A Hybrid TF-IDF and N-Grams Based Feature Extraction Approach for Accurate Detection of Fake News on Twitter Data

SUHASINI V , Dr . N.VIMALA

Research Scholar, Department of Computer Science, L.R.G Government Arts College (W), Tirupur

E-Mail: hasini_ss86@yahoo.com

Assistant Professor, Department of Computer Science, L.R.G Government Arts College (W), Tirupur

E-Mail:vimalananjappan@gmai l.com

Article History: Received: 10 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 7 June2021

Abstract

As there is an exponential growth of social networks and due to large usage of social media, there is an increasing demand for data in the web for the users which leads to current inclinations concepts in the area of research. Sentiment, text analysis and social media analysis, especially in user reviews and a tweet has become a popular area of research. Fake and Real data classification from user responses plays a key role in fake detection on social media platforms. Fake news and lack of trust in the media are growing problems with huge difficulties in our society. Evidently in a deceptive story or fake news in social Medias leads to change its description. The main objective of this research is to detect the fake news, which is a classic text classification problem with a straightforward proposition. There is needed to build a model that can differentiate between “Real” news and “Fake” news with NLP (Natural Language Processing) and ML (Machine Learning) techniques for discovering the 'fake news', or deceptive news stories that arises from the defective bases. Often, some preprocessing steps and feature extraction techniques are applied to obtain features from twitter data to improve the accuracy using supervised classification algorithm. This paper elucidates the ML classification approaches with different feature extraction techniques to obtain a text analysis and the results obtained are compared to identify the best possible approach.

Keyword: Feature Extraction, Fake News, Twitter, SVM

1. Introduction

Social media has substituted the usual media. It has also developed as one of the key platforms for spreading news which travel faster and also easier than traditional news bases due to low cost internet convenience. However, not all the news published on social media is genuine and it may come from unverified sources. Fake information can be created and spread easily through social media and this fake news can potentially or intentionally mislead readers. The extensive spread of fake news brings negative impact not only to individual but also society. So, fake news may affect how readers perceive online news on social media and indirectly mislead the way they respond to real news. However, some existing manual fact in section websites are established to inspect if a news is realistic, it does not measure with the bulk of the wide range online facts, particularly on social media [1]. To solve this problem, many mechanized fact inspection applications are established to tackle the requirement for automation and also scalability. Numerous computational techniques are available which helps to spot the false contents. Many of these methods are utilized for fact checking websites like “Politi Fact” also “Snopes.” Usually, several repositories are conserved using researchers, which have lists of websites that are recognized as uncertain and also fake. However, the problem with these resources is that human expertise is required to identify fake articles. More importantly, the fact checking websites contain articles from particular domains such as politics, entertainment, sports, and technology and they are not generalized to identify fake news articles.

Text mining is the process of discovering useful and interesting knowledge from unstructured text. In order to determine information from unstructured text data, the 1st step is to convert text data into a wieldy representation. A common practice is to model a text in a document which has set of word features, i.e., “bag of words” (BOW). Before feature extraction process, in this method involves data pre-processing in the initial stage and then extract the requires features from the preprocessed data and finally the fake news is detected from the extracted features using ML algorithms [2].

For efficient use of learning system, feature extraction method is used which helps to reduce the unwanted variations in the twitter data and avoid the computational expensive. Improving the robustness of feature usually reduces the effort from classifier which helps to improve the classification system performance. Feature extraction starts from an initial set of measured (preprocessed) data and builds derived values intended to be informative, facilitating the subsequent learning steps and in some cases leading to better human interpretations [3].

This paper explores new hybrid features extraction techniques that are not thoroughly explored in the existing systems. These features, Support Vector Machine learning algorithms which help to distinguish fake tweets from real. A SVM uses classification algorithms for two-group classification problems. After giving an SVM model sets of labeled training data for each category and it categorize new text.

The hybrid systems have proven to be useful in a wide variety of applications. As the models have the tendency to reduce error rate by using techniques such as TF-IDF and N-Grams. TF-IDF is a statistical measure that evaluates how relevant a word is to a document in a collection of documents. It has many uses, most importantly in automated text analysis, and is very useful for scoring words in machine learning algorithms for Natural Language Processing. N-gram is a contiguous sequence of n-items from a given sample of text or speech. These techniques facilitate the feature extraction process in an effective and efficient manner.

2. Literature Review

Yang et al. [5] proposed an efficient model for early detection of fake news through classifying news propagation paths using a multivariate time series. They realized a new deep learning model, which was comprised of four major components, i.e., propagation path construction and transformation, Recurrent Neural Network (RNN) based propagation path representation, CNN-based propagation path representation, and propagation path classification, which were integrated together to detect fake news at the early stage of its propagation.

Mykhailo Granik et al., [6] proposed a simple approach for fake news detection using naïve Bayes classifier. This approach was implemented as a software system and tested against a data set of Face book news posts. They achieved classification accuracy of approximately 74% using naïve Bayes classifier.

With the help of Machine learning and natural language processing, Anjali Jainet al.,[7] tried to aggregate the news and determine whether the news is real or fake using Support Vector Machine. The proposed model is working well and defining the correctness of results up to 93.6% of accuracy.

Wangetal.[8] proposed an Event Adversarial Neural Network (EANN), which consists of three main components: the multi-modal feature extractor, the fake news detector, and the event discriminator. The multi-modal feature extractor is responsible for extracting the textual and visual features from posts. It cooperates with the fake news detect or to learn the discriminable representation for the detection of fake news.

Hardalovetal.[9] used a combination of linguistic, credibility and semantic features to differentiate between real and fake news. In their work, linguistic features include (weighted) n-grams and normalized number of unique words per article. Credibility features include capitalization, punctuation, pronoun usage and sentiment polarity features generated from lexicons. Text semantics were analyzed using embedding vectors method.

Ma et al. [10] Observed changes in linguistic properties of messages over the lifetime of a rumor using Support Vector Machine (SVM) based on time series features, then, they showed good results in the early detection of an emerging rumor.

3. Proposed methodology

3.1 Proposed Framework

In the proposed framework, as illustrated in Figure 1. In this model to emerge the existing research by introducing hybrid techniques with TF-IDF and N-Grams to classify tweets as real or fake. The hybrid technique along with preprocessed dataset used in this paper is the novelty of our proposed approach. The below figure shows the block diagram for the proposed architecture. This system consists of Preprocessing, Feature extraction, Feature selection and Classification models.

3.2 Pre-Processing

Preprocessing operations are very important for analyzing the tweets for reaching best results. Many preprocessing steps applied on unstructured twitter data’s like URL, Punctuation and User name removal then Letter casing, Tokenizing, Stop word removal, Stemming and Lemmatization to make a standard as well as structured dataset. Once the steps are completed, this research moves to the next main method called feature extraction. Extraction of valuable words from the tweet is called as feature extraction.

3.3 Feature Extraction

Feature Extraction (FE) is very significant step in data mining system. It enumerates the feature words which are extracted from the text to characterize the text information and also translates them from an unstructured original text data into a structured data. After completing this step only a system can distinguish-process, describe and replace the text by dimensionally reducing the text word space to establish its mathematical model. In the process of extracting text feature, irrelevant or redundant features will be deleted, and finally the important features (sentences, words or characters, etc.) will be combined with their weights to reflect the information contained in the text.

Text characterization based on word statistics is often used to extract text features. The BOW (bag of words) and the TF-IDF (term frequency–inverse document frequency) are the most typical models. These models can simplify the process of extraction and it is easy to understand. However, when extracting words for text feature, each word in the text is treated as a separate unit, so the Semantic features of the text cannot be effectively obtained.The Proposed model is demonstrated as in Fig.1.

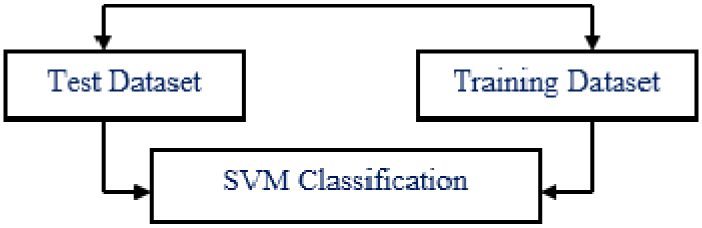

Fig.1.Proposed Framework

Extracting the semantic features of each word in the text needs to be measured by the context where the word is located. Word2Vector, which trains the corresponding word vector based on the context of the word in the text, plays an important role in extracting the semantic of words. Word2Vector is a technique for transforming word representation into space vector. It mainly uses the idea of machine learning to train corpus, by associating context of each word and mapping them into different N-dimensional vector.

In this way, the semantic features of each word can be expressed and recognized by the computer. For the data that already has semantic features, representations among words need to calculate semantic similarity and uniformly store synonyms. Using the density clustering algorithm in machine learning, the words with similar meanings can be clustered. This algorithm can flexibly control the minimum value of the distance of word vector (word similarity) in each cluster, and does not need to set the number of cluster in advance, which is very practical for the case where the number of cluster is not clear and not strict.

a. Term Frequency-Inverse Document Frequency (TF-IDF)

It is very common algorithm to transform text into a meaningful representation of numbers which is used to fit machine algorithm for prediction. TF-IDF is a weight metric which determines the importance of word for that document**.**

TF:

Term Frequency measures number of times a particular term t occurred in a document d. Frequency increases when the term has occurred multiple times.TF is calculated by taking ratio of frequency of term t in document td to number of terms in that particular document d.

$$\text{TF (t,d)}=\frac{\text{Number of times term t appears in a document d}}{\text{Total number terms in a document d}}$$

IDF:

TF processes simply the frequency of a word /term’t’ present in dataset. Some terms like stop words occur multiple times but may not be useful. Hence Inverse Document Frequency (IDF) is used to measure term’s importance. IDF gives more importance to the rarely occurring terms in the document d. IDF is calculated as:

$$\text{IDF (t)} = \log_e \frac{\text{Total number of documents}}{\text{Total number of documents with term t}}$$

The final weight for a term t in a document d is calculated as:

$$\text{T F - IDF (t,d)} = \text{T F(t,d)}\times\text{IDF(t)}$$

The above equations are tried to explain mathematical concept behind all the process like TF-IDF calculation by using Term Frequency and Inverse Document Frequency. Usually TF-IDF Vectorizer consider over all document weightage of a word. It aids in allocating with maximum occurrence words. TF-IDF Vectorizer weights the word counts by a measure of how often they appear in the documents. It is text vectorization based on the Bag of words (BoW) model. This method does well than the BoW model as it considers the significance of the word in a data set (text data).

a. Bag of Words (BoW):

One of the basic methods to translate text into structured features is called as BoW. These methods imply breaks apart the words in the review text into individual word count statistics.

Limitation:

- The main drawback of BoW method is that does not characterize the semantic meaning of the words.

- TF-IDF gives more weightage to the word that is rare in the data set.

- TF-IDF affords more significance to the word, which is more recurrent in the data set.

- Furthermore, the computation cost of TF-IDF is low if the vocabulary is huge.

This limitation of TF-IDF can be overcome by another technique such as word2Vec and N-Grams.

b. N-Grams

In the N-Gram technique, the text is structured in a matrix (term, weight) in which each term receives a weight, usually the frequency that appears. The terms can be the basic unit (token) or composite: 1, 2, 3... N-grams. Bag of n-grams is also called as bag of words. An N-gram is simply any sequence of n tokens (words).

An N-gram method is an adjacent sequence of ‘n‘items in a given sequence of input text in the area of computational morphology. The things may be syllables, phonemes, letters, words or else base pairs affording to the use. Typically, this approach collects input data from a text or else speech corpus.

Limitations

N-gram method~~s~~ include the following drawbacks:

- This method works well when the N value is high but this leads to lots of computation overhead, which needs huge computation power in point of RAM.

- This approach is a sparse exemplification of language. Normally, this model is build based on the possibility of words co-occurring in dataset. It will assign zero probability to every word, which do not exist in the training corpus.

c. TF-IDF and N-gram (proposed) approach

In this method follows Bag-of-Words (BoW) model for extracting TF-IDF features from the character N-grams contained within each event description. In this research, bi-grams process is utilized to exclude the N-grams occurring less than two event descriptions. It is observed during this experiment that these parameters could be slightly modified without important impact on the classification results. Here the dimensionality of the TF-IDF vectors varies depending on the training set size, and each event description is represented by a large sparse vector instead of the short full vector used in the word embedding representation.

Input: Each word from vector as Term T, All vectors V[i…n], n=get first 2 grams of the input string

Output: TF-IDF weight for each T

- **Step1:** Vector= {(c1,c2),(c2,c3),(c3,c4)….(cm, cn)}
- **Step2:**Aspectsavailableineachcomment
- **Step3:**D={cmt1,cmt2,cmt3….cmtn} and comments available in each document Calculate the TF score as
- **Step4:**TF(t,d)=(t,d);t=specific term; d=‘particular document in a word is to be originate‘.
- **Step 5:** IDF = t → sum(d)
- **Step 6:** Return TF*IDF

The above algorithm shows the pseudo code of the proposed method. It splits input corpus of the 2-gram variable for the bigram process. Then, TF-IDF provides the availability of the current vector and store into the feature data base. It is used to identify the density of the test object.

3.4 Machine Learning (ML) approach:

Machine learning investigates how computers can learn based on data. A main research area is for computer programs to automatically learn to recognize complex patterns and make intelligent decisions based on data.

ML algorithms are used to solve the opinion mining as a regular text classification problem that makes use of scientific features. The research have a set of training records D={X1, X2...Xn} where each record labeled to a class 'A' and class 'B'. The classification model is related to the features in the testing record to one of the class labels. Then for a given instance (features) of unknown class, the model is used to predict a class label for it. The supervised ML techniques based on the existence of labeled training features.

The concept of classification in machine learning is focused on building a model that separates data into distinct classes, for example, "real" or "fake" news in the context of this research. ML classification approach is a method of Supervised Learning. This model is built by inputting a set of training data for which the classes are pre- labeled as "real" or fake" news in order to learn the algorithm. The model is then used by inputting a different dataset (i.e. commonly called validation/training dataset) for which the classes are withdrawn, letting the model to predict their class value based on what it has learned from the training set. There are several well-known algorithms for classification machine learning including Decision Tree (DT), Naïve Bayes, Support Vector Machines (SVM) and Logistic Regression (LR) as this type of algorithms require explicit class labeling for training the model for classification. The below figure illustrates the functioning of supervised binary classification,

Yes (Fake)

No (Real)

Fig.2. Functioning of Supervised Binary Classification

Data Labeling:

The SVM model is utilized to classify the real or fake news on twitter data. This learning algorithm analyzes the training data and produces an inferred function that can be used for mapping unknown data (testing data). So, an optimal scenario will allow for the algorithm to determine correctly the class labels for unseen instances like real tweets is labeled as 0 and fake tweets is labeled as 1.

a. SVM

The SVM technique is depends on the idea of decision planes. SVM is useful in handling when the feature set has large features. SVM is powerful when there is a sparse set of instances. Also SVM is strong when the problems are linearly divisible. SVM is useful in sentiment analysis.

Mathematically SVM draws hyper plane in 'n' dimensional vector spaces. SVM uses co-ordinates (x, y), where 'x' is the feature 'y' is the class. Hyper plane is defined as

$$b.x + b0 = 0$$

In the below mentioned formula we find maximal margin of hyper plane.

$$f(x^*) = b.\ x^* + b$$

'b' and 'b0' are the values used to find maximal margin hyper plane

To create maximal margin hyper plane each data point must be on the correct side, and a significant distance from it. In soft margin, the data point is on the incorrect side of the margin. The parameters ∈ and C are used for the purpose of soft margin.

To calculate the maximum margin M is used.

$$p \qquad b^2 \qquad =1$$

and $j=1$

$$y_i(b.\ x + b_0) \geq M\ (1-\epsilon_i),\ \forall i = 1\ldots\ n$$

where

$$\epsilon\ i \geq 0, \sum\ n\ i = 1\ \epsilon\ i \leq C$$

Parameter ‘C’ controls how much ‘i’ can be modified to create soft margin. The SVM classifier is designed using python for opinion mining and classification. Only the data in numerical format will be suitable in SVM.

This paper uses python tools for opinion mining and classification techniques. The text data is not suitable for SVM. The data is available in the numerical format for the extracted features. TF-IDF, N-Grams and proposed method converts a collection of twitter text to a normalized representation of feature for classification process.

4. Result and Discussion

4.1 Dataset Description

This section explains in details about how the experiment is conducted and also discusses the results of the algorithms utilized in this research for determining the best approach to detect twitter fake news in terms of Feature Extraction. The experiments are performed using real time twitter data sets. The data set consisted of approximately 5,800 tweets centered on Donald drump stories. The tweets are collected and processed in the works using python tool. The data set consisted of original tweets and they are labeled as fake and real.

The Input Features are described here: Id only used as a 18 digit number and which is unique to every tweet. Tweet contains the tweet’s text, Follower describes who has clicked the follow button and it includes three parts: read the tweets from accounts, interact with tweets-like, comment, and share, account allows send the messages directly. Following measure the accounts that are following other account. Actions represents total no of times user interacted with tweet, it includes retweets, replies, profile photo, hash tags, links, likes. Location denotes geographic location of the tweet’s poster.

Donald Drump related twitter data sets are fetched from twitter and performed different experiments on them to predict the fake or real from the user’s tweets. This section further explain how the details collected, how the features are chosen and how they are split for training and testing for fake news prediction task with machine learning classifier. At this point, it shows the results of various algorithms that are considered for this problem which effectively tunes their parameters to be suitable for this prediction task. For this research, Python Software is used in this research. Python is an advanced tool that consists of collection of library files.

4.2 Evaluation of Accuracy

The accuracy of prediction has been evaluated based on the different messages used for the training which are then optimized by glow worm swarm optimization algorithm in the proposed system. Following table shows the accuracy obtained at various feature selection methods for the SVM algorithm. The proposed technique is evaluated with other techniques by measuring the following evaluation parameters.

□ Accuracy: It is the percentage of test set tuples that are correctly classified by the classifier.

$$\text{Accuracy} = \frac{(\text{TP}+\text{TN})}{(\text{P}+\text{N})}$$

□ Precision: It is the ratio of predicted positive examples which are really positive

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

□ Recall or sensitivity: it measures how much a classifier can recognize positive examples

$$\text{Recall} = \frac{\text{TP}}{(\text{TP}+\text{FN})} = \frac{\text{TP}}{\text{P}}$$

□ F1-Score: It is to combine precision and recall into a single measure.

$$\text{F} = \frac{(2\times\text{Precision}\times\text{Recall})}{(\text{Precision}+\text{Recall})}$$

From the below table and figures, it is observed that proposed feature extraction technique have highest values when compared with TF-IDF and N-Grams algorithms for the SVM classification. The precision, recall and F1-Score obtained by different techniques are shown in below figures respectively.

Table 1. Comparison of Proposed Model with SVM Algorithms using Accuracy value

S.No	**Algorithms**	**Accuracy Obtained**
1	TF-IDF	84%
2	N-Grams	87%
3	Proposed	90%

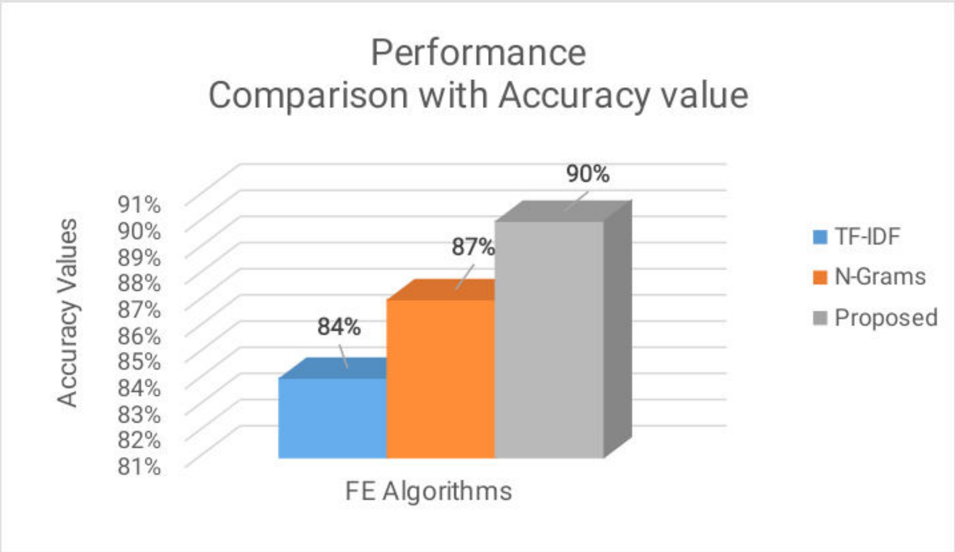

Fig.3. Comparison of Proposed Model with SVM Algorithms using Accuracy value

The accuracy of prediction has been evaluated based on the different messages used for the training which are then extracted by various FE algorithms in this research system.

Table 2. Performance Evaluation

	TF-IDF	**N-Grams**	**Proposed**
Precision	0.83	0.87	0.90
Recall	0.84	0.87	0.90
F1-Score	0.83	0.87	0.90

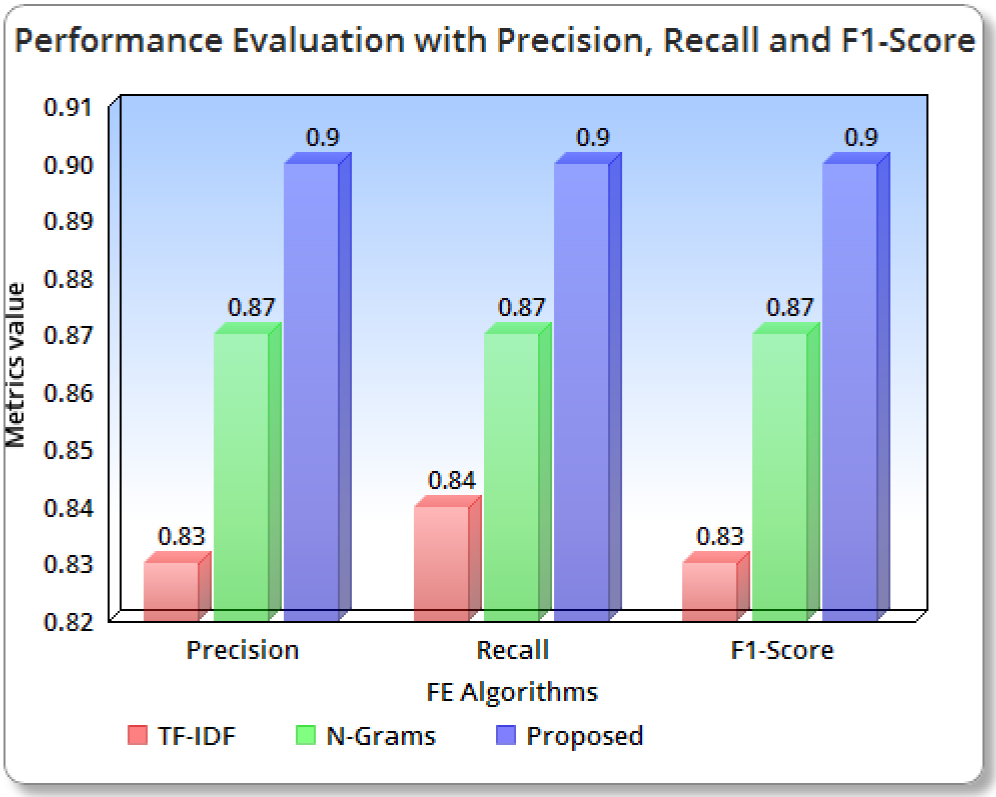

Fig.4. Performance Evaluation with Precision, Recall andF1-Score

The above results illustrates that the proposed technique has best performance when compared to other FE algorithms.

8. Conclusion

The proposed Feature Extraction algorithm has been designed to get features from twitter data for the detection of fake and real twitter data set. The Precision, Recall and F1 score for above algorithm proves best outcome. As indicated by the outcomes this research exhibits the possibility to meet a definite objective to automatically identify fake news from twitter data. The performance of TF-IDF and N-Grams feature extraction algorithms does not meet the accuracy of the algorithm proposed. In a more extensive study it is possible to clarify and achieve the information. In further a novel machine learning algorithm can be developed to reduce misclassification rate.

Reference

1. Lei Zhang, Bing Liu "Sentiment Analysis and Opinion Mining "Encyclopedia of Machine Learning and Data Mining , Springer Science , Business Media New York, 2017.

2. Atul Negi, Atul Negi" A Review of AI and ML Applications for Computing Systems" 2019 9th International Conference on Emerging Trends in Engineering and Technology-Signal and Information Processing (ICETET-SIP-19), IEEE.

3. Jhonathan de Godoi Brandão; Wesley Pacheco Calixto "N-Gram and TF-IDF for Feature Extraction on Opinion Mining of Tweets with SVM Classifier" 2019 International Artificial Intelligence and Data Processing Symposium (IDAP), IEEE.

4. Z.Jin, J.Cao, Y.Zhang, J.Zhou ,and Q.Tian , "Novel visual and statistical image features for micro blogs news verification, "IEEE transactions on multimedia, vol. 19, no.3, 2016, pp.598–608.

5. Y. Liu and Y.-F. B. Wu, "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, 2018.

6. Mykhailo Granik, Volodymyr Mesyura "Fake news detection using naive Bayes classifier "2017, IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON).

7. Anjali Jain, Avinash Shakya, Harsh Khatter , Amit Kumar Gupta" A smart System for Fake News Detection Using Machine Learning" 2019, IEEE 2nd International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT).

8. W. Yaqing et al, "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. ACM, 2018, pp.849–857.

9. M.Hardalov, I.Koychev, and P.Nakov, "In search of credible news, " in International Conference on Artificial Intelligence: Methodology, Systems, and Applications. Springer, 2016,pp.172–180.

10. J.Ma, W.Gao, Z.Wei, Y.Lu, and K.-F.Wong, "Detect rumors using time series of social context information on micro blogging websites, "in Proceedings of the 24th ACM International on Conference on Information and Knowledge Management.ACM, 2015, pp.1751–1754.

11. Neetu Anand, Dhruvi Goyal and Tapas Kumar "Analyzing and Preprocessing the Twitter Data for Opinion Mining" Springer Nature Singapore Pte Ltd. 2018 B. Tiwarietal. (eds.), Proceedings of International Conference on Recent Advancement on Computer and Communication, Lecture Notes in Networks and Systems34.

12. R.Ahuja, A.Chug, S.Kohli, S.Gupta, and P.Ahuja, "The Impact of Features Extraction on the Sentiment Analysis." Procedia Computer Science 152, 2019, pp.341-348.

13. K L Santhosh Kumar, Jayanti Desai, Jharna Majumdar" Opinion mining and sentiment analysis on online customer review" 2016, IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), IEEE, 2473-943X.

14. Gleen A. Dalaorao; Ariel M. Sison; Ruji P.Medina "Integrating Collocation as TF- IDF Enhancement to Improve Classification Accuracy" 2019 IEEE 13th International

Conference on Tele communication Systems, Services, and Applications (TSSA)

15. Jhonathan de Godoi Brandão; Wesley Pacheco Calixto "N-Gram and TF-IDF for Feature Extraction on Opinion Mining of Tweets with SVM Classifier" 2019 International Artificial Intelligence and Data Processing Symposium (IDAP), IEEE.

16. P.Yang and Y.Chen, "A survey on sentiment analysis by using machine learning methods, " in 2017 IEEE 2nd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), pp.117-121, 2017.

17. Z.Jianqiang, "Combing Semantic and Prior Polarity Features for Boosting Twitter Sentiment Analysis Using Ensemble Learning," in 2016 IEEE First International Conference on Data Science in Cyber space (DSC), pp. 709-714,2016.

b_0