IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

A Combined Machine Learning and Deep Learning Techniques based Approach for Fake News Spreaders Detection

[1]Pandarinath Potluri, [2]Satyasriram Potluri, [3]K.Ramalakshmi, [4]LVS Narayana

[1]professor and principal, [2]Imaginer (Data Scientist and Analyst), [3,4]Assistant Professor
[1]Computer Science and Engineering, Swarnandhra Institute of Engineering and Technology, Narasapur, AP
[2]Fractal Analytics, Devarabisanahalli, Bengaluru, India
[3,4]Electronics and Communications Engineering, Swarnandhra Institute of Engineering and Technology, Narasapur, AP

Abstract: The textual data is increasing exponentially in the internet through various social network platforms like discussion Forums, Twitter, Reviews sites, Facebook, Blogs and etc. These platforms changed the method of communication among people. Most of the people are interested to exchange the genuine information in these platforms. Some of them generate false or fake information and spread this information in these platforms. The fake or false information is spreading to defame the reputation of companies, people, services, products, and places. The detection of fake news in people's communication becomes a popular research area in recent times. Most of the researchers proposed several approaches to detect the fake news or false information by analysing the written text. One way of restricting the spreading of information is when every user checks the truth of news content before spreading news into different social media groups. It is very difficult in this abundant information world to know the correctness of the information. In this context, identification of fake news spreaders is useful for the community of people to recognize whether the textual message came from fake source or genuine source. The PAN competition organizers introduced a task of Fake News Spreaders (FNS) detection in 2020. The task is detecting whether the Twitter author is fake news spreader or not. The organizers provided Twitter dataset for fake news spreaders detection. In this work, we developed a new approach by combining the feature representation methodologies of machine learning techniques and concepts of BERT for features extraction. The documents are represented with the combination of features selected by the feature selection algorithm and the documents representation by the BERT model. Support Vector Machine (SVM) classifier is used for evaluating the efficiency of the proposed approach. The SVM classifier with combined vector representation shows best accuracy for fake news spreaders detection**.**

Index Terms **- Fake News, Fake News Spreader, BERT, Term Weight Measure, Feature Selection Algorithm.**

I. INTRODUCTION

In recent times, the people are more trusted on social media platforms for knowing about latest updates on any type of news. This becomes an advantage for some group of people to spread false information in the social media websites. The Fake News (FN) or False News is a type of information created by some people to defame the reputation of people, products and services. In latest times, the fake news spreading was become a common problem in social media platforms. Recent observations told that the fake news is spreading more viral and faster than real news [1]. Fake news spreading generally used for influencing the decisions of people on certain aspects like outcomes of elections [2], management of emergency situations and their responses [3], threatening of public health [4], and citizens trust on social media platforms [5]. The fake news spreaders detection is one primary problem to control spreading of fake news. For example, bots are fake accounts that are propagating fake news through follower networks, which influences stock markets and decisions of elections.

In last decade of time, most of the researchers are concentrated on detection of fake news in different varieties of datasets. The problems of fake news detection are solved in two different ways such as social context model and news content model [6]. The social context model detects the fake news in two ways such as propagation based and credibility based techniques. The propagation based techniques focused on the way fake news is spreading through a social media network. Credibility based techniques investigates the credibility of person who is created and who is spreading these news. In news content model, fake news detected in two ways such as style based detection and knowledge based detection. The style based detection techniques focused on the writing styles of fake news rather than knowledge on content of news. The knowledge based detection techniques focused on the truth-value of the knowledge content of the news.

The controlling of fake news spreading is very important issue to reduce the damage happening to one's reputation. The persons who create and pread fake news are called fake news spreaders. The differentiation of FNSs and Real News Spreaders (RNSs) by seeing the messages came from the author is one important challenge for researcher's community and social media groups. The

dataset of FNSs writings and RNSs writings in the form of text is required to detect whether the new message came from FNS or RNS. PAN competition organized a competition on FNSs detection task in 2020. They provided a Twitter dataset for FNSs detection. The training dataset contains 150 author profiles of FNSs and 150 author profiles of RNSs. In this work, we conducted the experiment on the dataset provided in the PAN competition.

The existing researchers proposed different approaches by using the style based features for distinguishing the author's writing styles, content based features like words used in the text, concepts of machine learning techniques like feature selection algorithms, term weight measures and Deep Learning (DL) techniques like RNN, LSTM, GRU, BERT for FNSs detection. In Machine Learning (ML) techniques based approaches, the features are identified by using different techniques and these features are used for document vectors representation. In deep learning techniques based approaches, the documents are converted into vectors automatically by the models of deep learning. In this work, we are proposing an approach by combining the vector representation of ML techniques and DL techniques. In this approach, the chi-square measure [7] is used as a Feature Selection Algorithm (FSA) for identifying the important words in the dataset. The identified features are used for representing the document vectors. The value of a feature is computed by using Term Weight Measure (TWM) of TFRF [8]. This document vector is combined with the document vector generated by using the BERT model. The combined document vectors are trained with Support Vector Machine (SVM) classifier. This classifier predicts the accuracy of fake news spreaders detection.

This paper is structured in 8 sections. Section 2 analyses the existing works done in the field of fake news spreaders detection. The dataset description is mentioned in section 3. The evaluation measures for representing the performance of proposed approach is explained in section 4. The proposed approach is discussed in section 5 and this section also discusses about FSA, TWM, ML algorithm and BERT model that are used in the proposed approach. The experimentation results of proposed approach are discussed in section 6. The conclusions of this work are explained in section 7 with future directions.

II. EXISTING WORKS ON FAKE NEWS SPREADERS DETECTION

The main complex task in the approaches based on ML techniques is the recognition of appropriate features for discriminating the style of writing in real news spreaders and fake news spreaders. This feature identification problem is avoided in the approaches based on deep learning methods like CNN, RNN, LSTM etc., for detecting the fake news spreaders. In the context of text processing, researchers are popularly used RNN ("Recurrent Neural Networks") method. LSTM ("Long Short Term Memory") is a variety of RNN which is proposed to address the disadvantages of RNN method. Soumayan Bandhu Majumder et al., developed [9] a framework of deep learning by using LSTM. They experimented with Google's pre-trained sentence embedding for generating embedding vectors. These embedding vectors are passed to LSTM as input. Further, the output embedding vector is fed into attention layer for predicting the fake news spreader. The developed approach achieved 72% accuracy for fake news spreaders detection.

Alvaro Lopez et al., experimented [10] with various deep learning methods and identified that the CNN shows good performance for fake news spreaders detection. They proposed a deep learning based method by following two separate ways to solve the task of FNSs detection. First, train the classifier with a sequence of tweet messages and assigns a class label to anonymous author. Secondly, train a classifier with single tweet and voting was conducted for an anonymous tweet with all single tweet classifiers for predicting the class label of author. Based on the kind of encoding used for tweets, three models such as pre-trained multilingual encoder, pre-trained neural net language model and untrained embedding are combined together. The first model considered a layer of untrained embedding for encoding the tweets words. The second model used pre-trained embedding by considering the Feed-Forward NNLM (Neural Net Language Model) [11] with complete tweet encoding for generating 128-dimensional embedding. Finally, the last model used pre-trained multilingual encoder which was trained CNN with 16 languages.

Many research works experimented with the combination of RNN and CNN methods for FNSs detection. Oleg Bakhteev et al., developed [12] a neural network based approach for FNSs detection on dataset of Twitter. They considered this problem as binary classification and treated FNSs and RNSs as two class labels. RNN and CNN methods are applied in their work for handling the dataset of tweets. FastText [13] was used for producing the word embeddings of 100 dimensions. They identified from experiment results that the RNN method performance was slightly higher than the performance of CNN method and also identified that the combination of RNN and CNN methods slightly increases the prediction accuracy of fake news spreaders detection.

BERT ("Bidirectional Encoder Representations from Transformers") is a popular and efficient deep learning method based on transformers for solving problems of NLP ("Natural Language Processing"). Arup Baruah et al., developed [14] a model by using pre-trained large cased BERT for predicting the FNSs. This BERT model was created with 24 layers and 16 attention heads. BERT models generate contextualized word embeddings that were opposite to word embeddings generated through GloVE and Word2Vec. BERT model represents the words with 1024 dimensional vectors. The experiment was carried out with the concatenation of all tweets of individual authors and vectors of these tweets are used for classification. These vectors are generated by using max pooling technique on vectors of 1024 dimensions that are extracted from sub-strings of concatenated string.

Kaushik Amar Das et al., used [15] ELECTRA models by presenting an ensemble classifier for predicting fake news spreaders. The main aim of trained ELECTRA ("Efficiently Learning an Encoder that Classifies Token Replacements Accurately") method [16] was for retaining all BERT capabilities and for addressing the disadvantages of BERT. All the fine-tuned 15 models in the developed ensemble were represented on top of ELECTRA model which was pre-trained. In every model, the pre-trained ELECTRA model generates the embeddings of 256 dimensions. Every model in the ensemble yields a prediction by considering a different random tweet sample of an individual author. The majority voting technique was used for final prediction, where the final label for tweet was decided based on the label which was having most frequency. They created two ensembles separately for two languages that are present in the dataset.

Shih-Hung Wu et al., adopted [17] a BERT pretrained model as tweet classifier for developing a two stage classification method. In the training phase, BERT pre-trained model acted as a classifier for tweets classification and use this classifier for classifying every tweet as possible fake news or real news. Later, the proposed classification method identifies fake news spreaders by testing the proportion of tweets of each author were identified as fake news. The classifier considers the author as FNS when the percentage of fake tweets was larger than a specified threshold. The proposed method attained 0.71% accuracy for fake news spreaders detection on the training dataset of English. However, the accuracy was dropped to 0.56% when experiment conducted on test dataset.

The Transfer Learning (TL) method used the knowledge that was obtained in learning of one model (source model) in development of another model (target model). The TL methods are more helpful when the target dataset contains fewer amounts of training data. In this scenario, the pretrained TL model that was trained on huge amount of source dataset was used for transferring its knowledge to develop a correct model for target dataset even when the target and source datasets have dissimilar features or distributions [18, 19]. H. L. Shashirekha et al., developed [20] a Universal Language Fine-Tuning model by using the methods of transfer learning for predicting the possible FNSs on Twitter. The developed model gathers textual information of wiki for giving training to the Language Model. This model captures the language's common features and this knowledge was transferred for developing a classifier by using a dataset for FNSs detection. The proposed model obtained accuracies of 64% and 62% for FNSs detection in Spanish and English datasets.

Several researchers developed methods for FNS detection by using the combination of both ML and DL techniques. Most of the researchers experienced that the ML based approaches return good accuracies for FNSs detection than the accuracies of DL based approaches. They also noticed that the DL based approaches shows good efficicency when the dataset contains large amount of data. Xinhuan Duan et al., noticed [21] that the fake news spreaders detection was not possible by using a set of features. Then, they trained a BERT fine-tuned model for extracting most relevant language specific features that was used to separate two classes of authors. The BERT pretrained model was considered for determining the word embeddings and these embeddings are fed into a GRU ("Gated Recurrent Unit") [22] to predict the probability of the tweet that belongs to FNS. The authors identified from experiment results that the combination of Hashtags, Emojis, sentiment, and TLSP features obtained best accuracies for fake news spreaders detection.

Hamed Babaei Giglou et al., developed [23] the representation of LSACoNet by grouping various levels of document representations with a FCNN ("Fully Connected Neural Network") classifier. A FCFFNN ("Fully Connected Feed Forward Neural Network") [24] contains one input layer, three hidden layers and output layer. The input layer contains 1024 neurons and 3 hidden layers contain different counts of neurons like 256, 128, and 64 neurons. They conducted experiment with various classifiers such as Stacking Ensemble, Multi-layer Perceptron, Logistic Regression, Linear SVM, RBF SVM, K Nearest Neighbours, Naive Bayes, and Ridge classifier and various representations such as LSA (Latent Semantic Analysis), ConceptNet Numberbatch, N-gram, and TF-IDF. They identified from experimented results that the combination of certain representations are crucial in prediction of fake/real news spreaders.

Roberto Labadie-Tamayo et al., developed [25] two representations for each tweet in the dataset of FNSs detection. The first representation used LSTM and CNN for evaluating tweets at word level, while the second representation used the similar architecture for evaluating tweets at character level without weights sharing. They also conducted experiment with different representation by considering stylistic traits, which consists of correlations among the grammatical structures like nouns, adjectives, word lengths, function words, and so on. They defined total number of 177 features for the text representation. The features are separated into six subsets such as Boolean, character, sentence, paragraph, syntactic and document by using different textual layers. The LSTM nets with attention mechanism (LSTM-Att) were used to build the overall classification model.

Jacobo Lopez Fernandez et al., experimented [26] with various classifiers such as Multilayer Perceptron, traditional RNN with LSTM, Gaussian Naive-Bayes, Support Vector Machine, Gradient Boosting, Stochastic Gradient Descent, and K-nearest Neighbours for FNSs detection. They developed a model for Spanish language based on trigrams and bigrams of words, where the top 1000 features are identified from the total dataset as vocabulary based on their frequency and also considered the punctuation symbols. The linear SVM classifier return best accuracies for FNSs detection in Spanish language and the Gradient Boosting classifier attained good results in English language.

Usman Saeed et al., analysed [27] the impact of DL and machine learning techniques for the FNSs detection. The experiment performed with various methods such as Bi-LSTM, LSTM with and without attention, Logistic Regression (LR), Decision tree (DT), SVM, Multi-Layer Perceptron (MLP), and KNN for evaluating the proposed approach. The researchers considered only English dataset for experimentation. They generated GloVe embeddings [28] of 50 dimensional vectors for deep learning methods and the input feature vectors are generated for machine learning methods by using CountVectorizer and TF-IDF transformer.

Anu Shrestha et al., developed [29] a machine learning techniques based approach by using various types of features such as writing style based features specific to Twitter, terms based TFIDF scores, character n-grams (where n range is from 1 to 3), semantic embeddings generated through BERT, features specific to sentiment in the tweets. They used various classifiers such as Random Forest, Extra Trees, Logistic Regression, and SVM with linear kernel in their experiment. For Spanish language dataset, they experimented with Extra Trees as classifier for n-grams, Logistic Regression for sentiment analysis, Random Forest for tweet embedding, and SVM for style features. For English language dataset, they experimented with SVM for both n-grams and tweet embeddings, Logistic Regression for sentiment analysis, and Extra Trees as the classifier for style features. N-grams achieved highest accuracy of 0.72 for FNSs detection on English dataset, followed by BERT embeddings attained an accuracy of 0.69, sentiment features obtained an accuracy of 0.66 and style based features return an accuracy of 0.64.

III. DATASET PROPERTIES

In this work, the FN spreaders detection dataset was taken from the 2020 PAN competition sub-task of FNSs detection [30]. The information about the dataset is presented in Table 1.

Table 1: The information about FNSs detection dataset

Language	Training Dataset		Testing Dataset		Total
	FNSs	RNSs	FNSs	RNSs	
Spanish	150	150	100	100	500
English	150	150	100	100	500

The organizers of competition released Twitter dataset for fake news spreaders detection in two different languages like English and Spanish. The training dataset in both languages contains 150 FNSs and 150 RNSs tweets and each author contain 100 tweets. The testing dataset contains 100 FNSs and 100 RNSs tweets and each author contain 100 tweets. The dataset is balanced which means that both FNSs and RNSs contain equal number of authors. The organizers hides the sensible information like "URL", "hashtag", "rt" (re-tweet), and "user" in the dataset by replacing with standard keywords to make author as anonymous. In this work, the English dataset is considered for experimentation. Some of the characteristics of English dataset are out of 500 authors 284 authors are not used emojis in tweets, average tweet length is nearly 15 words, the length of longest tweet is 86 words and the length of shortest tweet is one word, and 343 author's documents contain all unique tweets out of 500 authors [31]. In this work, the English language dataset is considered for fake news spreaders detection.

IV. Evaluation Measures

The classification metrics play an important role for evaluating the effectiveness of proposed models. The researchers in the task of FNS classification used various classification metrics like Recall, Precision, Accuracy and F1-score for evaluating the efficiency of their proposed approach [32]. To define these metrics, we used confusion matrix. Table 2 displays the confusion matrix.

Table 2: Confusion Matrix

		Predicted Class	
Actual Class		**FNSs**	**RNSs**
	FNSs	TRPO (TRue POsitive)	FANE (FAlse NEgative)
	RNSs	FAPO (FAlse POsitive)	TRNE (TRue NEgative)

From Table 2, the parameters used to represent the evaluation metrics are TRPO, TRNE, FAPO and FANE. True Positive (TRPO) refers to samples count in the dataset that are predicted as FNSs. True Negative (TRNE) refers to samples count in the dataset that are predicted as RNSs. False Positive (FAPO) refers to samples count in the dataset that are incorrectly predicted as FNSs. False Negative (FANE) refers to samples count in the dataset that are incorrectly predicted as RNSs.

Accuracy (ACC) is the most intuitive metric and it can be defined as the ratio among the number of samples that are correctly predicted and the total count of predictions. Equation (1) is used to determine the accuracy of a model M.

$$ACC(M) = \frac{TRPO + TRNE}{TRPO + FAPO + FANE + TRNE} \quad (1)$$

Precision (P) is the count of positive predictions for a class divided by the count of total predictions for that class. Equation (2) is used for calculating the precision of a model M.

$$P(M) = \frac{TRPO}{TRPO + FAPO} \quad (2)$$

Recall (R) is the count of positive predictions for a class divided by the total count of instances in the positive class. Equation (3) is used for calculating the recall of a model M.

$$R(M) = \frac{TRPO}{TRPO + FANE} \quad (3)$$

F-score is a combination of precision and recall where the relative importance of the two metrics can be specified by using β. Equation (4) is used for determining the F-Score of a model M.

$$F_\beta(M) = \left(1 + \beta^2\right) \frac{P \times R}{\beta^2 \left(P + R\right)} \quad (4)$$

F1-score (F1) is widely used in the literature which is essentially the harmonic mean of recall and precision. Equation (5) is used for calculating the F1-Score of a model M.

$$F_1(M) = 2 \times \frac{P \times R}{\left(P + R\right)} \quad (5)$$

In this work, the experiment results are presented by using Accuracy evaluation metric.

V. Proposed Approach

In this work, we proposed an approach by combining the techniques of machine learning and deep leaning. The steps followed in proposed approach are displayed in Figure 1.

In this proposed approach, the document vectors are created by using ML techniques and DL techniques. Finally, these two vectors are concatenated to produce the final document vector for training the classification algorithm. In this approach, the first step is collection of standard dataset for experimentation. We used a standard dataset of fake news spreaders detection provided in the PAN competition 2020. Once collected the dataset, the next step in machine learning based vector representation is identifying the suitable preprocessing techniques. In this work, we used lowercase conversion, stopword removal and stemming as preprocessing techniques. Lowercase conversion converts the all characters in a text into lower case letters. Stopwords are words like prepositions, punctuations, pronouns etc., which are frequently occurred in the dataset but they are not having any distinguishing power [33]. We removed the stop-words from reviews dataset. Stemming is used to convert the words into its base form by using stemming algorithms. Stemming reduces the unique words in a dataset. After applying pre-processing techniques, the data become ready to process the next step. After cleaning the non-informative words, apply FSA to find the important features based on the scores of features. We extracted top scored features of 4000 as feature set. The documents are represented as vectors with identified features and vector value is determined by using the TWM.

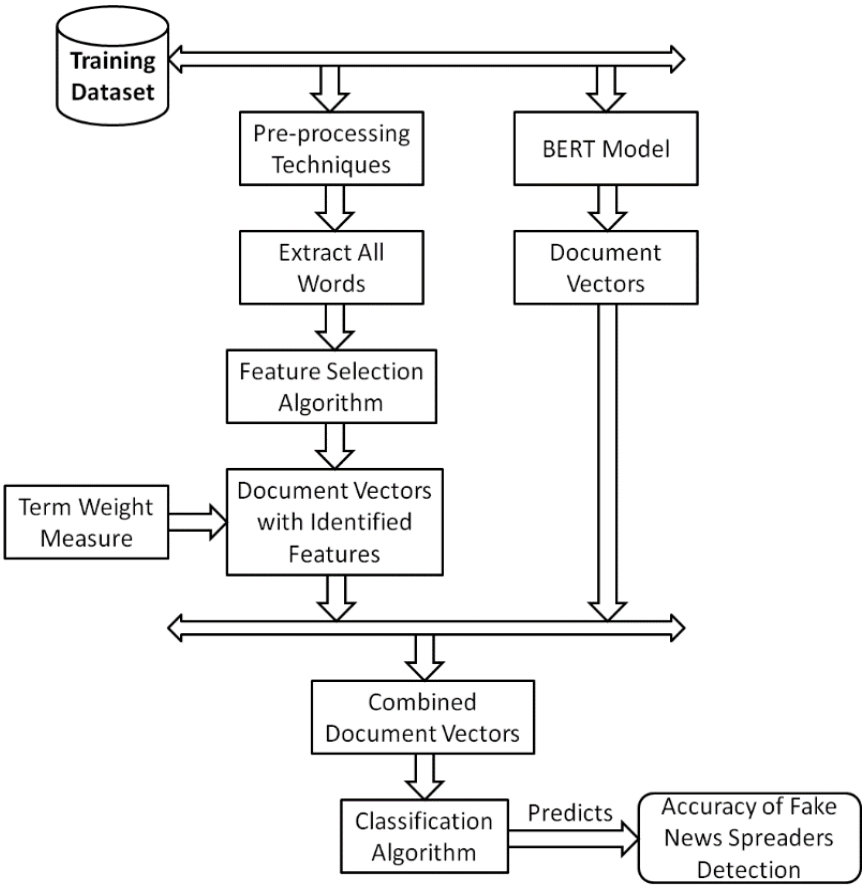

Figure 1: The proposed Approach for FNSs Detection

In other way of vector representation, the deep learning technique of BERT is used for generating the document vectors by aggregating the vector representations of words in that document. Finally, concatenate the feature based vector representation and BERT based vector representation. These concatenated document vectors are forwarded to classification algorithms for generating the model for classification. This model predicts the accuracy of fake news spreaders detection.

5.1 Feature Selection Algorithm - Chi Square (CHI2) Measure

Chi-Square (χ^2) is a FS technique for recognizing the significant features by using the concept of relationship among feature and class [34]. The CHI2 measure assigns values to the terms in between 0 to 1 [35]. The term score value 1 indicates there exist a strong relevancy among a term and a class and score value 0 denotes the term is totally not relevant to the class. CHI2 of a term T_i specific to class C_j is computed by using Equation (6).

$$\chi^2(T_i, C_j) = \frac{N \times (a_i d_i - b_i c_i)^2}{(a_i + b_i) \times (a_i + c_i) \times (b_i + d_i) \times (c_i + d_i)} \quad (6)$$

Where, N denoted documents count in dataset, a_i and b_i are the documents count in class C_j those contain term T_i and those doesn't contain term T_i. c_i and d_i are the documents count in other than class C_j those contain term T_i and those doesn't contain term T_i.

5.2 Term Weight Measure – TFRF ("Term Frequency and Relevance Frequency") Measure

Lan, M. et al., developed [36] TFRF measure for determining the term importance in a document. According to this TFRF measure, the terms that are occurred in positive class of documents are having more weight than the terms that are appeared in negative class of documents when the term weight is computed in positive class of document. These terms are more helpful to separate the positive

class documents from negative class documents. This method was achieved best accuracies in several text classification problems [37]. This is the reason we used this measure in our experiment. The TFRF value of a term T_i in document D_k is computed by using Equation (7).

$$TFRF\left(T_i, D_k\right) = TF_{ik} \ * \ \log\left(\frac{A}{MAX\left(B,1\right)} + 2\right) \tag{7}$$

In Equation (7), A is count of documents in positive class those contain the term T_i, B is documents count in negative class those contain the term T_i, TF_{ik} is number of times T_i appeared in D_k.

5.3 BERT

Bengio et al., (2003) introduced [38] the concept of word embeddings. Word Embeddings is a technique used for feature engineering. In word embedding approach, every word is represented as a real-valued, dense vector by considering the distributional semantics and syntactic information [39]. Word embeddings are used to replace the usage of one-hot encoding representation and overcome many problems faced by such encoding schema. A relation between words in the dataset is drawn by the learning process and translated into numbers updated in the distributed representation vector ending up in distribution similar words adjacent to each other in the continuous space. There are many ways to represent the words as vectors. In this work, we obtain the word vectors by constructing BERT model.

As opposed to context-free word embeddings such as Word2Vec, GloVe and FastText, BERT uses transformers, an attention mechanism that learns the contextual relationships between input tokens (which can be word or character sequences). The transformer model tries to learn using the entire window of input tokens, with no directional input (i.e. reading the left or right most words first). This results in what is known as a "bi-directional model". At the time of proposal, BERT is arguably the state-of-the-art model in almost all-natural language problems. BERT relies on the attention mechanism. BERT is a group of transformers that help BERT to understand the context of the natural language by looking at both directions. As an example, considering the sentence, 'I visited the bank.' If the sentence is followed by the word 'to deposit funds,' the context of the sentence helps us understand the term related to a 'financial institution,' given the scope of other words like 'blood bank' and 'river bank.'

BERT works in two major steps, the first step is commonly known as the "pre-training" step which is a semi-supervised task where the objective is to predict masked words from input sentences and then predict the next sentence. The second major step is widely known as "fine tuning". The fine tuning process of BERT is an example of transfer learning. Depending on the task, the network will change the input and outputs accordingly (such as sentence classification, question answering, and named entity recognition). BERT is primarily used on classification tasks, however, feature extraction can also be done to generate word embeddings to be put in existing models. BERT has been provided with two variants based on the number of transformers they contain at the time of proposal, one being base BERT with 12x transformers and 110 million parameters, and the other being large BERT with 24x transformers and 340 million parameters.

The base BERT generate the word embeddings size of 768 dimensional and large BERT generate the word embeddings size of 1024 dimensional. In this work, we used base BERT to generate the word embeddings. The base BERT used 12 transformers and each word in the document passed through these 12 transformers. Each transformer creates 768-dimensional representation for each word and transfers to next transformer. In general, the 12^{th} transformer output is considered for final vector representation of word. But, in this work, we considered the concatenations of different transformer layers outputs as word embeddings for generating the document vector representation.

5.4 Machine Learning (ML) Algorithms

The ML algorithms are used for evaluating the effectiveness of proposed approaches. In this work, SVM is used as a machine learning algorithm. Support Vector Machines originally introduced in 1992 [40] and become increasingly popular for supervised classification tasks and regression tasks also. After representing our data points as vectors in a vector space, the goal is to find a hyperplane that divides the two classes of data. Specifically, the algorithm finds a maximum margin hyperplane which is meant to maximize the distance between the hyperplane and the nearest data points of either class. Most real-world data is not linearly separable. SVM has two mechanisms to deal with this nonlinearity. The first approach allows the algorithm to make mistakes and misclassify a few examples in the training set. It introduces slack variables to penalize the model for the misclassified examples. The second approach is what is popularly known as the "kernel trick". They transform the feature space to a higher dimensional space, and the intuition is that the data would be linearly separable in that higher-dimensional space. This allows the algorithm to learn better complex decision boundaries. There exist many types of kernel functions such as polynomial, linear and Gaussian radial basis function (RBF) [41].

VI. Experimental Results

In this work, the experiment performed with the combination of ML techniques and DL techniques for generating the document vectors. The chi-square FSA is used for identifying the important features. We used top scored features of 4000 for experimentation. The experiment started with top scored 1000 features and incremented by 1000 in every iteration. The vector value is calculated by using TFRF measure. In each iteration, these features are combined with the output of BERT model. The small cased BERT is used in this experimentation. This BERT model has12 layers and each layer output is forwarded to next layer. The experiment conducted with concatenation of different layers output for generating the document vector. All the words of a document are passed through all BERT layers and BERT model generates the 768-dimensional word embedding for each word after every layer. The document vector is generated by aggregating the 768-dimensional vectors of all words in that document. In this work, the first iteration starts with the BERT outcome of 12^{th} layer with 1000 features identified by feature selection algorithm. Second iteration is executed with

concatenation of 11th and 12th layers outputs of BERT with 2000 features identified by feature selection algorithm. The third iteration is implemented with concatenation of 10th, 11th and 12th layers outputs of BERT with 3000 features identified by feature selection algorithm. The fourth iteration is implemented with concatenation of 9th, 10th, 11th and 12th layers outputs of BERT with 4000 features identified by FSA. After 4th iteration it was observed that the accuracies of FNSs detection were reduced. The Table 2 displays the accuracies of proposed approach with SVM classifier for FNSs detection.

Table 2: The SVM Accuracies of Proposed Approach for FNSs Detection

Features Through FSA	Vector size Through BERT	Total Document Vector Size	Accuracy of Fake News Spreaders Detection
1000	768	1768	83.56
2000	1536	3536	85.49
3000	2304	5304	86.23
4000	3072	7072	88.67

Form Table 2, the concatenation of top scored 4000 features and outputs of last 4 layers achieved best accuracy of 88.67 for FNSs detection. We observed that the accuracy of fake news spreaders detection is improved when the number of features is augmented up to 4000 and concatenation of last four layers output of BERT. It is also observed that the accuracy of fake news spreaders detection was reduced when the number of features is used more than 4000 in the experiment.

VII. Conclusion and Future Scope

Due to the exponential growth in user generated text in social media platforms, the fake information in the form of text also steadily increased. The fake news is fabricated information which is created and forwarded by people to change the opinion of people and to improve hate feeling on set of people or products. Identification of persons who spread the fake information is one challenging task in the research community. In this work, we proposed an approach by combining the methods of ML and DL techniques to predict the fake news spreaders. The chi-square algorithm was used for identifying the best informative features from the dataset. The BERT model is used for representing the documents as vectors. The experiment conducted with the concatenation of feature based document vector and BERT based document vector. The SVM classifier with proposed document vector representation attained best accuracy of 88.67 for fake news spreaders detection.

In future work, we are planning to develop a new TWM for improving the importance of a feature in document vector representation and also planning to identify other DL techniques that are suitable for increasing the accuracy of FNSs detection.

References

[1] Vosoughi, S., Roy, D., Aral, S.: The spread of true and false news online. Science 359(6380), 1146–1151 (2018)

[2] Isaak, J., Hanna, M.J.: User data privacy: Facebook, cambridge analytica, and privacy protection. Computer 51(8), 56–59 (2018)

[3] Spiro, E.S., Fitzhugh, S., Sutton, J., Pierski, N., Greczek, M., Butts, C.T.: Rumoring during extreme events: A case study of deepwater horizon 2010. In: Proceedings of the 4th Annual ACM Web Science Conference. pp. 275–283 (2012)

[4] Vogel, L.: Viral misinformation threatens public health (2017)

[5] Barometer, E.T.: Edelman trust barometer global report. Edelman, available at: https://www. edelman. com/sites/g/files/aatuss191/files/2019-02/2019_Edelman_Trust_Barometer_Global_Report_2.pdf (2019)

[6] Zhou, X., Zafarani, R.: Fake news: A survey of research, detection methods, and opportunities. arXiv preprint arXiv:1812.00315 (2018)

[7] Raghunadha Reddy T, Vishnu Vardhan B, GopiChand M, Karunakar K, "Gender prediction in Author Profiling using ReliefF Feature Selection Algorithm", Proceedings in Advances in Intelligent Systems and Computing, Volume 695, PP. 169-176, 2018.

[8] Swathi Ch, Karunakar K, Archana G, T. Raghunadha Reddy, "A New Term Weight Measure for Gender Prediction in Author Profiling", Proceedings in Advances in Intelligent Systems and Computing, Volume 695, PP. 11-18, 2018.

[9] Soumayan Bandhu Majumder, Dipankar Das, " Detecting Fake News Spreaders on Twitter Using Universal Sentence Encoder", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[10] Alvaro Lopez and Pasqual Marti, "Profiling Fake News Spreaders on Twitter", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[11] Yoshua Bengio, R_ejean Ducharme, P.V.C.J.: A neural probabilistic language model. Journal of Machine Learning Research 3, 1137-1155 (2003)

[12] Oleg Bakhteev, Aleksandr Ogaltsov, and Petr Ostroukhov, "Fake news spreader detection using neural tweet aggregation", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[13] Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics 5, 135–146 (2017)

[14] Arup Baruah, Kaushik Amar Das, Ferdous Ahmed Barbhuiya, and Kuntal Dey, "Automatic Detection of Fake News Spreaders Using BERT", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[15] Kaushik Amar Das, Arup Baruah, Ferdous Ahmed Barbhuiya, and Kuntal Dey, "Ensemble of ELECTRA for Profiling Fake News Spreaders", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[16] Clark, K., Luong, M.T., Le, Q.V., Manning, C.D.: Electra: Pre-training text encoders as discriminators rather than generators (2020)

[17] Shih-Hung Wu and Sheng-Lun Chien, "A BERT based Two-stage Fake News Spreaders Profiling System", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[18] Faltl Sandra, Michael Schimpke and Constantin Hackober. "Ulmfit: State-Of-The-Art in Text Analysis". Seminar Information Systems (WS18/19), 2019.

[19] Semwal Tushar, Promod Yenigalla, Gaurav Mathur, and Shivashankar B. Nair. "A Practitioners Guide to Transfer Learning for Text Classification Using Convolution Neural Networks". In Proceedings of the 2018 Society for Industrial and Applied Mathematics (SIAM) International Conference on Data Mining, pp.513-521, 2018.

[20] H. L. Shashirekha, F. Balouchzahi, "ULMFiT for Twitter Fake News Spreader Profiling", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[21] Xinhuan Duan, Elham Naghizade, Damiano Spina, and Xiuzhen Zhang, "RMIT at PAN-CLEF 2020: Profiling Fake News Spreaders on Twitter", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[22] Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP'14). pp. 1724–1734 (2014)

[23] Hamed Babaei Giglou, Jafar Razmara, Mostafa Rahgouy, and Mahsa Sanaei, "LSACoNet: A Combination of Lexical and Conceptual Features for Analysis of Fake News Spreaders on Twitter", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[24] Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016), http://www.deeplearningbook.org

[25] Roberto Labadie-Tamayo, Daniel Castro-Castro, and Reynier Ortega-Bueno, "Fusing Stylistic Features with Deep-learning Methods for Profiling Fake News Spreaders", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[26] Jacobo Lopez Fernandez and Juan Antonio Lopez Ramirez, "Approaches to the Profiling Fake News Spreaders on Twitter Task in English and Spanish", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[27] Usman Saeed, Hammad Fahim, and Dr. Farid Shirazi, "Profiling Fake News Spreaders on Twitter", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[28] Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543 (2014), Doha, Qatar, October 2014. Association for Computational Linguistics.

[29] Anu Shrestha, Francesca Spezzano, and Abishai Joy, "Detecting Fake News Spreaders in Social Networks via Linguistic and Personality Features", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[30] Rangel, F., Giachanou, A., Ghanem, B., Rosso, P.: Overview of the 8th Author Profiling Task at PAN 2020: Profiling Fake News Spreaders on Twitter. In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) CLEF 2020 Labs and Workshops, Notebook Papers. CEUR Workshop Proceedings (Sep 2020), CEUR-WS.org

[31] Kaushik Amar Das, Arup Baruah, Ferdous Ahmed Barbhuiya, and Kuntal Dey, "Ensemble of ELECTRA for Profiling Fake News Spreaders", Notebook for PAN at CLEF 2020, 2020, 22-25 September 2020, Thessaloniki, Greece.

[32] Dr. T. Murali Mohan, Dr. T. Raghunadha Reddy, Dr. A. Balakrishna, T V Satya Sheela, "Stylistic features based Approach for Bot Detection", Design Engineering, Issue: 7, 2021, Pages: 12699 – 12712.

[33] P Buddha Reddy, Dr. T Murali Mohan, Dr. P Vamsi Krishna Raja and Dr. T Raghunadha Reddy, "A Novel Approach for Authorship Verification", SPRINGER 3rd International Conference on Data Engineering and Communication Technology (ICDECT), Stanley College of Engineering and Technology for Women, Abids, Hyderabad, Telangana, India, 15 – 16 March, 2019.

[34] Yang Y. and Pedersen J., "A Comparative Study on Feature Selection in Text Categorization," in Proceedings of the 14th International Conference on Machine Learning, Nashville, pp. 412-420, 1997.

[35] Srikanth Reddy G, Murali Mohan T, Raghunadha Reddy T, "Author Profiling Approach for Location Prediction", first international conference on Artificial Intelligence and Cognitive computing conducted by MLR Institute of Technology, Dundigal, Hyderabad, 02-03 February, 2018.

[36] Lan, M., Tan, C. L., Su, J., & Lu, Y. (2009). Supervised and traditional term weighting methods for automatic text categorization. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31 (4), 721–735. http://doi.org/10.1109/TPAMI.2008.110

[37] Para Upendar, T Murali Mohan, S. K. LokeshNaik, T Raghunadha Reddy, "A Novel Approach for Predicting Nativity Language of the Authors by Analyzing their Written Texts", SPRINGER 6th International Conference on Innovations in Computer Science and Engineering, Guru Nanak Institute of Technology, Hyderabad, Telangana, 17-18, August 2018.

[38] Bengio, Yoshua, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. "A neural probabilistic language model. Journal of machine learning research, Vol. 3, No." (2003): 1137-1155.

[39] T. Raghunadha Reddy, P. Vijayapal Reddy, T Murali Mohan, Raju Dara, "An Approach for Suggestion Mining based on Deep Learning Techniques", International Conference on Computer Vision, High Performance Computing, Smart Devices and Networks(CHSN-2020), 28-29 December, 2020, JNTUK, Kakinada, Andhra Pradesh.

[40] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classi_ers. In Proceedings of the Fifth Annual Workshop on Computational Learning Theory, COLT '92, pages 144-152, New York, NY, USA, 1992. ACM.

[41] Thomas Hofmann, Bernhard Sch • olkopf, and Alexander J. Smola. Kernel methods in machine learning. Ann. Statist., 36(3):1171-1220, 06 2008.