Available online at www.sciencedirect.com

ScienceDirect

Cognitive Systems Research 61 (2020) 32–44

Cognitive Systems RESEARCH

www.elsevier.com/locate/cogsys

FNDNet – A deep convolutional neural network for fake news detection

Action editor: Erik Cuevas

Rohit Kumar Kaliyar [a], Anurag Goswami [a], Pratik Narang [b,*], Soumendu Sinha [c]

[a] *Department of Computer Science Engineering, Bennett University, Greater Noida, India*

[b] *Department of CSIS, BITS Pilani, Rajasthan, India*

[c] *Smart Sensors Area, CSIR - Central Electronics Engineering Research Institute, Pilani, Rajasthan, India*

Received 25 September 2019; received in revised form 26 November 2019; accepted 30 December 2019
Available online 21 January 2020

Abstract

With the increasing popularity of social media and web-based forums, the distribution of fake news has become a major threat to various sectors and agencies. This has abated trust in the media, leaving readers in a state of perplexity. There exists an enormous assemblage of research on the theme of Artificial Intelligence (AI) strategies for fake news detection. In the past, much of the focus has been given on classifying online reviews and freely accessible online social networking-based posts. In this work, we propose a deep convolutional neural network (FNDNet) for fake news detection. Instead of relying on hand-crafted features, our model (FNDNet) is designed to automatically learn the discriminatory features for fake news classification through multiple hidden layers built in the deep neural network. We create a deep Convolutional Neural Network (CNN) to extract several features at each layer. We compare the performance of the proposed approach with several baseline models. Benchmarked datasets were used to train and test the model, and the proposed model achieved state-of-the-art results with an accuracy of 98.36% on the test data. Various performance evaluation parameters such as Wilcoxon, false positive, true negative, precision, recall, F1, and accuracy, etc. were used to validate the results. These results demonstrate significant improvements in the area of fake news detection as compared to existing state-of-the-art results and affirm the potential of our approach for classifying fake news on social media. This research will assist researchers in broadening the understanding of the applicability of CNN-based deep models for fake news detection.

Keywords: Fake news; Social media; Machine learning; Deep learning; Neural network

1. Introduction

Over the past several years, fake news dissemination has become a major problem. Fake news is defined by the New York Times as ”made-up stories are written to deceive”, and they are published in formats similar to those used by traditional news agencies (Pan et al., 2018). Fake news has been identified for contributing to increased political polarization and partisan conflict in recent times. Recent examples included the controversy created during the 2016 presidential campaign for the United States (Pan et al., 2018) and Indian Airstrike in Balakot in 2019. Fake news is a text classification issue with a straight forward proposition (Shu, Sliva, Wang, Tang, & Liu, 2017).

* Corresponding author.
E-mail addresses: rk5370@bennett.edu.in (R.K. Kaliyar), anurag.goswami@bennett.edu.in (A. Goswami), pratik.narang@pilani.bits-pilani.ac.in (P. Narang), soumendu@ceeri.res.in (S. Sinha).
URL: http://www.rohitkaliyark.com (R.K. Kaliyar).

https://doi.org/10.1016/j.cogsys.2019.12.005
1389-0417/

Building a robust AI framework that can distinguish between "Genuine" news and "Fake" news is required to identify fake news (Pan et al., 2018; Shu et al., 2017). It is a challenging task for social media platforms such as Facebook®, Twitter®, etc. to identify the authentic content (Shu et al., 2017) within the large volume of data posted by the users. There exists a huge risk of posting and publishing such fake and non-authentic content over social media platforms. This research is a positive step towards addressing this critical issue. Fig. 1 shows a few examples of fake news which spread over social media platforms during the 2016 U.S. Presidential General Election (Shu et al., 2017). These fake news hampered the public emotionally and spread a negative impact (Persily, 2017) on the society during the 2016 U.S. Presidential General Election. The recent researches (Kumar & Shah, 2018; Pan et al., 2018; Shu et al., 2017) that have been done in the area of detecting fake news are based on both supervised as well as unsupervised methods (Dougherty, Kohavi, & Sahami, 1995).

Almost all unsupervised methods for learning word representations use the statistics of word occurrences in a corpus as the primary source of information (Kumar & Shah, 2018; Mikolov, Chen, Corrado, & Dean, 2013). Deducing relevant meaning from these statistics and representing them using the resultant word vectors is a problem to be addressed (Mikolov, Chen, et al., 2013). Another notable model that takes in vectors or words from their co-occurrence data is Global Vectors (GloVe) (Fu, Liu, Xu, & Cui, 2017). The GloVe is a count-based model for pre-training. The algorithms of word representations (Pennington, Socher, & Manning, 2014) can be divided into two main streams, i.e. statistics-based LDA (Linear discriminant analysis) (Zhang & Wallace, 2015) and learning-based (Word2Vec). LDA produces low dimensional word vectors by singular value decomposition (SVD) (Caliskan, Bryson, & Narayanan, 2017) on the co-occurrence matrix, while Word2Vec employs a three-layer neural network to do centre-context word pair classification task, where word vectors are the by-products. Word2vec is a predictive model, a feed-forward neural system that learns vectors to improve the predictive capacity (Fu et al., 2017). An attractive feature of Word2Vec algorithm is that the similar words are located together in the vector space, and arithmetic operations on word vectors can pose semantic or syntactic relationships (Caliskan et al., 2017; Zhang & Wallace, 2015). Besides being intuitive, the count-based model also provides an encoding function which can be used to infer unseen words or phrases.

Count-based models (Lebret & Collobert, 2015) can be dynamically adjusted in terms of its size and vocabulary. In count-based models, the semantic similarity between words is learned by counting the co-occurrence frequency (Pennington et al., 2014). Firstly, a large matrix of co-occurrence information is constructed (Tang et al., 2014), which contains the information about how frequently each "word", is seen in some "context" (the columns). The number of "contexts" needs to be large as it is combinatorial in size. Further, the matrix is factorized to yield a low-dimensional matrix of words and features, where each row yields a vector representation for each word. It is achieved by minimizing a "reconstruction loss" which looks for low-dimensional representations that can explain the variance in the high-dimensional data. In the case of GloVe, the word count matrix is preprocessed by normal-

Fig. 1. Examples of some Fake News spread over social media. (Source: Facebook®).

izing the counts and log-smoothing them. In contrast to Word2Vec, GloVe allows parallel implementation (Fu et al., 2017; Tang et al., 2014), which makes it easier to train the model over large datasets. GloVe algorithm combines the benefits of the word2vec skip-gram model in the word analogy tasks with matrix factorization methods exploiting global statistical information (Tang et al., 2014). It is a challenging task in NLP (Natural Language Processing) to develop deep architectures that can learn hierarchical representations (Stein, Jaques, & Valiati, 2019) of whole sentences in the form of fake content. Several existing research works (Ahmed, Traore, & Saad, 2017; Pan et al., 2018) have been described with the traditional machine learning algorithms (Ahmed et al., 2017) as well as convolutional neural networks (Vasudevan, Zoph, Shlens, & Le, 2019) to identify the fake news. Researchers (Pan et al., 2018; Stein et al., 2019) have adopted Convolutional Neural Network (CNN) in their research for the task of text classification and have shown quite successful and motivated results (Zhong, Xing, Love, Wang, & Luo, 2019). This work is motivated by the well-known fact (Zhong et al., 2019) that efficient results can be obtained using a deep problem-specific architecture (Founta et al., 2019) which develops hierarchical representations with the help of gradient descent (Wei et al., 2019).

In this paper, we have proposed a model (FNDNet) for fake news detection with a deeper convolutional approach. Our approach does not rely on extracting hand-crafted features. Instead, our model (FNDNet) is designed to learn the discriminatory features through the deep neural network automatically. We create a deep convolutional neural network (CNN), and a number of features are extracted at each layer. The design of our model's architecture is inspired by recent progress in the area of fake news detection (Fu et al., 2017; Pan et al., 2018; Ruchansky, Seo, & Liu, 2017; Zhang & Wallace, 2015). In contrast to existing approaches for fake content detection, our model has shown outstanding performance on large-scale real-world fake news datasets. Convolutional neural networks have accomplished excellent performance in numerous text classification tasks and also in different industrial applications (Liang, He, Xu, Chen, & Zeng, 2015; Roy, Basak, Ekbal, & Bhattacharyya, 2018; Zhou & Zafarani, 2018). Our proposed model would also be a helpful tool to provide a better solution for such industrial applications in the fields of business, retails and insurance, etc. Our proposed model also addresses the problem of selecting an optimal depth of CNNs for one particular text classification problem (Fake News Detection), and the results outperform the existing machine learning as well as deep learning-based implementations for classification. This model achieved better performance as compared to other classification models, with decreased classification error. We demonstrate the performance of our proposed model for language representation, where the accuracy of 98.36% has been achieved. Using our proposed approach, we obtain improved results in comparison to the baseline approaches, which makes FNDNet a promising model for accurate detection of fake news.

1.1. Motivation & research goal

Fake news detection is one of the emerging topics that has caught the attention of researchers across the world in the field of artificial intelligence. Despite receiving significant attention in the research community, fake news detection accuracy has not improve significantly due to insufficient context-specific news data. In contrast to the classical feature-based model, deep learning is advantageous because it doesn't require handcraft-based features, rather it recognizes the best feature set on it's own for a specific issue or problem for classification.

Research goal: Improve the accuracy of existing fake news detection using a Deep Convolutional Neural Network (FNDNet).

2. Related work

The popular fake news detection methods focus on news content and social context-based information (Egele, Stringhini, Kruegel, & Vigna, 2013; Fu et al., 2017; Kumar & Shah, 2018; Mikolov, Chen, et al., 2013; Pan et al., 2018). News content-based features are primarily extracted from textual aspects and also from visual aspects, for the classification of fake news. From textual features, explicit writing styles can be observed (Ghosh & Shah, 2018; Mikolov, Chen, et al., 2013) in addition to feelings or emotions (Liu & Wu, 2018; Wang et al., 2018) that ordinarily occur in fake news content. Various researchers have explored and investigated the detection approaches using content and context level information for fake news detection (Ghosh & Shah, 2018; Liu & Wu, 2018; Shu et al., 2017). Additionally, textual representations are modeled and mainly expressed utilizing tensor factorization method (Shu et al., 2017), deep neural systems (Mikolov, Chen, et al., 2013, 2018, 2017), which perform well to identify fake news. Visual features are extracted from visual components such as pictures and recordings to capture the various characteristics of fake news (Liang et al., 2015).

For social context-based methods, the features incorporate (i) user-based features, (ii) post-based features, and (iii) network-based features. User-based features are extracted from user-based profiles to gauge their attributes (Basak, Sural, Ganguly, & Ghosh, 2019; Potthast, Kiesel, Reinartz, Bevendorff, & Stein, 2017). Post-based features highlight user's social engagements regarding various stances (Pérez-Rosas, Kleinberg, Lefevre, & Mihalcea, 2017; Potthast et al., 2017) and the credibility of users (Yang et al., 2019). Network-based features are mainly extracted by building accurate detection systems, such as the diffusion networks (Karimi, Roy, Saba-Sadiya, & Tang, 2018; Potthast et al., 2017; Yang et al., 2019), association networks (Gupta, Thirukovalluru, Sinha, & Mannarswamy, 2018; Yang et al., 2018), and propagation networks (Gupta et al., 2018; Yang et al., 2019, 2018) for fake news classifications. With the widespread adoption of social media, research additionally accounts for social media activities for detecting fake news, for example, early detection of fake news by social learning (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013) and user-based relations (Araque, Corcuera-Platas, Sanchez-Rada, & Iglesias, 2017), semi-supervised detection (Ren, Wang, & Ji, 2016), unsupervised detection (Araque et al., 2017; Ren et al., 2016), and also through meta attributes (Araque et al., 2017; Giatsoglou et al., 2017; Ren et al., 2016). In this work, we study the existing detection methods for fake news using content and context-level information, mainly focusing on user comments and check-worthy content (Kim, 2014). In one of the research, authors (Vishwakarma, Varshney, & Yadav, 2019) have investigated their rule-based model for veracity analysis of information on various social media platforms and achieved an accuracy of 88.00% with real-world fake news dataset (FakeNewsNet).

In most of the context-related researches, authors have investigated the problem of Fake News using Kaggle fake news dataset. In one of the research, authors (Ahmed et al., 2017) have utilized different machine learning algorithms using TF-IDF (Term Frequency and Inverse Document Frequency) as the feature extraction for fake news detection. In one of the research, (Ahmed et al., 2017) have performed fake news classification using LR (Linear-regression-based uni-gram model) and achieved an accuracy of 89.00%. They have achieved an accuracy of 92% using LVSM (Linear Support Vector Machine as a classifier). In their investigation, (Yang et al., 2018) have used Convolutional Neural Networks for FakeNews Detection. They have used the concept of sensitivity analysis in their approach and achieved an accuracy of 92.10%. In their investigation, (O'Brien, Latessa, Evangelopoulos, & Boix, 2018) have used deep learning approaches for fake news classification. In their investigation, they have used a deep neural network (black-box approach) and achieved an accuracy of 93.50%. In their investigation, (Ghanem, Rosso, & Rangel, 2018) have used word embeddings and n-gram features to detect the stance in fake news. They have achieved an accuracy of 48.80% in their investigation. In their investigation, (Ruchansky et al., 2017) have used a hybrid model for fake news classification. They have taken user relationships as an important factor for fake news classification and achieved an accuracy of 89.20%. In their investigation, (Singh, Dasgupta, & Ghosh, 2017) have used different machine learning approaches for fake news detection using LIWC (Linguistic Analysis and Word Count-based approach). They have achieved an accuracy of 87.00% using SVM (support vector machine as a classifier). For better re-presentation, context-related research using kaggle fake news dataset is summarized in Table 1.

Table 1
Existing classification results using Kaggle fake news dataset.

Authors	Accuracy (%)
Ghanem et al.	48.80
Singh et al.	87.00
Ahmed et al. using LR-unigram model	89.00
Ruchansky et al.	89.20
Ahmed et al. using LSVM model	92.00
Yang et al.	92.10
Latessa et al.	93.50

3. Methodology

3.1. Pre-trained word embeddings

Pre-trained word embedding models are the simplest way to start working with word embedding techniques (Caliskan et al., 2017; Camacho-Collados, Pilehvar, & Navigli, 2016; Wang, Huang, & Zhao, 2016; Zhang & Wallace, 2015). The primary advantage of using these models is the capability of training with massive datasets. Conventionally, building such a massive dataset utilizing billions of different words is challenging, with a large corpus of language that captures word implications in a statistically robust manner. Embeddings generally represent geometrical encodings Zhang, Zhao, and LeCun (2015) of words based on how frequently they appear together in a text corpus. By utilizing a pre-trained model, time consumption can be reduced for training the model, cleaning, and processing, as applicable for huge datasets. Pre-prepared models can be categorized into two types of models, viz., context-free and contextual-based. Contextual-based models can further be divided as unidirectional or bidirectional (Cerisara, Kral, & Lenc, 2018; Iyyer, Manjunatha, Boyd-Graber, & Daumé, 2015; Kamkarhaghighi & Makrehchi, 2017) for pre-training. Context-free models, such as word2Vec or GloVe, create a solitary "word embedding" representation for each word in the vocabulary. Contextual-based models create a representation of each word that depends on different words in a sentence.

3.2. GloVe

GloVe is an unsupervised learning algorithm (Ahmed et al., 2017; Cerisara et al., 2018; Hailong, Wenyan, & Bo, 2014; Medhat, Hassan, & Korashy, 2014) that is utilized to discover the closeness of two words, with their separation in a vector space. These created vector representations are called word embedding vectors. In GloVe, training is performed on aggregated global word-word co-occurrence statistics (Ahmed et al., 2017) from a corpus. In the case of GloVe, the count's matrix is preprocessed by normalizing the counts and log-smoothing them. As discussed in Section I, GloVe allows parallel implementation, which makes it easier to train over more data. In this research work, we have used the smallest bundle of word embeddings is 822 Mb, called "glove.6B.zip". It was prepared on a dataset of one billion tokens (words) with a vocabulary of 400 thousand words. There are two or three different embedding vector sizes, including 50, 100, 200 and 300 dimensions. We picked the 100-dimensional version; along these lines, the Embedding layer must be characterized with output_dim set to 100. At last, we do not prefer to update the learned word weights in our model; hence we set the trainable attribute for the model to be False. The goal of GloVe is straightforward, i.e., to enforce the word vectors to capture sub-linear relationships in the vector space. GloVe gives lower weight for highly frequent word pairs to prevent the meaningless stop words such as "the", "an", etc., which do not dominate the training progress. Before training the actual model, a co-occurrence matrix X is constructed based on words, where a cell X_{ij} is a "strength", which represents how often the word i appears in the context of the word j. Once X is ready, it is necessary to decide our vector values in continuous space for each word in the dataset.

$$w_i^T \tilde{w}_j + b_i + \tilde{b}_j = logX_{ij} \tag{1}$$

where b_i and b_j are scalar bias terms associated with words i and j, respectively.

Our end goal is to minimize the target objective function J, which is helpful in recording all the squared errors, weighted with a function f:

$$J = \sum_{i,j=1}^{V} f(X_{ij})(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - logX_{ij})^2 \tag{2}$$

where V is defined as the size of the vocabulary.

3.3. Importance of pre-trained word embeddings

Recently, word embedding techniques have been broadly utilized in classification tasks using textual content. The increasing accuracy with pre-trained word embedding models greatly affect fake news classification for large datasets. For pre-trained embedding experiments, we replace the parameters of the processing layer with pre-trained embedding-based vectors, maintaining the index and freezing this layer, preventing it from being updated during the process of gradient descent. We illustrate our findings in terms of (i) training loss, (ii) confusion matrix, and (iii) accuracy. Our experiment demonstrates that word embedding-based vectors play an effective role in fake news detection.

3.4. FNDNet

From Fig. 2, we can observe the computational flow of our proposed model FNDNet. In existing researches (Cerisara et al., 2018; Zhong et al., 2019) fake news detection has been investigated by adopting a standard text classification model that consists of an Embedding layer as input in the form of word embedding vectors (Zhong et al., 2019), followed by a one-dimensional convolutional layer, followed by a max-pooling layer, and finally a prediction-based output layer (Zhong et al., 2019). The design of our model is motivated by the concept of multiple parallel channels-variable-size-based neural networks (Zhong et al., 2019). Our proposed model reaps the benefits of both traditional feature engineering and automated feature engineering (Seide, Li, Chen, & Yu, 2011). In our proposed model, inputs are the word-embedding vectors

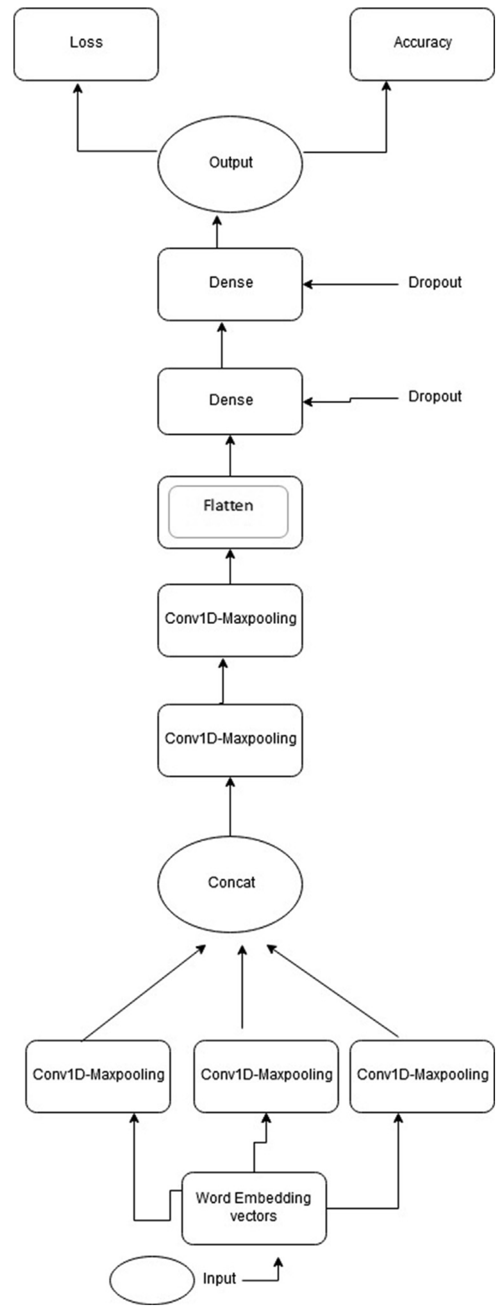

Fig. 2. Computational Graph for FNDNet Model.

generated from GloVe. We provide the same input word embedding vectors to all three parallel convolutional layers followed by a max-pooling layer. Next, we discuss the FNDNet elements, viz., the choices of number of convolutional layers, different kernels sizes in each convolutional layer, number of dense layers, dropout, and the selection of activation function to make our detection model more efficient and deep convolutional-based, as follows:

Convolutional layer: The convolutional layer is the core functional block of CNN-based networks for classification (Zhong et al., 2019). This layer consists of a set of filters or kernels (Zhong et al., 2019). With the help of these filters, a limited part of the input data is taken at a time for processing and applied across the full input. The main operations performed by this convolutional layer are matrix multiplications-based operations (Zhong et al., 2019), which passes through an activation function (Zhong et al., 2019) to produce the final output, which is usually a non-linear operation. These filters in the network act as a learner when they detect some specific type of feature at some spatial position in the given input. In our proposed model, we have used three parallel convolutional layers with different kernel sizes. In the convolutional layer, kernel size defines the number of words to consider as the convolution is passed across the input incomplete content-based document. The main motivation behind it is to contain more information in the form of a different set of word vectors during training. This might help unknown words across different embedding.

Max-pooling layer: Another key concept of convolutional neural networks is pooling, which is known as a form of non-linear down-sampling (Nagi et al., 2011; Zhong et al., 2019). Down-sampling is a combination of non-linear functions among which max pooling is the most common and effective function. To design our model, we are motivated by the fact that consecutive layers of the network that are activated by more complex features (Zhong et al., 2019) can play an important role in content-based classification (Nagi et al., 2011; Zhong et al., 2019). In a neural network, a pooling layer effectively down-samples the output of the prior layer, reducing the number of operations required for all the following layers present in the network. One of the main functions of this layer is that it passes the valid information from the previous layer to the next following layer. In our architecture, three max-pooling layers consolidate the output from the convolutional layers. All the outputs from the convolutional layers is concatenated and passed to the next level convolutional layer. We have two more convolutional layers, followed by max-pooling layers, to make our architecture more deep convolutional-based to achieve efficient results.

Flatten layer: We define the flatten layer as a function that converts the features taken from the pooling layer and map it to a single column, which is further passed to the fully connected layer. We have used one flatten layer in our designed model.

Dense layer or fully connected layer: We can understand the functionality of a dense layer as a linear operation (Zhong et al., 2019) in which every input is connected to every output by some weight. We have taken two dense layers to make our proposed model dense in nature. Researchers (Vasudevan et al., 2019; Zhong et al., 2019) have used one or two dense layers before the final softmax layer in their work. In our model, the first dense layer takes the output from the flatten layer and passed to the first dense layer and then the second dense layer predicts the final output.

Dropout: We can define the dropout as a regularization technique (Vasudevan et al., 2019; Zhong et al., 2019), which aims to reduce the complexity of any model with the end goal to prevent over-fitting (Zhong et al., 2019). We have applied dropout to fully-connected layers/dense layers. The application of dropout at each layer of the network has shown good results. We can understand the concept of dropout with an example that if we set the value of dropout is 40% of the total activation's in a given layer to zero, we would need to scale up the remaining ones by a factor of 2.5 for more accurate classification. Considering this aspect, we have taken the value of dropout is 0.2 throughout our experiments.

Activation Function: We have used ReLU (Rectified Linear Unit) as activation function (Li & Yuan, 2017). The main motivation behind to choose ReLU is that it is the elision of the rectified linear unit, which applies the non-saturating activation function. The main functionality of ReLU is that it successfully removes negative values from an activation map by setting them to zero in a network. ReLU also increases the nonlinear properties (Li & Yuan, 2017) of the decision-making function in the complete network without affecting the receptive fields (Li & Yuan, 2017) of the convolution layer. It is the most commonly used activation function in Deep Learning due to efficient results. It is 0 for all negative values of input z and equal to z for all positive values of input z. It is com-

Table 2
FNDNet layer Architecture Model.

Layer (type)	Input size	Output size
Embedding	1000	1000 × 100
Conv1d	1000 × 100	998 × 128
Conv1d	1000 × 100	997 × 128
Conv1d	1000 × 100	996 × 128
Maxpool	998 × 128	199 × 128
Maxpool	997 × 128	199 × 128
Maxpool	996 × 128	199 × 128
Concatenate	199 × 128, 199 × 128, 199 × 128	597 × 128
Conv1d	597 × 128	593 × 128
Maxpool	593 × 128	118 × 128
Conv1d	118 × 128	114 × 128
Maxpool	114 × 128	3 × 128
Flatten	3 × 128	384
Dense	384	128
Dense	128	2

putationally efficient than sigmoid or Tanh and solves the vanishing gradient problem. The equation of ReLU can be written as:

$$\sigma = max(0, z) \tag{3}$$

In Table 2, the layered architecture of our proposed FNDNet model is shown. In this architecture, the input is distributed into three parallel convolutional layers having 128 kernels. The first convolution layer has 128 kernels of size 3, which reduces input vector from 1000 to 998; the second convolution layer has 128 kernels of size 4, which reduces input vector from 1000 to 997; and the third convolution layer has 128 kernels of size 5 which reduces input vector from 1000 to 996 after convolution. After each convolution layer, a max-pooling layer is present to reduce the dimension. Following this, the max-pooling layer has filter size 5, which further reduces the vector to 1/5th of 996 i.e. 199. After concatenation, a convolution layer is applied of size 5 with 128 kernels followed by a max-pooling layer. This is followed by a dense hidden layer of 128 neurons. The output of the FNDNet is passed through a dense layer with dropout value 0.2.

3.4.1. Importance of deeper CNN

In the current era of computing, neural networks with 100 or more layers exist. Neural networks are trained using back-propagation and forward-propagation algorithms. In these algorithms, the gradient (derivative) of the cost function is used to update the weights of each layer. With each new layer, the value of gradient diminishes, especially when the sigmoid activation function is used. This results in prolonged training time. This problem is also known as the vanishing gradient. Direct connection in dense or deeper CNN solves this problem. Deeper CNN is also less prone to over-fitting as compared to normal CNN.

4. Comparative analysis with existing results

To evaluate the effectiveness of our proposed model (refer Table 3 for more details), a comparative analysis is outlined with the existing state-of-art methods for the detection of fake news using the Kaggle news dataset. The highest existing benchmark results were reported with an accuracy of 93.50% for fake news detection using the same dataset. Also, from Table 18, we can observe that our proposed model shows comparatively better results and effectiveness (training accuracy, testing accuracy, lightweight model for training). From Table 18, it can be observed that pre-trained word embedding models play a significant role in the fake news detection. Using the GloVe-enabled deep convolutional-based approach, we were able to achieve an accuracy of 98.36%. Our proposed model has achieved better results (refer Table 18 for more details) with real-world text-based fake news datasets as compared to existing works.

Table 3
Comparison-based classification results using Kaggle fake news dataset.

Authors	Testing accuracy (%)
Ghanem et al.	48.80
Singh et al.	87.00
Ahmed et al. using LR-unigram model	89.00
Ruchansky et al.	89.20
Ahmed et al. using LSVM model	92.00
Yang et al.	92.10
Latessa et al.	93.50
FNDNet	**98.36**

Table 4
Attributes & Number of Instances in the Fake News dataset.

Attribute	Number of instances in the dataset
ID	20800
title	20242
author	18843
text	20761
label	20800

5. Experimental results & analysis

5.1. Dataset description

Experiments were conducted using the fake news dataset.[1] It consists of two files (i) train.csv: A full training dataset (refer Table 4 for more details), and (ii) test.csv: A testing dataset with the following attributes as train.csv without the label (refer Table 4 for more details). This dataset is related to the fake news spread during the time of the 2016 U.S. Presidential Election. In this dataset, ID represents the unique value to the news article, the title represents the main heading related to particular news, the author represents the name of the creator of that news. Text, is the main core part of this dataset, which represents the complete news article, and labels provides the information about that the article as potentially unreliable or reliable.

5.1.1. Hyperparameter tuning

The process of selecting hyperparameters is a key aspect of any deep learning solution. Most deep learning algorithms explicitly define specific hyperparameters that control different aspects such as memory or cost of execution. Hyperparameters are the variables that are set before applying a learning algorithm to a context-specific dataset. The best numbers depend on each task and each context-dependent dataset. For selecting and optimizing hyperparameters, there are two basic approaches: manual and automatic selection. Both approaches are technically viable and the decision typically represents a trade-off between the deep understanding of the model required to select hyperparameters manually versus the high computational cost required by automatic selection algorithms.

[1] The dataset can be downloaded from: https://www.kaggle.com.

Table 5
Hyperparameters for Convolution Neural Network (CNN).

Hyperparameter	Description or value
No. of convolution layers	3
No. of max pooling layers	3
No. of dense layers	2
Loss function	Categorical-crossentropy
Activation function	Relu
Learning rate	0.001
Optimizer	Ada-delta
Number of epochs	5
Batch size	128

Table 6
Hyperparameters for LSTM.

Hyperparameter	Description or value
No. of convolution layers	2
No. of max pooling layers	2
No. of dense layers	4
Dropout rate	.2
Optimizer	Adam
Activation function	Relu
Loss function	Binary-crossentropy
Number of epochs	10
Batch size	64

Table 7
Hyperparameter for FNDNet.

Hyperparameter	Description or value
No. of convolution layers	5
No. of max pooling layers	5
No. of dense layers	4
Dropout rate	.2
Optimizer	Adadelta
Activation function	Relu
Loss function	Categorical-crossentropy
Number of epochs	5
Batch size	128

Tables 5–7 lists the values of hyperparameters used in our experiments for achieving accurate classification.

5.2. Performance parameters

In order to assess the performance of our proposed model, we have used precision, recall, F_1-Score, true negative rate, false-positive rate, accuracy as evaluation matrices. To control the different embedding types, we fixed the following hyper-parameters (refer Tables)5–7 and carried out the experiments with the same hyper-parameters.

5.2.1. Confusion matrix

The information about actual and predicted classifications performed by a classifier is represented by a confusion matrix. Performance evaluation of a classifier is commonly done using the data in the confusion matrix. A confusion matrix for the two-class problem is given in Table 8.

Table 8
Representation of Confusion Matrix.

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

5.2.2. Precision & recall

The measure of the ability of the model to accurately identify the occurrence of a positive class instance is determined by recall. It is defined as:

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

where precision is:

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

5.2.3. F_1-Score

F_1 score is the harmonic mean of Precision and Recall.

$$F_1 = \frac{2 * (Precision * Recall)}{(Precision + Recall)} \tag{6}$$

5.2.4. Specificity or True Negative Rate (TNR)

It measures the proportion of actual negatives that are correctly identified as such.

$$TrueNegativeRate(TNR) = \frac{TN}{FP + TN} \tag{7}$$

5.2.5. False Positive Rate (FPR)

The false positive rate is calculated as the ratio between the number of negative events wrongly categorized as positive (FP) and the total number of actual negative events (TN).

$$FalsePositiveRate(FPR) = \frac{FP}{FP + TN} \tag{8}$$

5.2.6. Accuracy

Accuracy is a measure of total correctly identified samples out of all the samples. It is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{9}$$

where True positive (TP) = correctly identified
False positive (FP) = incorrectly identified.
True negative (TN) = correctly rejected.
False negative (FN) = incorrectly rejected.

Table 9
Confusion Matrix for MNB.

	Predicted Positive	Predicted Negative
Actual Positive	898 (TP)	73 (FN)
Actual Negative	111 (FP)	853 (TN)

Table 10
Confusion Matrix for KNN.

	Predicted Positive	Predicted Negative
Actual Positive	836 (TP)	200 (FN)
Actual Negative	762 (FP)	282 (TN)

Table 11
Confusion Matrix for DT.

	Predicted Positive	Predicted Negative
Actual Positive	901 (TP)	135 (FN)
Actual Negative	413 (FP)	631 (TN)

Table 12
Confusion Matrix for RF.

	Predicted Positive	Predicted Negative
Actual Positive	802 (TP)	234 (FN)
Actual Negative	361 (FP)	683 (TN)

Table 13
Confusion Matrix for CNN with GloVe.

	Predicted Positive	Predicted Negative
Actual Positive	882 (TP)	76 (FN)
Actual Negative	90 (FP)	952 (TN)

Table 14
Confusion Matrix for LSTM with GloVe.

	Predicted Positive	Predicted Negative
Actual Positive	995 (TP)	47 (FN)
Actual Negative	8 (FP)	1030 (TN)

Table 15
Confusion Matrix for FNDNet with GloVe.

	Predicted Positive	Predicted Negative
Actual Positive	995 (TP)	32 (FN)
Actual Negative	6 (FP)	1018 (TN)

5.2.7. *Results*

Firstly, various experiments have been conducted for evaluating the performance of different machine learning classifiers using fake news dataset (refer Table 4 for more details). Respective confusion matrices (Tables 9–15) are also included for each machine learning as well as the deep learning-based classifier for evaluating the performance using performance evaluating parameters (for more details refer to the Section 5.2). In our investigation, we have found that using Multinational Naive Bayes as a classifier, we have achieved an accuracy of 89.97%. Machine Learning-based classification results are shown in Fig. 3. In our investigation, parameters for evaluating the performance of any classifier (Precision, Recall, F_1-Score,True Negative Rate, and False Positive Rate) are also included to validate our classification results. It has been found that among all machine learning-based models, MNB consists of higher true negative rate and less false positive rate. We have validated our results with others performance evaluation parameters such as precision, recall, and F1-Score with MNB as a classifier. During our investigation with machine learning-based models, we found that performance decreases as the scale of data increases. Accuracy is also low in the case of machine learning-based classifiers due to limited handwritten features present in the respective dataset. Facing these issues, we were motivated towards deep-learning-based implementations. Deep learning automatically finds out the features which are important for classification, whereas in Machine Learning, we had to manually provide/feed the features. In order to implement deep learning-based models, firstly, we implemented the perceptron-based neural networks, with other pre-trained word embedding models such as Word2Vec, BoW, etc. We have achieved better accuracy (94.31%) as compared to previous machine learning-based models with GloVe. In order to test the performance of our deep neural network model, various experiments have been conducted with GloVe. The end goal of this research work is to design a more accurate system for fake news classification.

Considering the issues in machine-learning based implementations, we implemented deep learning-based models (CNN), LSTM (Long Short Term Memory), and our deep neural network (FNDNet)) and recorded their performances for fake news classification. It was found that using the GloVe-enabled pre-trained word embedding technique with CNN, we obtained a training accuracy of 64.50%, with a validation accuracy of 91.50%, with 5 epochs. We implemented LSTM using the same pre-trained technique (GloVe) and recorded a training accuracy of 99.74% and testing accuracy of 97.25% with 10 epochs. We then implemented our proposed model (FNDNet) using the GloVe pre-trained word embedding model and recorded a testing accuracy of 98.36%. It has been shown that the best accuracy is achieved via our proposed model. Tables 16 and 17, summarizes the values of different performance parameters (Precision, Recall, and F_1-Score in Table 16 and (true negative rate and false-positive rate in Table 17) for machine learning as well as deep learning-based classification models. These results validate the performance of our proposed model (FNDNet) with other classification models. From Fig. 4, the accuracy and model-loss or cross-entropy loss of our implemented deep CNN-based model with training and testing data values can be seen. It is observed that with the increasing number of epochs, testing-accuracy increases, and model-loss reduces significantly with our proposed model. Cross-entropy loss measures the performance of a classification model whose output is a value between 0 and 1. Cross-entropy loss increases as the predicted probability diverges from the actual label.

From the above analysis as seen in Fig. 4, the training loss decays rapidly with the GloVe-enabled deep convolutional based model in comparison to the normal embedding-layer-based model. Table 2, the training loss

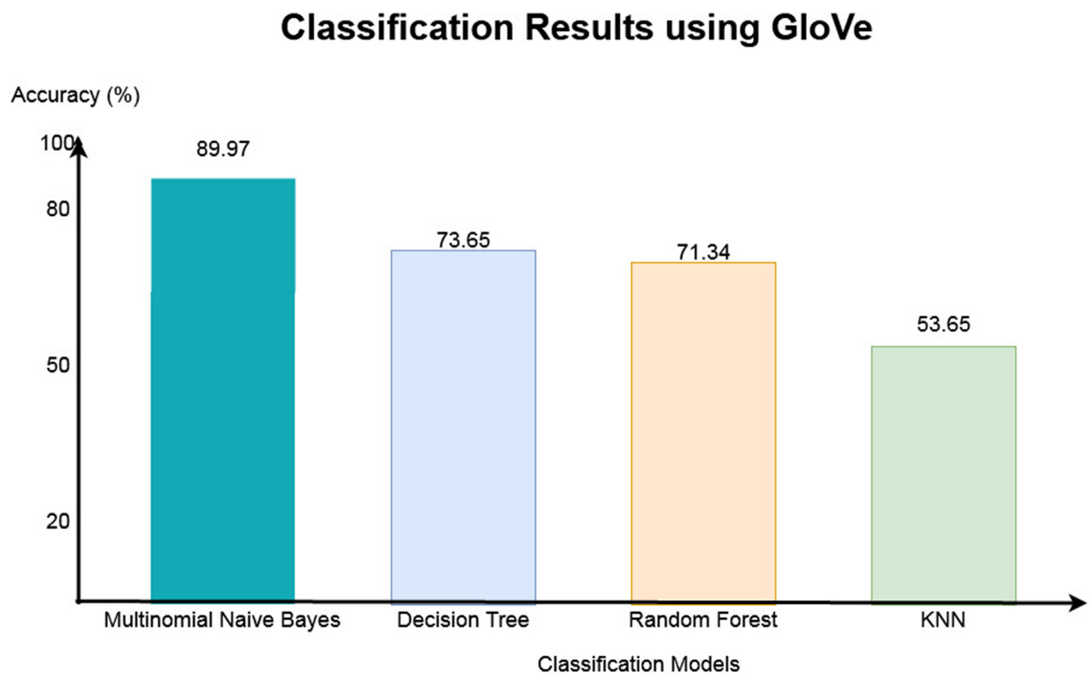

Fig. 3. Machine learning-based Classification results using GloVe.

Table 16
Classification results using Machine Learning and Deep Learning-based models.

Word Embedding Model	Classification Model	Precision (%)	Recall (%)	F_1-Score (%)
Tf-Idf on unigrams and bigrams	Neural Network	95.31	92.78	94.03
BoW without unigram and bigrams	Neural Network	91.45	87.67	89.57
Word2Vec	Neural Network	80.23	72.34	76.08
GloVe	Mutinomial Naive Bayes	88.99	92.48	90.70
GloVe	Decision Tree	68.56	86.97	73.68
GloVe	Random Forest	68.95	77.41	72.94
GloVe	KNN	52.31	80.69	63.37
GloVe	CNN	90.74	92.07	91.40
GloVe	LSTM	99.20	95.49	97.31
GloVe	**Our Proposed model (FNDNet)**	**99.40**	**96.88**	**98.12**

Table 17
True Negative Rate (TNR) and False positive rate (FPR) using Machine Learning and Deep Learning-based models.

Word Embedding Model	Classification Model	TNR (%)	FPR (%)
Tf-Idf on unigrams and bigrams	Neural Network	85.12	14.52
BoW without unigram and bigrams	Neural Network	62.17	12.23
Word2Vec	Neural Network	59.35	37.65
GloVe	Mutinomial Naive Bayes	88.49	11.51
GloVe	Decision Tree	60.44	39.56
GloVe	Random Forest	65.42	34.58
GloVe	KNN	27.01	72.98
GloVe	CNN	91.36	8.64
GloVe	LSTM	99.22	0.77
GloVe	**Our Proposed model (FNDNet)**	**99.41**	**0.59**

for pre-trained embedding-based models decays relatively fast and without any fluctuations. Fig. 4 shows that cross-entropy loss reduces significantly using the FNDNet model and it achieved the highest accuracy in comparison to traditional machine learning-based models as well as other deep learning-based models, with minimal losses.

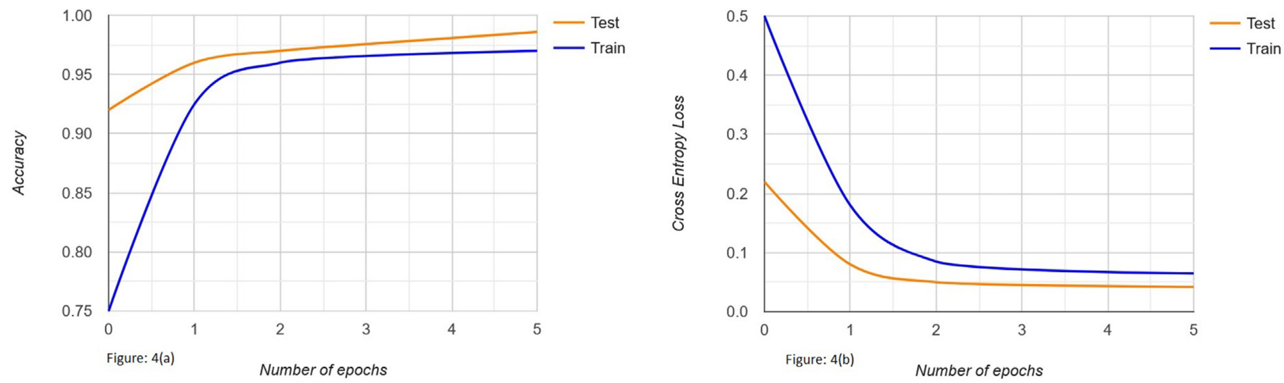

Fig. 4. Accuracy & Model loss or Cross-entropy loss of FNDNet model with Training and Testing samples.

Table 18
Classification results using Machine Learning and Deep Learning-based models.

Word Embedding Model	Classification Model	Accuracy (%)
Tf-Idf on unigrams and bigrams	Neural Network	94.31
BoW without unigram and bigrams	Neural Network	89.23
Word2Vec	Neural Network	75.67
GloVe	Mutinomial Naive Bayes	89.97
GloVe	Decision Tree	73.65
GloVe	Random Forest	71.34
GloVe	KNN	53.75
GloVe	CNN	91.50
GloVe	LSTM	97.25
GloVe	**Our Proposed model (FNDNet)**	**98.36**

6. Conclusion

In this research, we have proposed FNDNet, a deep convolutional neural network for fake news detection. FNDNet was analyzed using GloVe as a pre-trained word embedding. We have used pre-trained word embedding, which is unidirectional in training. Various machine learning algorithms, as well as deep learning algorithms, have been employed for classifications. Results demonstrate that our proposed model (FNDNet) provides the state-of-art results for predicting fake news with an accuracy of 98.36%. To validated our classification results, different performance evaluation parameters (such as false positive rate, true negative rate, precision, recall, etc.) have been included. Using our proposed model (FNDNet), there are very fewer chances for inaccurate classification having very less false-positive rate (0.59%) and high true negative rate (99.41%). The cross-entropy rate is also less with our proposed model. Results strongly encourage us to use our proposed model in the area of fake news classification. In the future, we plan to include a bi-directional transformer encoder (BERT) for pre-training the model followed by deep-learning architectures for fake news classification. Our further plan is also to use a multi-model based approaches with recent pre-trained word embeddings (such as Elmo and XLNet, etc.).

Despite the high performance of our classifier, there is a scope for improvement. We evaluated our models using binary class datasets to predict the content of any news article as real or fake. Our further plan is to use a hybrid approach (detection based on content, context, and temporal level information as a combination) for fake news classification. This type of hybrid approach can create more impact in the case of multi-label datasets. These issues limit our analysis and thus prevent broader generalization. Our further focus would be to look at fake news detection from the perspective of echo-chambers, which is defined as a group of people having the same understanding for any social issue or aspect, for example, political echo-chamber. The primary motivation to include echo-chambers is that any user is not isolated to any social media platform but exists in the form of a community. For future aspects, a deep focus on news-post relationships is a key area to explore, followed by the inclusion of different echo-chambers for better classification of news articles. In future, user profiles-based features can also be included for better prediction of news articles. An approach with the multiple parallel channel-based deep convolutional neural networks of different kernel sizes can also be helpful to perform the classification news articles. These networks would be helpful to read the different groups of words for the better classification of texts. Due to limited research is available for visual information such as video and images, a video-image-based analysis can be an exploration area to build a better detection system for video forensic investigation. The creation of context-specific datasets (video and images-based) can be a milestone for better research prospects. We can explore the problem of Fake news detection with knowledge-based and fact-based approaches with other automated approaches. A multi-model approach (a combination of different learning techniques) is the main necessity for fake news detection for solving the multi-class fake news detection problem.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Ahmed, H., Traore, I., & Saad, S. (2017). Detection of online fake news using n-gram analysis and machine learning techniques. In *International conference on intelligent, secure, and dependable systems in distributed and cloud environments* (pp. 127–138). Springer.

Araque, O., Corcuera-Platas, I., Sanchez-Rada, J. F., & Iglesias, C. A. (2017). Enhancing deep learning sentiment analysis with ensemble techniques in social applications. *Expert Systems with Applications, 77*, 236–246.

Basak, R., Sural, S., Ganguly, N., & Ghosh, S. K. (2019). Online public shaming on twitter: Detection, analysis, and mitigation. *IEEE Transactions on Computational Social Systems, 6*, 208–220.

Caliskan, A., Bryson, J. J., & Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. *Science, 356*, 183–186.

Camacho-Collados, J., Pilehvar, M. T., & Navigli, R. (2016). Nasari: Integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities. *Artificial Intelligence, 240*, 36–64.

Cerisara, C., Kral, P., & Lenc, L. (2018). On the effects of using word2vec representations in neural networks for dialogue act recognition. *Computer Speech & Language, 47*, 175–193.

Dougherty, J., Kohavi, R., & Sahami, M. (1995). Supervised and unsupervised discretization of continuous features. In *Machine learning proceedings 1995* (pp. 194–202). Elsevier.

Egele, M., Stringhini, G., Kruegel, C., & Vigna, G. (2013). Compa: Detecting compromised accounts on social networks. In *NDSS*. .

Founta, A. M., Chatzakou, D., Kourtellis, N., Blackburn, J., Vakali, A., & Leontiadis, I. (2019). A unified deep learning architecture for abuse detection. In *Proceedings of the 10th ACM conference on web science* (pp. 105–114). ACM.

Fu, X., Liu, W., Xu, Y., & Cui, L. (2017). Combine hownet lexicon to train phrase recursive autoencoder for sentence-level sentiment analysis. *Neurocomputing, 241*, 18–27.

Ghanem, B., Rosso, P., & Rangel, F. (2018). Stance detection in fake news a combined feature representation. In *Proceedings of the first workshop on fact extraction and verification (FEVER)* (pp. 66–71).

Ghosh, S., & Shah, C. (2018). Towards automatic fake news classification. *Proceedings of the Association for Information Science and Technology, 55*, 805–807.

Giatsoglou, M., Vozalis, M. G., Diamantaras, K., Vakali, A., Sarigiannidis, G., & Chatzisavvas, K. C. (2017). Sentiment analysis leveraging emotions and word embeddings. *Expert Systems with Applications, 69*, 214–224.

Gupta, S., Thirukovalluru, R., Sinha, M., & Mannarswamy, S. (2018). Cimtdetect: A community infused matrix-tensor coupled factorization based method for fake news detection. In *2018 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM)* (pp. 278–281). IEEE.

Hailong, Z., Wenyan, G., & Bo, J. (2014). Machine learning and lexicon based methods for sentiment classification: a survey. In *2014 11th web information system and application conference* (pp. 262–265). IEEE.

Iyyer, M., Manjunatha, V., Boyd-Graber, J., & Daumé, H. III, (2015). Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (Volume 1: Long Papers)* (pp. 1681–1691).

Kamkarhaghighi, M., & Makrehchi, M. (2017). Content tree word embedding for document representation. *Expert Systems with Applications, 90*, 241–249.

Karimi, H., Roy, P., Saba-Sadiya, S., & Tang, J. (2018). Multi-source multi-class fake news detection. In *Proceedings of the 27th international conference on computational linguistics* (pp. 1546–1557).

Kim, Y. (2014). Convolutional neural networks for sentence classification. arXiv preprint arXiv: 1408.5882.

Kumar, S., & Shah, N. (2018). False information on web and social media: A survey. arXiv preprint arXiv: 1804.08559.

Lebret, R., & Collobert, R. (2015). Rehabilitation of count-based models for word vector representations. In *International conference on intelligent text processing and computational linguistics* (pp. 417–429). Springer.

Liang, G., He, W., Xu, C., Chen, L., & Zeng, J. (2015). Rumor identification in microblogging systems based on users' behavior. *IEEE Transactions on Computational Social Systems, 2*, 99–108.

Liu, Y., & Wu, Y. F. B. (2018). Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Thirty-second AAAI conference on artificial intelligence*. .

Li, Y., & Yuan, Y. (2017). Convergence analysis of two-layer neural networks with relu activation. In *Advances in neural information processing systems* (pp. 597–607).

Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal, 5*, 1093–1113.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv: 1301.3781.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111–3119).

Nagi, J., Ducatelle, F., Di Caro, G. A., Cireşan, D., Meier, U., Giusti, A., Nagi, F., Schmidhuber, J., & Gambardella, L. M. (2011). Max-pooling convolutional neural networks for vision-based hand gesture recognition. In *2011 IEEE international conference on signal and image processing applications (ICSIPA)* (pp. 342–347). IEEE.

O'Brien, N., Latessa, S., Evangelopoulos, G., & Boix, X. (2018). The language of fake news: Opening the black-box of deep learning based detectors.

Pan, J. Z., Pavlova, S., Li, C., Li, N., Li, Y., & Liu, J. (2018). Content based fake news detection using knowledge graphs. In *International semantic web conference* (pp. 669–683). Springer.

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532–1543).

Pérez-Rosas, V., Kleinberg, B., Lefevre, A., & Mihalcea, R. (2017). Automatic detection of fake news. arXiv preprint arXiv: 1708.07104.

Persily, N. (2017). The 2016 us election: Can democracy survive the internet? *Journal of Democracy, 28*, 63–76.

Potthast, M., Kiesel, J., Reinartz, K., Bevendorff, J., & Stein, B. (2017). A stylometric inquiry into hyperpartisan and fake news. arXiv preprint arXiv: 1702.05638.
Ren, Y., Wang, R., & Ji, D. (2016). A topic-enhanced word embedding for twitter sentiment classification. *Information Sciences, 369*, 188–198.
Roy, A., Basak, K., Ekbal, A., & Bhattacharyya, P. (2018). A deep ensemble framework for fake news detection and classification. arXiv preprint arXiv: 1811.04670.
Ruchansky, N., Seo, S., & Liu, Y. (2017). Csi: A hybrid deep model for fake news detection. In *Proceedings of the 2017 ACM on conference on information and knowledge management* (pp. 797–806). ACM.
Seide, F., Li, G., Chen, X., & Yu, D. (2011). Feature engineering in context-dependent deep neural networks for conversational speech transcription. In *2011 IEEE workshop on automatic speech recognition & understanding* (pp. 24–29). IEEE.
Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter, 19*, 22–36.
Singh, D. V., Dasgupta, R., & Ghosh, I. (2017). Automated fake news detection using linguistic analysis and machine learning. In *International conference on social computing, behavioral-cultural modeling, & prediction and behavior representation in modeling and simulation (SBP-BRiMS)* (pp. 1–3).
Stein, R. A., Jaques, P. A., & Valiati, J. F. (2019). An analysis of hierarchical text classification using word embeddings. *Information Sciences, 471*, 216–232.
Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., & Qin, B. (2014). Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of the 52nd annual meeting of the association for computational linguistics (Volume 1: Long Papers)* (pp. 1555–1565).
Vasudevan, V., Zoph, B., Shlens, J., & Le, Q. V. (2019). Neural architecture search for convolutional neural networks. US Patent App. 16/040,067.
Vishwakarma, D. K., Varshney, D., & Yadav, A. (2019). Detection and veracity analysis of fake news via scrapping and authenticating the web search. *Cognitive Systems Research, 58*, 217–229.
Wang, Y., Huang, M., Zhao, L., et al. (2016). Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing* (pp. 606–615).
Wang, Y., Ma, F., Jin, Z., Yuan, Y., Xun, G., Jha, K., ... Gao, J. (2018). Eann: Event adversarial neural networks for multi-modal fake news detection. In *Proceedings of the 24th acm sigkdd international conference on knowledge discovery & data mining* (pp. 849–857). ACM.
Wei, W., Zhou, B., Maskeliūnas, R., Damaševičius, R., Połap, D., & Woźniak, M. (2019). Iterative design and implementation of rapid gradient descent method. In *International conference on artificial intelligence and soft computing* (pp. 530–539). Springer.
Yang, Y., Zheng, L., Zhang, J., Cui, Q., Li, Z., & Yu, P. S. (2018). Ti-cnn: Convolutional neural networks for fake news detection. arXiv preprint arXiv: 1806.00749.
Yang, S., Shu, K., Wang, S., Gu, R., Wu, F., & Liu, H. (2019). Unsupervised fake news detection on social media: A generative approach. In *Proceedings of 33rd AAAI conference on artificial intelligence*. .
Zhang, Y., & Wallace, B. (2015). A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. arXiv preprint arXiv: 1510.03820.
Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Advances in neural information processing systems* (pp. 649–657).
Zhong, B., Xing, X., Love, P., Wang, X., & Luo, H. (2019). Convolutional neural network: Deep learning-based classification of building quality problems. *Advanced Engineering Informatics, 40*, 46–57.
Zhou, X., & Zafarani, R. (2018). Fake news: A survey of research, detection methods, and opportunities. arXiv preprint arXiv: 1812.00315.

Rohit Kumar Kaliyar is a Research Scholar in the Department of Computer Science Engineering, Bennett University, India. Formerly, he has worked as a Senior Software Engineer in HCL Technologies from June 2015 to June 2017. He worked as a Research Assistant at the Shiv Nadar University, Greater Noida where he was actively involved in the development of the first successful Big Data analytics center with IBM and the health system for Apollo Hospital. His research interests include Fake News detection, Rumours Detection, and Analysis of Abusive Content and Sentiment Analysis for Fake content. He is currently developing a real-time fake news detection systems and tracking fake news detection using Deep Learning Techniques.

Dr. Anurag Goswami is an Assistant Professor in the Department of Computer Science Engineering at Bennett University, India. His area of research is in improving software quality via empirical evaluation. He is also involved in research that enables the reduction of the skill gap between academia and industry. He has collaborated with multiple researchers across the globe and is now researching in the area of machine learning to understand the factors where machine learning could help in the improvement of software quality.

Dr. Pratik Narang is an Assistant Professor in the Department of CSIS at BITS Pilani, Rajasthan, India. His research focuses on building systems using Artificial Intelligence and Machine Learning for improving the existing technologies in Cybersecurity and Computer Vision. Pratik completed his undergraduate and doctoral studies from BITS Pilani in 2011 and 2016 respectively. His research was funded by grants from the Ministry of Electronics \& IT, Govt. of India. He has also held research positions with New York University, Abu-Dhabi campus and National University of Singapore, Singapore.

Soumendu Sinha is a Scientist in Smart Sensors Area at CSIR-Central Electronics Engineering Research Institute (CEERI), Pilani. He received his B.E. (Hons.) in Electrical and Electronics Engineering from Birla Institute of Technology and Science (BITS), Pilani and M.Tech. in Advanced Semiconductor Electronics from Academy of Scientific and Innovative Research (AcSIR). Prior to joining CSIR-CEERI, Pilani, he has worked as Research Assistant at National University of Singapore (NUS). His current research interests are in the interdisciplinary areas of Machine Learning and Electrochemical/Biochemical sensors.