches are based on structural features [13], and linguistic and lexical features [15].

Stance prediction in tweets and online debates is different from stance detection in a news article in which the stance detection of a news article is relative to the headline in NLP. The authenticity of claims is predicted with the use of the stance of articles and the reliability of their sources in [38]. In detecting the reliability of fake news, stance features are used, which are also defined as unsupported claims [42]. A researcher used tweets publishing time and stances as the only features for determining the authenticity of tweets by using Hidden Markov Models [43]. Another study [44] provides an approach to the claim-relevance discovery problem by leveraging various information retrieval and machine learning techniques and yielding 91.6% accuracy.

The first fake news stance detection challenge was initiated back in 2017. The inspiration behind the FNC-1 stance detection task was taken from the work proposed in [45], in which they classify the stance of a single sentence of a news headline towards a specific claim. The dataset used in the FNC-1 challenge was partially labeled and based on the Emergent dataset [45]. In FNC-1, the stance is detected on document level in which the entire news article is classified relative to a headline. The top-performing system in FNC-1 is developed by Talos Research Intelligence team called SOLAT in the SWEN [46]. It is based on a 50/50 weighted average ensemble method that combines deep CNN with Google News pre-trained vectors, and gradient-boosted decision trees. The model achieved 82.02% accuracy.

The 1^{st} runner up 'Athene' team, consisting of members from the Ubiquitous Knowledge Processing Lab and the Adaptive Preparation of Information from Heterogeneous Sources Research Training Group at Technische Universität Darmstadt (TU Darmstadt), uses a multi-layer perceptron (MLP) as an ensemble of six hidden layers with hand-crafted features [47] and obtains 81.97% accuracy. The 2^{nd} runner up team, UCL Machine Reading (UCLMR), uses two bags of words (BOW), term frequency (TF), and term frequency-inverse document frequency (TF-IDF) as features and proposes a model consisting of MLP with 81.72% accuracy [48]. The fourth-best team extract both semantic embedding and lexical matching features and pass them to another gradient boosting trees. Additionally, a two-step logistic regression classifier [4] and an ensemble models of five classifiers [49] achieve 9^{th} and 11^{th} places respectively. All three challenge winners [46]–[48] in SemEval and FNC make use of both hand-crafted and neural network-based features with classification-based algorithms [50].

In another research, the focus was on predicting rumor news using an agreement aware article search. They developed an agreement-aware search framework designed to provide users with a holistic view of a question, for which the ground truth was not confident. They designed a two-step model consisting of a tree-based model based on handcrafted features and an RNN plus attention model focusing on only a few key sentences [51]. The proposed model in [50] is a single, end-to-end ranking-based algorithm with MLP. TF-IDF is used to extract features to represent both headlines and bodies of the news articles. The model obtains 86.66% accuracy on FNC-1.

In [12], a deep learning method is used for addressing the stance detection problem from the FNC-1 task. It incorporates bi-directional RNNs together with max-pooling and neural attention mechanisms to build representations from headlines and from the body of news articles and combine these representations with external similarity features. The use of pre-training and the combination of neural representations together with external similarity features produces 83.8% accuracy. Another work [9] uses deep recurrent model to compute the neural embedding, weighted n-gram bag-of-words model to compute the statistical features and feature engineering heuristics to extract hand crafted external features. Finally, all the features are combined, by using deep neural layer for the classification of the headline-body news pair as agree, disagree, discuss, or unrelated. The obtained accuracy is 89.29%. It is proved in [30] that neural network outperforms hand-crafted features. By implementing bilateral multi-perspective matching models (BiMPM) and improving the existing Attentive Reader with a full attention mechanism between words in body text and headlines makes the model able to achieve 86.5% accuracy. A Conditional Encoding LSTM model with attention yields a 80.8% score in [31]. In another work [29], a conditioned bidirectional LSTM with global features is used. It demonstrates that the combination of global features and local word embedding features is better at predicting the stance of headline-article pairs than each of them individually by obtaining 87.4% accuracy. Rather than using a classification-based method, this research tackles the news stance detection task by using a ranking-based method. The ranking-based method compares and maximizes the difference between the true and false stances of a given pair of headlines and article bodies. This approach results in 86.66% accuracy [50].

Recently, a novel stacked Bi-LSTM layers based approach was introduced containing a model consisting of stacked Bi-LSTM layers in [52] and novel stacked CNN was introduced in [53]. The LSTM layer is used for sequence modeling. Bi-LSTM contains information on both ends of the sentence which results in much better accuracy. In [54], many models were applied and tested on FNC-1. These models include CNN, LSTM, a combination of CNN and LSTM, and end-to-end memory networks. They also propose a novel extension of the general architecture based on a similarity-based matrix. Their works show that the proposed model sMemNN with TF achieves the highest accuracy of 88.57%. Whereas CNN+LSTM and LSTM+CNN show limited results by achieving 48.54% and 65.36% accuracy, respectively. The reason is that the data taken for training is 80% and for testing is 20%. In order to balance the data during training, equal instances of each class are randomly selected for each epoch. Furthermore, there is no mention of pooling layer in CNN architecture which might have caused low accuracy. A large-scale language model for stance detection was proposed which performed transfer learning on a Roberta deep bidirectional transformer language model [55]. The model achieved 93.71% accuracy on the Fake News Challenge Stage 1 (FNC-I) benchmark dataset.

However, the aforementioned researches that employ machine learning models make use of hand-crafted features. These features do not take the context of the text into account hence, produce limited results. In addition, most of the models are unsuccessful in obtaining adequate detection performance for the agree and disagree classes. To overcome these limitations, we employ CNN and LSTM layers along with dimensionality reduction techniques including PCA and Chi-square. Overall, the proposed pipeline leads to better results than other described deep learning strategies by resulting in 97.8% accuracy.

III. DATASET AND PREPROCESSING

A. DATASET

The benchmark dataset of Fake News Challenges was collected from the official website [56]. The FNC dataset consists of 75, 385 labeled instances and 2, 587 article bodies, which relate to 300 headlines approximately, and for each claim, there are 5 to 20 news articles. Of these headlines, 7.4% are agreed, 2.0% are disagreed, 17.7% are discussed and 72.8% are unrelated as shown in Table 2. The claims related to the articles' bodies are labeled manually. The details of the labels are as follows:

- *Agree*: There is a relation between headline and article body.
- *Disagree*: There is no relation between headline and article body.
- *Discuss*: There is a little bit of match between headline and article body, taking it as neutral.
- *Unrelated*: The topic discussed in headline and body are completely different.

TABLE 2. Dataset statistics.

Dataset	Headlines	Tokens	Instances	Agree	Disagree	Discuss	Unrelated
FNC-1	2,587	372	75,385	7.4%	2.0%	17.7%	72.8%

Dataset has been divided into 49, 972 and 25, 413 instances for training and testing respectively. This distribution of training and testing data is made based on the rules mentioned for FNC-1 challenge. In training data, the headlines are 1, 648 and the bodies of the articles are 1, 683. The test data contains around 880 headlines with 904 articles bodies.

B. PRE-PROCESSING

Pre-processing is a data mining technique that transforms incomplete and inconsistent raw data into a machine-understandable format. Several tasks for texts pre-processing were performed on FNC-1 dataset. In order to perform these tasks, NLP techniques such as the conversion of the texts' characters to lowercase letters, stopwords removal, stemming, and tokenization was applied, with the application of algorithms available in Keras's library.

Stopwords are very common words that exist in the text and have very minor importance in terms of features and are irrelevant for this work e,g 'of', 'the', 'and', 'an', etc. By removing the stopwords, we reduce the processing time and save space otherwise taken by meaningless words mentioned above. In the text, words having similar meanings can occur more than once e.g. games and games. If so, reducing the words to a common basic form is very effective. This process is known as stemming and it is performed with the open-source implementation of the NLTK's Porter stemmer algorithm.

After the execution of the pre-processing steps described above, the number of terms in the headlines was reduced to 372. The tokenizer function from Keras's library was used to split each headline into a vector of words. Once the pre-processing is done, we use word embedding (word2vec) to map word/text to a list of vectors. Finally, a dictionary of the 5, 000 uni-gram words of headlines and article bodies is created. The length of all the headlines is set to the maximum length of the headline. The headlines with a length smaller than maximum length are zero-padded. Next, the features are fed into the hybrid Neural Network architecture consisting of CNN [57] and LSTM layers.

IV. PROPOSED METHODOLOGY

A. PROPOSED MODEL

The utmost contribution of this work is to propose a feature reduction techniques along with hybrid deep learning models, involving two neural network layers, i.e., CNN and LSTM. The proposed approach produces higher predictive performance when compared to the traditional deep learning models. To analyze the relationship, four data models are developed. In the first model, all the features are used without preprocessing for classification. In the second model, the non-reduced features set is used after preprocessing.

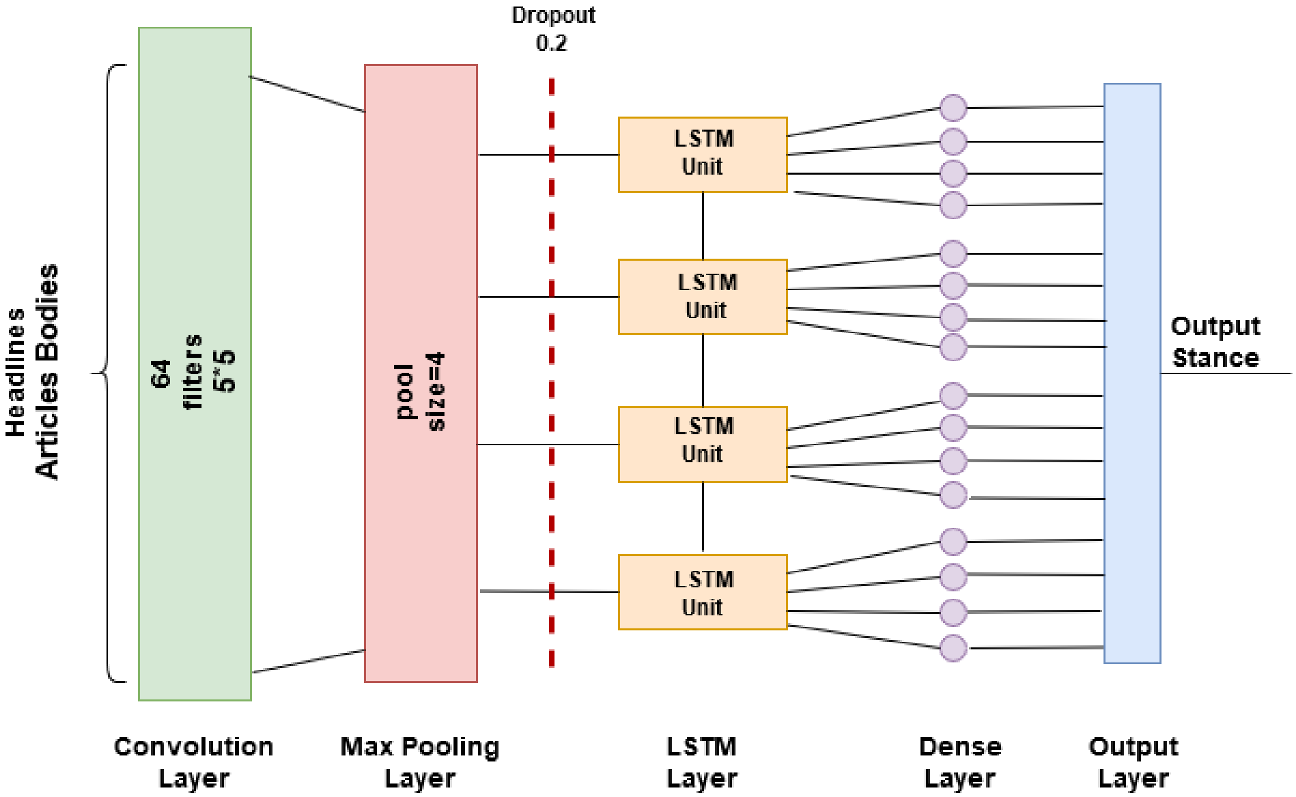


FIGURE 1. Proposed model architecture diagram.

Model third and fourth is developed by using dimensionality reduction [58], [59] techniques including PCA and Chi-square. This work further investigates which of these models are most suited for being used in conjunction with hybrid CNN and LSTM model when dealing with text data.

After the features are selected by any of the four models discussed above, the selected features are fed to the CNN-LSTM architecture. The first layer of the model is the embedding layer that accepts the input headlines and article bodies and converts each word into a vector of size 100. The number of features is 5000, thus, this layer will output a matrix of size $5000 * 100$. The output matrix will contain weights that we get through matrix multiplication, to produces a vector for each word. These vectors are passed to the CNN layer to extract contextual features. The output of the CNN layer is fed into LSTM and then passed to a fully connected dense layer to produce a single stance as final output. The proposed model is shown in Fig. 1 is trained and tested on small batches of size 32.

B. DIMENSIONALITY REDUCTION METHODS

There are two ways to perform dimensionality reduction in text categorization: feature extraction and feature selection. In feature selection methods, the most significant and relevant features are retained and the remaining features are discarded [60]. On the other hand, in feature extraction methods, a new vector space with special characteristics is created by transforming the original vector space. The features are reduced in new vector space [32].

The advantage of reducing features is that the processing speed is reduced which in turn results in higher performance [61]. Feature reduction has a great impact on the text classification results [62]. Therefore, it is extremely crucial to choose the right selection algorithm to reduce dimensions. Information Gain (IG), Mutual Information (1v1I) [62], Gini Coefficient (GI), Term Frequency-Inverse Document Frequency (TF-IDF) [63], Principal Component Analysis (PCA) and Chi-Square Statistics (CHI) are some of the common feature reduction algorithms. To improve the scalability of the text classifier, PCA and Chi-square are two-dimensionality reduction approaches that are used in combination with deep learning models.

C. PRINCIPAL COMPONENT ANALYSIS (PCA)

Principal Component Analysis (PCA) is a widely used technique that uses a linear transformation to reduce the dimensions of a feature set. The resulting dataset is simplified but it retains the characteristics of the original data set [35]. The new dataset might have an equal or lesser number of features than the original dataset. The covariance matrix is used to compute the principal components. These components are arranged in decreasing order of importance [64]. Let us assume that the original matrix comprises 'a' dimensions and 'b' observations and it is required to reduce the dimensionality into a 't' dimensional subspace then its transformation can be given by the following equation.

$$Y = (E^Z X) \tag{1}$$

In above equation, $E_{a\times t}$ is the projection matrix which contains t eigen vectors corresponding to t highest eigen values, and where $X_{a\times b}$ is mean centered data matrix.

D. CHI-SQUARE

Chi-Square Statistics is one of the most effective feature selection algorithms [65] It is designed for testing relationships between categorical variables. It is used to estimate the lack of independence between a and b as well as compare to the chi-square distribution with one degree of freedom to judge extremeness [62], [66]. Test for independence and test for goodness of fit are two types of tests for which Chi-square is used. For feature selection, test for independence is implemented and the dependency of target label is examined on feature(s). Chi-square investigates the correlation of the features. The feature having correlation are kept and the remaining features are discarded. For each feature, chi-square is calculated independently towards the target class and its significance is decided based on a predefined threshold (which is 0.05 commonly). The greater the value of chi-square, the lesser the significance of the feature. Similarly, the smaller the value of chi-square, the more the significance of the feature. Many researchers have proved to improve the results by using chi-square for feature reduction in text categorization [61], [65].

The formula of chi-square feature selection is shown in the equation 2, where c is the degree of freedom (threshold value), O is the observed value, E is the expected value, and X^2 is chi-sqaure computed value for feature.

$$X_c^2 = \Sigma \frac{(O_i - E_i)^2}{E_i} \tag{2}$$

E. INPUT AND CONVOLUTION LAYER

In the dataset, text Sequence 'a' contains 'w' entries. A d-dimensional dense vector is used to represent each entry 'w'. The feature map of input 'a' have $d \times w$ dimensions.

In the first step, we tokenize the headline and body texts using Keras tokenizer. After that, Keras embedding layer makes use of word2vec word embedding for transforming the tokens into word-vectors. For model one and two, the word vectors obtained from the word embedding layer are fed as input to the convolution layer. On the contrary, for model three and four, significant features are extracted from PCA and Chi-square first. Then these features are converted into word-vectors by embedding layer. Finally, these word vectors are passed to convolution layer.

The function of convolution layers is to capture a specific semantic or structural feature from the input matrix. Each word vector is passed to CNN neurons n. By applying filters with different sizes, we can get different kinds of features. Multiple filters f of different kernel sizes c is applied on each word embedding e and the output is generated as $(c \times e)$. In our work, the kernel size is 5, therefore, the filter of size 64 will create 5-word combinations. The input and output shape of CNN with several parameters is shown in Table 3 and Table 4.

TABLE 3. Layers structure of proposed model used in this work.

CNN-LSTM
Conv (5 × 5, @64), activation='relu'
Dropout(0.2) Max Pooling (4 × 4)
LSTM (100 neurons)
Dense (4 neurons)
Softmax (4-class)

TABLE 4. Model parameter structure.

Layer (type)	Output Shape	Param#
embedding_3 (Embedding)	(2240,100)	500000
dropout_3 (Dropout)	(2240, 100)	0
conv1d_3 (Conv1D)	(2236, 64)	32064
max_pooling1d_3 (MaxPooling1D)	(559, 64)	0
lstm_3 (LSTM)	(100)	66000
dense_3 (Dense)	(4)	404

F. ACTIVATION FUNCTION, MAX-POOLING AND DROPOUT

On the output of each CNN neuron, the ReLu activation function is applied. The purpose of using this activation layer is to convert any negative value to zero and to show non-linearity in the network. The function does not affect the output shape of the CNN layer, thus, it is the same as input shape.

The value of each neuron, after passing through the ReLu activation function is then fed to a 1-D max-pooling layer. This layer converts the input of each kernel size into one output by selecting the maximum value obtained in each kernel. This will greatly reduce the size of input features for the next layers and will avoid overfitting. The pool size p in our case is 4 thus, the output of this layer will reduce the features by kernel/pool size (p).

The dropout rate D for the whole network model is 0.2. The dropout layer is another way to reduce overfitting by dropping the input with values less than the dropout rate. In the FNC-1 dataset, the output of the dropout layer is the same as input passed to it because no value is lower than 0.2.

G. LSTM

The next layer is LSTM with units of 100. We have to generate a long chain-like sequence structure of our data and to keep the knowledge of previous inputs. For this purpose, LSTM is the most suitable choice as it consists of three gates named input gate i_k, output gate o_k and forget gate f_k. These gates decide which information is important for classification and which information is forgettable based on the dropout value. Previous input, which is necessary for prediction, is stored in cell memory block CK. Many variants are available for LSTM but the one we used in our model is as follows.

$$i_k = \sigma(W_i s_k + V_i h_{k-1} + b_i) \quad (3)$$

$$f_k = \sigma(W_f s_k + V_f h_{k-1} + b_f) \quad (4)$$

$$o_k = \sigma(W_o s_k + V_o h_{k-1} + b_o) \quad (5)$$

$$c_k = tanh(W_c x_k + V_c h_{k-1} + b_c) \quad (6)$$

where s is the input sequence $(s_1, s_2, s_3, \ldots, s_N)$ to s_{kth} vector representation. W and V are the weights associated with each matrix element. h is the hidden state related to time step $k-1$, where s_k is the input at that time and b is the bias vector. c is the cell memory block which gets updated each time at step $k-1$. In the output of LSTM layer, all the 100 units are connected to every unit in dense layer.

H. DENSE

The last layer of the proposed model is a fully connected dense layer, which produces a single output as a result. This layer is followed by a softmax activation function. Softmax activation is used for multi-class classification. We have used softmax activation in our dataset because it contains four classes (agree, disagree, discuss, and unrelated). We used Adam as the optimizer for testing purposes. The batch size used in testing is 32 and the number of epochs is set to 50.

V. PERFORMANCE EVALUATION METRICS

To compare and evaluate our model, we use accuracy (A), precision (P), recall (R), and F1-score (F) as evaluation metrics. Precision and Recall are computed using equations 7 and 8. Whereas, F1-score is the harmonic mean of precision and recall as expressed in equation 9.

$$P = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (7)$$

Precision is calculated as the ratio of correctly classified positive class and the sum of correctly and falsely classified values of the positive class. It tells us about the factualness of the model.

$$R = \frac{TruePositive}{TruePositive + FalseNegative} \quad (8)$$

A recall rate is calculated as the ratio of correctly classified positive class and the sum of correctly classified values of the positive class and falsely classified values of the negative class. It tells us about the completeness of the model.

$$F1 = 2 * \frac{\cdot precision \cdot recall}{precision + recall} \quad (9)$$

F1-score determines the accuracy of the model for each class. The F1-score metric is usually used when the dataset is imbalanced. As the dataset of FNC-1 is also highly imbalanced therefore, to calculate the class-wise accuracy, we use F1-score as evaluation metrics to show the completeness of the proposed model.

VI. RESULTS AND DISCUSSION

In the final set of experiments, the proposed ensemble model of CNN-LSTM is trained on 49, 972 samples and tested on 25, 413 headlines and articles. The training is performed using a 2 *GB* Dell PowerEdge *T*430 graphical processing unit on 2*x* Intel Xeon 8 Cores 2.4*Ghz* machine which is equipped with 32 GB DDR4 Random Access Memory (RAM). The training takes 3 hours to run epochs on 'Fake News Challenge Dataset' using pre-trained word embedding and to show the classification results. On the contrary, the feature reduction techniques take 1.8 hours for the computation.

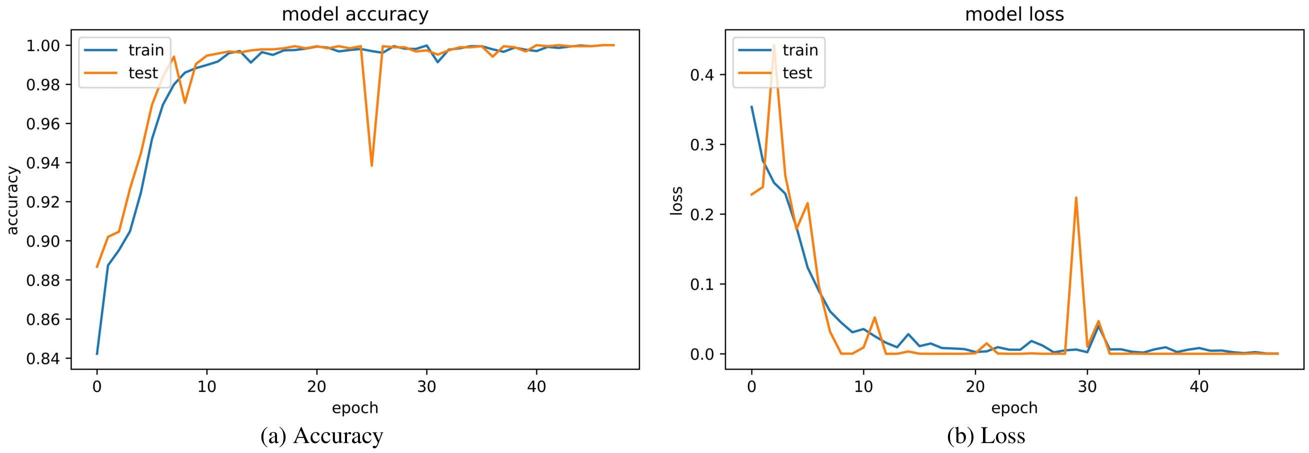
We have compared the outputs of the non-reduced feature set, PCA, and chi-square used by a CNN-LSTM architecture. By analyzing all the results, one can conclude that using PCA is more effective for severe dimensionality reduction as it significantly improved the accuracy. The presented model outperforms all other models by producing an accuracy of 97.8%. The average precision, recall, and F1-score for all classes are 97.4%, 98.2%, and 97.8% respectively as shown in Table 5. The detailed statistical results of our proposed model are shown in Table 5. The statistical significance ensures that one can easily classify any news as fake or legitimate using our proposed model. The train and test, accuracy and loss is shown in Figures 2a and 2b whereas the accuracy and F1-score comparison of our proposed model with state-of-the-art techniques are shown in Figure 3.

TABLE 5. **CNN-LSTM model classification results.**

Model	Accuracy	Precision	Recall	F-score
CNN-LSTM without pre-preprocessing	78.4%	81.4%	82.4%	81.9%
CNN-LSTM with pre-preprocessing	93%	96%	97%	96%
CNN-LSTM with Chi-Square	95.2%	92.3%	91.1%	91.49%
CNN-LSTM with PCA	97.8%	97.4%	98.2%	97.8%

As we know the LSTM can handle sequential data and If the amount of data is quite large it takes a lot of time to generate sequences and is likely to overfit. Whereas CNN cannot handle sequences of data, as it does not contain a memory unit. However, we can use principal component analysis (PCA) and chi-square to extract significant features to feed into the CNN-LSTM model.

It is evident from the results that when the features are used without any data cleaning or preprocessing, the accuracy is only 78% which is remarkably low. It indicates that the original dataset contains inconsistent, redundant, and noisy data in abundance. After performing the preprocessing steps and eliminating useless data, the accuracy goes up, dramatically, to be 93.0%. Besides, the application of chi-square further raises accuracy by selecting relevant features and making it 95%. Finally, We observed that the use of PCA with CNN and LSTM architecture resulted in the highest accuracy, 97.8%, with an insignificant decrease in categorization effectiveness. There was a sharp increase in precision, recall and F1-score as well. Results of k-fold to show sample variations are presented in Table 7. Moreover, there is a drastic decrease in the time required for performing a prediction when using dimensionality reduction methods. One of the many reasons



(a) Accuracy

(b) Loss

FIGURE 2. **Training and Testing Accuracy and Loss of Proposed Model.**

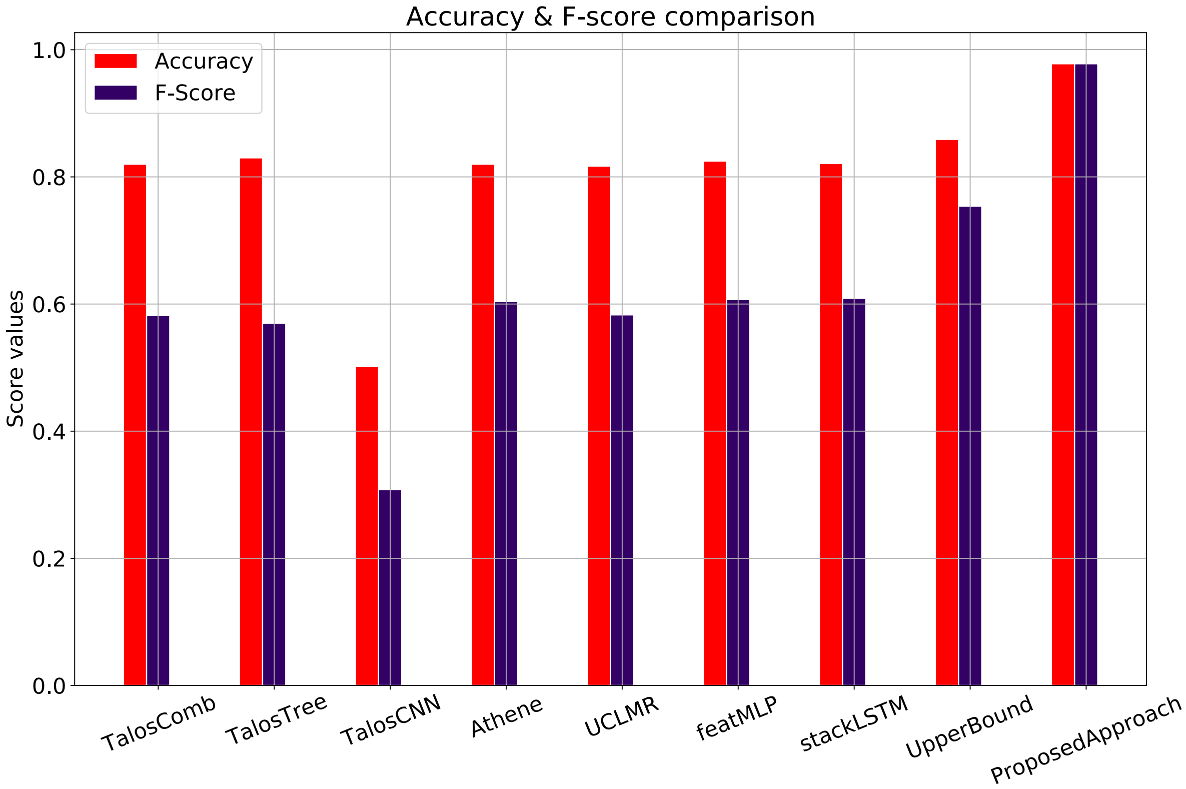


FIGURE 3. **Statistical and Accuracy Comparison of Different Methodologies.**

is that PCA is sensitive to low noise. It requires low capacity and memory. Moreover, it does not require large computation [32]. Thus, it has great advantages in terms of time and space complexity.

However, there is a limitation of using feature reduction techniques. The features in the dataset should be co-related enough to produce better results. Otherwise, these techniques would not have much effect on the final outcome. Furthermore, this work is limited to the training of claims and articles in English only. If this work is extended to other languages, cultural norms and differences in writing style might result in different performance.

Table 6 shows the comparison of the F1-score of all the different approaches experimented in [52] with our proposed model. It is evident from the results that the F-score of 'unrelated' is the highest among all the stances in every model. The reason is that the dataset is not balanced and many records of class 'unrelated' are far more than the other classes. Our model has the highest F1-score of discussing and agree with stances. Upper bound has slightly more F1-score

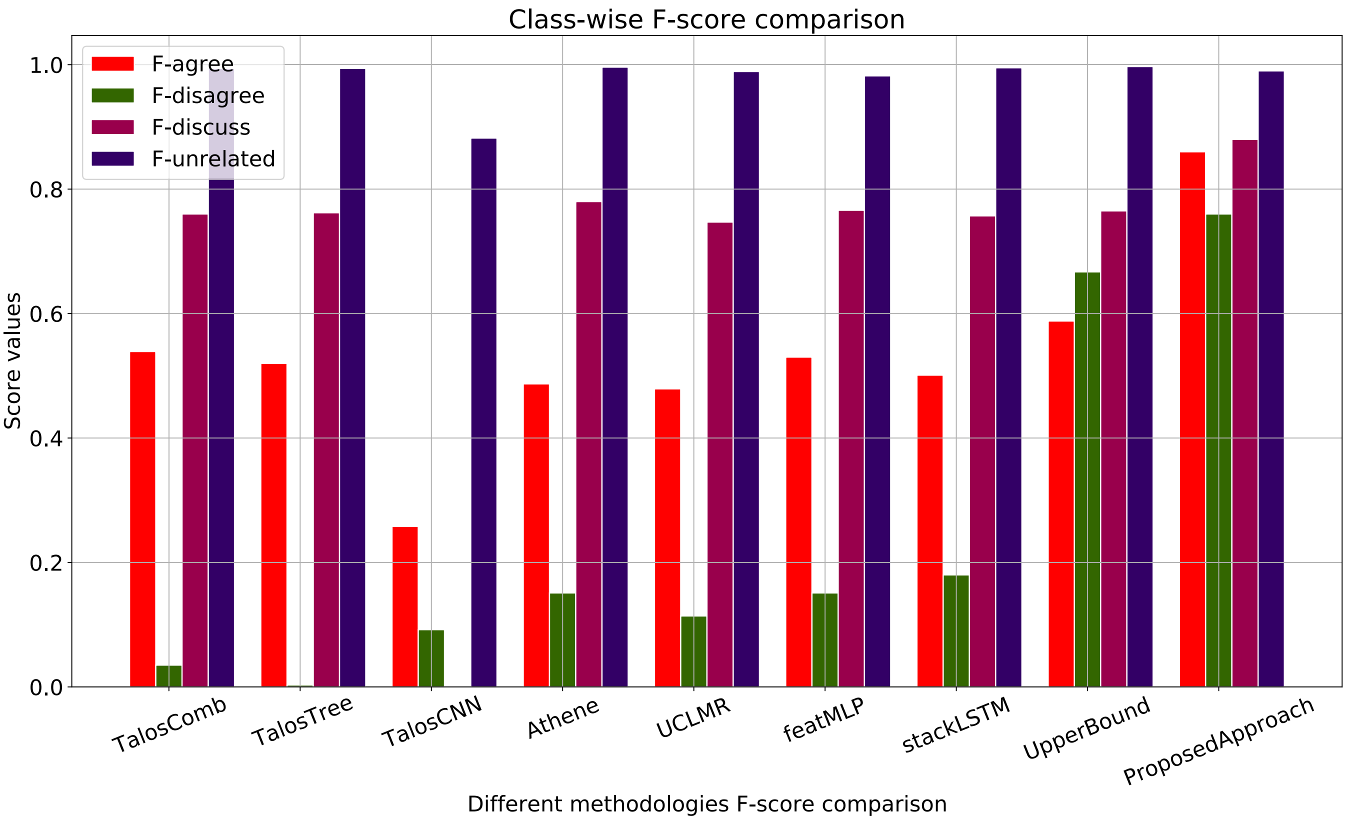


FIGURE 4. Class-wise F1 score comparison.

TABLE 6. Accuracy and F1-score comparison of different approaches.

Model	Accuracy	F-score	F-Agree	F-Disagree	F-Discuss	F-Unrelated
TalosComb	0.820	0.582	0.539	0.035	0.760	0.994
TalosTree	0.830	0.570	0.520	0.003	0.762	0.994
TalosCNN	0.502	0.308	0.258	0.092	0.0	0.882
Athene	0.820	0.604	0.487	0.151	0.780	0.996
UCLMR	0.817	0.583	0.479	0.114	0.747	0.989
featMLP	0.825	0.607	0.530	0.151	0.766	0.982
stackLSTM	0.821	0.609	0.501	0.180	0.757	0.995
Upperbound	0.859	0.754	0.588	0.667	0.765	0.997
Proposed Model						
CNN-LSTM with PCA	0.978	0.978	0.860	0.760	0.880	0.990

TABLE 7. CNN-LSTM model k-fold cross-validation with PCA.

K-folds	Accuracy	Precision	Recall	F-score
Fold-1	97.4%	97.4%	98.4%	97.9%
Fold-2	97.8%	96.2%	96.5%	96.3%
Fold-3	97.2%	96.3%	98.6%	97.4%
Fold-4	97.1%	96.4%	97.4%	96.9%
Fold-5	97.8%	97.9%	99.2%	98.5%
Fold-6	95.9%	96.3%	97.1%	96.7%
Fold-7	96.9%	96.8%	97.4%	97.1%
Fold-8	97.9%	97.5%	99.2%	98.3%
Fold-9	96.4%	96.9%	98.1%	97.5%
Fold-10	97.2%	97.3%	98.1%	97.7%
10-Fold Mean	**97.1%**	**96.9%**	**98.0%**	**97.4%**

of disagree stance than our model. The complete comparison of the class-wise f-score is shown in Figure 4.

A. COMPARISON WITH DEEP LEARNING MODELS

1) BERT

BERT stands for Bidirectional Encoder Representations from Transformers [67] and the results presented on the FNC-1 task have used the fine-tuning approach where all parameters are jointly fine-tuned and a simple classification layer is added to the pre-trained model. All masked positions are predicted by BERT independently. This means that it neglects dependencies between predicted masked positions during training. Due to the reduction of some dependencies BERT learns simultaneously, it suffers from a pre-train fine-tune inconsistency. This model obtains 91.3% accuracy on the FNC-1 task. F1-score achieved by BERT is far lesser than our model as well as the F1-score of agree, disagree, and unrelated classes.

2) XLNet

XLNet combines a bidirectional context as well as avoids independent predictions [68]. It introduces "permutation language modeling" in which it predicts tokens in some random order rather than predicting them in sequence. XLNet uses Transformer XL as its foundation architecture and outperforms BERT on 20 tasks. These tasks include document ranking, including natural language inference, question answering, and sentiment analysis. It improves upon BERT on the FNC-1 task and obtained 92.1% accuracy and 76.0% F1-score.

3) RoBERTa

An open-source language model named Roberta (Robustly Optimized BERT Approach) was released in July of 2019 [69]. In [67], the author constructs the large-scale language model using transfer learning on the Roberta-based deep transformer model, consisting of 12-layers of 768-hidden units, each with 12 attention heads, adding up to 125M parameters. To perform transfer learning, they train for fifty epochs and follow hyperparameter recommendations by [69] thus it outperforms both BERT and XLnet models. The model achieves 93.71% accuracy which is quite lesser than our model. The better accuracy can be achieved by using our proposed model with PCA and only one layer of CNN and the LSTM model. We modified only a limited number of parameters whereas in Roberta there are 125M parameters that require precise tuning. Moreover, with 12-layers of 768-hidden units in Roberta, the computational costs increase immensely. By comparing F1-scores, it is clear that the performance of Roberta on individual classes is lesser than our model. It will result in insufficient performance on even agree and disagree classes. The F1-score of discussing and unrelated are almost the same. The complete comparison of all these deep learning approaches are shown in Table 8.

TABLE 8. Accuracy and F1-score comparison with other deep learning approaches.

Model	Accuracy	F-score	F-Agree	F-Disagree	F-Discuss	F-Unrelated
BERT	0.913	0.728	0.647	0.478	0.800	0.986
XLNet	0.921	0.760	0.686	0.548	0.821	0.845
RoBERTa	0.937	0.781	0.707	0.580	0.845	0.991
Proposed Model	0.978	0.978	0.860	0.760	0.880	0.990

VII. CONCLUSION AND FUTURE WORK

This study proposed a fake news stance detection model, based on the headline and the body of the news irrespective of the previous studies which only considered the individual sentences or phrases. The proposed model incorporates principal component analysis (PCA) and chi-square with CNN and LSTM, in which PCA and chi-square extract the quality features which are passed to the CNN-LSTM model. First, we pass the non-reduced feature set with and without preprocessing to the neural network. Then the dimensionality reduction techniques are applied and the results are compared. PCA elevates the performance of the classifier for fake news detection as it removes the irrelevant, noisy, and redundant features from the feature vector. This process produces promising results by scoring up to 97.8% accuracy which is considerably better than the previous studies. It is pertinent to say that dimensionality reduction approaches can reduce the number of features while preserving the high performance of classifiers. Our future work entails: (a) validate the performance of our proposed model on larger datasets, (b) A tree-based learning may perform better than simple approaches, (c) different textual features and their fusion shall be analyzed to boost the performance.

CONFLICT OF INTEREST

Authors declare no conflict of interest exists.

REFERENCES

[1] T. Mihaylov, G. Georgiev, and P. Nakov, ‘‘Finding opinion manipulation trolls in news community forums,’’ in *Proc. 19th Conf. Comput. Natural Lang. Learn.*, Beijing, China, Jul. 2015, pp. 310–314. [Online]. Available: https://www.aclweb.org/anthology/K15-1032

[2] T. Mihaylov, I. Koychev, G. Georgiev, and P. Nakov, ‘‘Exposing paid opinion manipulation trolls,’’ in *Proc. Int. Conf. Recent Adv. Natural Lang. Process.*, Hissar, Bulgaria, Sep. 2015, pp. 443–450. [Online]. Available: https://www.aclweb.org/anthology/R15-1058

[3] T. Mihaylov and P. Nakov, ‘‘Hunting for troll comments in news community forums,’’ in *Proc. 54th Annu. Meeting Assoc. for Comput. Linguistics*, vol. 2, 2016, pp. 399–405, doi: 10.18653/v1/P16-2065.

[4] P. Bourgonje, J. Moreno Schneider, and G. Rehm, ‘‘From clickbait to fake news detection: An approach based on detecting the stance of headlines to articles,’’ in *Proc. EMNLP Workshop: Natural Lang. Process. meets Journalism*, 2017, pp. 84–89. [Online]. Available: https://www.aclweb.org/anthology/W17-4215

[5] S. Vosoughi, D. Roy, and S. Aral, ‘‘The spread of true and false news online,’’ *Science*, vol. 359, no. 6380, pp. 1146–1151, Mar. 2018.

[6] A. M. Michael Barthel and J. Holcomb. (2016). *Many Americans Believe Fake News is Sowing Confusion*. Accessed: Sep. 29, 2019. [Online]. Available: https://www.journalism.org/2016/12/15/many-americans-believe-fake-news-is-sowing-confusion/

[7] A. K. Chaudhry, ‘‘Stance detection for the fake news challenge: identifying textual relationships with deep neural nets,’’ in *Proc. Natural Lang. Process. Deep Learn.*, 2017, pp. 1–10.

[8] S. Chopra, ‘‘Towards automatic identification of fake news: Headline-article stance detection with LSTM attention models,’’ Stanford Univ., Stanford, CA, USA, Tech. Rep., 2017.

[9] G. Bhatt, A. Sharma, S. Sharma, A. Nagpal, B. Raman, and A. Mittal, ‘‘Combining neural, statistical and external features for fake news stance identification,’’ in *Proc. Companion Web Conf. Web Conf. (WWW)*, Geneva, Switzerland, 2018, p. 1353, doi: 10.1145/3184558.3191577.

[10] L. Konstantinovskiy, O. Price, M. Babakar, and A. Zubiaga, ‘‘Towards automated factchecking: Developing an annotation schema and benchmark for consistent automated claim detection,’’ 2018, *arXiv:1809.08193*. [Online]. Available: http://arxiv.org/abs/1809.08193

[11] D. M. J. Lazer, M. A. Baum, Y. Benkler, A. J. Berinsky, K. M. Greenhill, F. Menczer, M. J. Metzger, B. Nyhan, G. Pennycook, D. Rothschild, M. Schudson, S. A. Sloman, C. R. Sunstein, E. A. Thorson, D. J. Watts, and J. L. Zittrain, ‘‘The science of fake news,’’ *Science*, vol. 359, no. 6380, pp. 1094–1096, 2018.

[12] L. Borges, B. Martins, and P. Calado, ‘‘Combining similarity features and deep representation learning for stance detection in the context of checking fake news,’’ *J. Data Inf. Qual.*, vol. 11, no. 3, pp. 1–26, Jul. 2019, doi: 10.1145/3287763.

[13] M. A. Walker, P. Anand, R. Abbott, and R. Grant, ‘‘Stance classification using dialogic properties of persuasion,’’ in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, Stroudsburg, PA, USA, 2012, pp. 592–596. [Online]. Available: http://dl.acm.org/citation.cfm?id=2382029.2382124

[14] D. Sridhar, J. Foulds, B. Huang, L. Getoor, and M. Walker, ‘‘Joint models of disagreement and stance in online debate,’’ in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process.*, vol. 1. Beijing, China, Jul. 2015, pp. 116–125. [Online]. Available: https://www.aclweb.org/anthology/P15-1012

[15] S. Somasundaran and J. Wiebe, ‘‘Recognizing stances in ideological online debates,’’ in *Proc. NAACL HLT Workshop Comput. Approaches Anal. Gener. Emotion Text*, Los Angeles, CA, USA, Jun. 2010, pp. 116–124. [Online]. Available: https://www.aclweb.org/anthology/W10-0214

[16] M. Lukasik, P. K. Srijith, D. Vu, K. Bontcheva, A. Zubiaga, and T. Cohn, ‘‘Hawkes processes for continuous time sequence classification: an application to rumour stance classification in twitter,’’ in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, vol. 2. Berlin, Germany, Aug. 2016, pp. 393–398. [Online]. Available: https://www.aclweb.org/anthology/P16-2064

[17] L. Derczynski, K. Bontcheva, M. Liakata, R. Procter, G. Wong Sak Hoi, and A. Zubiaga, "SemEval-2017 Task 8: RumourEval: Determining rumour veracity and support for rumours," in *Proc. 11th Int. Workshop Semantic Eval. (SemEval)*, Vancouver, BC, Canada, Aug. 2017, pp. 69–76. [Online]. Available: https://www.aclweb.org/anthology/S17-2006

[18] C. Stab and I. Gurevych, "Parsing argumentation structures in persuasive essays," *Comput. Linguistics*, vol. 43, no. 3, pp. 619–659, Sep. 2017. [Online]. Available: https://www.aclweb.org/anthology/J17-3005

[19] D. P. Rao, (2017). *Exploring How Artificial Intelligence Technologies Could be Leveraged to Combat Fake News*. Accessed: Oct. 29, 2019. [Online]. Available: http://www.fakenewschallenge.org/

[20] J. Chung, Ç. Gülçehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *CoRR*, vol. abs/1412.3555, 2014. [Online]. Available: http://arxiv.org/abs/1412.3555

[21] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Netw.*, vol. 18, pp. 602–610, Jul./Aug. 2005.

[22] P. Neculoiu, M. Versteegh, and M. Rotaru, "Learning text similarity with siamese recurrent networks," in *Proc. 1st Workshop Represent. Learn.*, Jan. 2016, pp. 148–157.

[23] H. He, K. Gimpel, and J. Lin, "Multi-perspective sentence similarity modeling with convolutional neural networks," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1576–1586. [Online]. Available: https://www.aclweb.org/anthology/D15-1181

[24] R. Kiros, Y. Zhu, R. Salakhutdinov, R. S. Zemel, A. Torralba, R. Urtasun, and S. Fidler, "Skip-thought vectors," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 3294–3302.

[25] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process.*, vol. 1, 2015, pp. 1–11. [Online]. Available: http://dx.doi.org/10.3115/v1/p15-1150

[26] L. Yang, Q. Ai, D. Spina, R.-C. Chen, L. Pang, W. B. Croft, J. Guo, and F. Scholer, "Beyond factoid qa: Effective methods for non-factoid answer sentence retrieval," in *ECIR*, 2016.

[27] Y. Yang, W.-T. Yih, and C. Meek, "WikiQA: A challenge dataset for open-domain question answering," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Lisbon, Portugal, Sep. 2015, pp. 2013–2018. [Online]. Available: https://www.aclweb.org/anthology/D15-1237

[28] Z. Imtiaz, M. Umer, M. Ahmad, S. Ullah, G. S. Choi, and A. Mehmood, "Duplicate questions pair detection using siamese MaLSTM," *IEEE Access*, vol. 8, pp. 21932–21942, 2020.

[29] B. Ghanem, P. Rosso, and F. Rangel, "Stance detection in fake news a combined feature representation," in *Proc. 1st Workshop Fact Extraction Verification (FEVER)*, Brussels, Belgium, Nov. 2018, pp. 66–71. [Online]. Available: https://www.aclweb.org/anthology/W18-5510

[30] Q. Zeng, "Neural stance detectors for fake news challenge," Stanford Univ., Stanford, CA, USA, Tech. Rep., 2017.

[31] S. R. Pfohl, "Stance detection for the fake news challenge with attention and conditional encoding," Stanford Univ., Stanford, CA, USA, Tech. Rep., 2017.

[32] D. A., A. I., and S. S., "A comparative study on using principle component analysis with different text classifiers," *Int. J. Comput. Appl.*, vol. 180, no. 31, pp. 1–6, Apr. 2018, doi: 10.5120/ijca2018916800.

[33] S. Karamizadeh, S. Abdullah, A. Manaf, M. Zamani, and A. Hooman, "An overview of principal component analysis," *J. Signal Inf. Process.*, vol. 4, no. 3B, p. 173, Aug. 2013.

[34] M. Ahmad, A. M. Khan, J. A. Brown, S. Protasov, and A. M. Khattak, "Gait fingerprinting-based user identification on smartphones," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2016, pp. 3060–3067.

[35] S. Deegalla and H. Bostrom, "Reducing high-dimensional data by principal component analysis vs. Random projection for nearest neighbor classification," in *Proc. 5th Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2006, pp. 245–250.

[36] M. Ahmad, D. I. U. Haq, Q. Mushtaq, and M. Sohaib, "A new statistical approach for band clustering and band selection using K-means clustering," *IACSIT Int. J. Eng. Technol.*, vol. 3, no. 6, pp. 606–614, Dec. 2011.

[37] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, "SemEval-2016 task 6: Detecting stance in tweets," in *Proc. 10th Int. Workshop Semantic Eval. (SemEval-2016)*, San Diego, CA, USA, Jun. 2016, pp. 31–41. [Online]. Available: https://www.aclweb.org/anthology/S16-1003

[38] K. Popat, S. Mukherjee, J. Strötgen, and G. Weikum, "Where the truth lies: Explaining the credibility of emerging claims on the Web and social media," in *Proc. 26th Int. Conf. World Wide Web Companion*, Apr. 2017, pp. 1003–1012.

[39] C. Stab, T. Miller, and I. Gurevych, "Cross-topic argument mining from heterogeneous sources using attention-based neural networks," 2018, *arXiv:1802.05758*. [Online]. Available: http://arxiv.org/abs/1802.05758

[40] I. Augenstein, T. Rocktäschel, A. Vlachos, and K. Bontcheva, "Stance detection with bidirectional conditional encoding," in *Proc. 2016 Conf. Empirical Methods Natural Lang. Process.*, Austin, TX, USA, Nov. 2016, pp. 876–885. [Online]. Available: https://www.aclweb.org/anthology/D16-1084

[41] G. Zarrella and A. Marsh, "MITRE at SemEval-2016 task 6: Transfer learning for stance detection," 2016, *arXiv:1606.03784*. [Online]. Available: http://arxiv.org/abs/1606.03784

[42] O. Enayet and S. R. El-Beltagy, "NileTMRG at SemEval-2017 task 8: Determining rumour and veracity support for rumours on Twitter," in *Proc. 11th Int. Workshop Semantic Eval. (SemEval)*. Vancouver, BC, Canada, Aug. 2017, pp. 470–474. [Online]. Available: https://www.aclweb.org/anthology/S17-2082

[43] S. Dungs, A. Aker, N. Fuhr, and K. Bontcheva, "Can rumour stance alone predict veracity?" in *Proc. 27th Int. Conf. Comput. Linguistics*. Santa Fe, NM, USA, Aug. 2018, pp. 3360–3370. [Online]. Available: https://www.aclweb.org/anthology/C18-1284

[44] X. Wang, C. Yu, S. Baumgartner, and F. Korn, "Relevant document discovery for fact-checking articles," in *Proc. Companion Web Conf.*, 2018, pp. 525–533.

[45] W. Ferreira and A. Vlachos, "Emergent: a novel data-set for stance classification," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, San Diego, CA, USA, Jun. 2016, pp. 1163–1168. [Online]. Available: https://www.aclweb.org/anthology/N16-1138

[46] B. Sean, S. Doug, and P. Yuxi. (2017). *Talos Targets Disinformation With Fake News Challenge Victory*. Accessed: Oct. 29, 2019. [Online]. Available: http://blog.talosintelligence.com/2017/06/talos-fake-news-challenge.html

[47] A. Hanselowski, A. PVS, B. Schiller, and F. Caspelherr. (2017). *Team Athene on the Fake News Challenge*. Accessed: Oct. 29, 2019. [Online]. Available: https://medium.com/@andre134679/team-athene-on-the-fake-news-challenge-28a5cf5e017b

[48] B. Riedel, I. Augenstein, G. P. Spithourakis, and S. Riedel, "A simple but tough-to-beat baseline for the fake news challenge stance detection task," 2017, *arXiv:1707.03264*. [Online]. Available: http://arxiv.org/abs/1707.03264

[49] J. Thorne, M. Chen, G. Myrianthous, J. Pu, X. Wang, and A. Vlachos, "Fake news stance detection using stacked ensemble of classifiers," in *Proc. EMNLP Workshop: Natural Lang. Process. Meets Journalism* Copenhagen, Denmark, Sep. 2017, pp. 80–83. [Online]. Available: https://www.aclweb.org/anthology/W17-4214

[50] Q. Zhang, E. Yilmaz, and S. Liang, "Ranking-based method for news stance detection," in *Proc. Companion Web Conf.*, Geneva, Switzerland, 2018, p. 41–42, doi: 10.1145/3184558.3186919.

[51] J. Shang, T. Sun, J. Shen, X. Liu, A. Gruenheid, F. Korn, A. Lelkes, C. Yu, and J. Han, "Investigating rumor news using agreement-aware search," 2018, *arXiv:1802.07398*. [Online]. Available: http://arxiv.org/abs/1802.07398

[52] A. Hanselowski, A. PVS, B. Schiller, F. Caspelherr, D. Chaudhuri, C. M. Meyer, and I. Gurevych, "A retrospective analysis of the fake news challenge stance-detection task," in *Proc. 27th Int. Conf. Comput. Linguistics*, Santa Fe, NM, USA, Aug. 2018, pp. 1859–1874. [Online]. Available: https://www.aclweb.org/anthology/C18-1158

[53] M. Umer, S. Sadiq, M. Ahmad, S. Ullah, G. S. Choi, and A. Mehmood, "A novel stacked CNN for malarial parasite detection in thin blood smear images," *IEEE Access*, vol. 8, pp. 93782–93792, 2020.

[54] M. Mohtarami, R. Baly, J. Glass, P. Nakov, L. Marquez, and A. Moschitti, "Automatic stance detection using End-to-End memory networks," 2018, *arXiv:1804.07581*. [Online]. Available: http://arxiv.org/abs/1804.07581

[55] C. Dulhanty, J. L. Deglint, I. Ben Daya, and A. Wong, "Taking a stance on fake news: Towards automatic disinformation assessment via deep bidirectional transformer language models for stance detection," 2019, *arXiv:1911.11951*. [Online]. Available: https://arxiv.org/abs/1911.11951

[56] D. Pomerleau and Rao. (2017). *Fake News Challenge Dataset*. Accessed: Oct. 29, 2019. [Online]. Available: http://www.fakenewschallenge.org/

[57] M. Ahmad, "A fast 3D CNN for hyperspectral image classification," 2020, *arXiv:2004.14152*. [Online]. Available: http://arxiv.org/abs/2004.14152

[58] M. Ahmad, D. Ihsan, and D. Ulhaq, "Linear unmixing and target detection of hyperspectral imagery using OSP," in *Proc. Int. Conf. Modeling, Simulation Control*, Singapore, Jan. 2011.

[59] M. Ahmad, M. A. Alqarni, A. M. Khan, R. Hussain, M. Mazzara, and S. Distefano, "Segmented and non-segmented stacked denoising autoencoder for hyperspectral band reduction," *Optik*, vol. 180, pp. 370–378, Feb. 2019.

[60] M. Ahmad, A. M. Khan, and R. Hussain, "Graph-based spatial–spectral feature learning for hyperspectral image classification," *IET Image Process.*, vol. 11, no. 12, pp. 1310–1316, 2017.

[61] P. Meesad, P. Boonrawd, and V. Nuipian, "A chi-square-test for word importance differentiation in text classification," in *Proc. Int. Conf. Inf. Electron. Eng.*, 2011, pp. 110–114.

[62] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *Proc. ICML*, 1997, pp. 412–420.

[63] S. M. H. Dadgar, M. S. Araghi, and M. M. Farahani, "A novel text mining approach based on TF-IDF and support vector machine for news classification," in *Proc. IEEE Int. Conf. Eng. Technol. (ICETECH)*, Mar. 2016, pp. 112–116.

[64] D. Hou, J. Zhang, Z. Yang, S. Liu, P. Huang, and G. Zhang, "Distribution water quality anomaly detection from UV optical sensor monitoring data by integrating principal component analysis with chi-square distribution," *Opt. Express*, vol. 23, no. 13, p. 17487, Jun. 2015, doi: 10.1364/OE.23.017487.

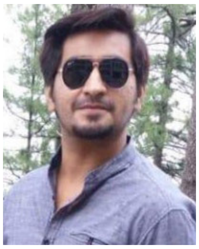
[65] Y. Zhai, W. Song, X. Liu, L. Liu, and X. Zhao, "A chi-square statistics based feature selection method in text classification," in *Proc. IEEE 9th Int. Conf. Softw. Eng. Service Sci. (ICSESS)*, Nov. 2018, pp. 160–163.

[66] X. Xia, D. Lo, W. Qiu, X. Wang, and B. Zhou, "Automated configuration bug report prediction using text mining," in *Proc. IEEE 38th Annu. Comput. Softw. Appl. Conf.*, Jul. 2014, pp. 107–116.

[67] V. Slovikovskaya, "Transfer learning from transformers to fake news challenge stance detection (FNC-1) task," 2019, *arXiv:1910.14353*. [Online]. Available: http://arxiv.org/abs/1910.14353

[68] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, "XLNet: Generalized autoregressive pretraining for language understanding," 2019, *arXiv:1906.08237*. [Online]. Available: http://arxiv.org/abs/1906.08237

[69] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," 2019, *arXiv:1907.11692*. [Online]. Available: http://arxiv.org/abs/1907.11692



MUHAMMAD UMER received the B.S. and M.S. degrees from the Department of Computer Science, Khwaja Fareed University of Engineering and IT (KFUEIT), Pakistan, 2018 and 2020, respectively, where he is currently pursuing the Ph.D. degree in computer science. He is also serving as a Research and Teaching Associate at KFUEIT. His recent research interests are related to data mining, mainly working machine learning and deep learning based IoT, text mining, and computer vision tasks. He is a Regular Reviewer for several top tier journals including but not limited to MTAP (Springer), PR Letter and IMAVIS (Elsevier), and so on.

ZAINAB IMTIAZ received the B.S. degree from the Department of Computer Science, University of Central Punjab (UCP), Pakistan, in 2015. She is currently pursuing the Master of Computer Science degree with the Khwaja Fareed University of Engineering and Information Technology (KFUEIT), Pakistan. She was a Microsoft Dynamic AX Developer from 2015 to 2017. She is also a Research Assistant with the Fareed Computing and Research Center, KFUEIT, and an Assistant Lecturer with the Computer Science Department. Her recent research interests include data mining, machine learning, and deep learning-based text mining.

SALEEM ULLAH was born in Ahmedpur East, Pakistan, in 1983. He received the B.Sc. degree in computer science from The Islamia University of Bahawalpur, Pakistan, in 2003, the M.I.T. degree in computer science from Bahauddin Zakariya University, Multan, in 2005, and the Ph.D. degree from Chongqing University, China, in 2012. From 2006 to 2009, he was a Network/IT Administrator with different companies. From August 2012 to February 2016, he was an Assistant Professor with The Islamia University of Bahawalpur. He has been an Associate Dean with the Khwaja Fareed University of Engineering and Information Technology, Rahim Yar Khan, since February 2016. He has almost 14 years of Industry experience in IT. His current research interests include adhoc networks, the IoTs, congestion control, data science, and network security.

ARIF MEHMOOD received the Ph.D. degree from the Department of Information and Communication Engineering, Yeungnam University, South Korea, in November 2017. Since November 2017, he has been a Faculty Member with the Department of Computer Science, KFUEIT, Pakistan. His recent research interests include data mining, mainly working on AI and deep learning-based text mining, and data science management technologies.

GYU SANG CHOI received the Ph.D. degree from the Department of Computer Science and Engineering, Pennsylvania State University, State College, PA, USA, in 2005. He was a Research Staff Member with Samsung Advanced Institute of Technology (SAIT), Samsung Electronics, from 2006 to 2009. Since 2009, he has been a Faculty Member with the Department of Information and Communication, Yeungnam University, South Korea. His research interests include non-volatile memory and storage systems.

BYUNG-WON ON received the Ph.D. degree from the Department of Computer Science and Engineering, Pennsylvania State University, State College, PA, USA, in 2007. He was a Full-Time Researcher with the Advanced Digital Sciences Center, The University of British Columbia, and the Advanced Institutes of Convergence Technology, for a period of seven years. Since 2014, he has been a Faculty Member with the Department of Software Convergence Engineering, Kunsan National University, South Korea. His recent research interests include data mining (probability theory and applications), machine learning, and artificial intelligence, mainly working on abstractive summarization, creative computing, and multiagent reinforcement learning.