Real Time Detection of Misinformation on Twitter using Ensemble Learning and Natural Language Processing

Dr. Shobha Tyagi1, Dr. Balamurgan Balaswamy2,

Dr. Pronika Chawla3, Dr. Neha Tyagi4, Dr. Pinki Sagar5

tyagishobha.fet@mriu.edu.in1, kadavulai@gmail.com2, pronika.fet@ mriu.edu.in 3

ntyagi2@amity.edu4, pinkisagar9@gmail.com**5**

# ABSTRACT

Online Social Media (OSM) has been playing an inherent part in influencing lives of internet users directly or indirectly. Social Media has penetrated its roots deep down in channelizing and voicing our opinion and thoughts. On one side it acts as a great platform to exchange ideas, populating news and sharing interests and could unite people for a noble cause while on the other hand it could index disasters by spreading wrong or fake information whose impact even could be detrimental and dangerous particularly if it has some political or heinous backgrounds like terrorism, attacks, kidnapping, hijacking, war etc. There is an emergent need to deal with the authenticity of the data/news before spreading it or decision making and thus consequently to avert unacceptable and ill impacts of fake news on human beings and ecosystem itself.

The authors of the paper envisages novelty by etching ensemble learning mode of topic modeling in proposed fake news detection model in real time. The model is weaved and encapsulated around the concept of context and semantic analysis of data to fetch detailed insights and context. The proposed model centered around advanced machine learning and NLP techniques, namely TF-IDF and Topic modeling to analyze a significant amount of information from multiple sources and recognize regular expression that may reflect the spread of fake news based on parameters like accuracy, precision and is compared and evaluated with other techniques namely Naïve Bayes, Decision Tree, Random Forest. Results of proposed model, clearly indicates that proposed model of ensemble learning outperformed the other machine learning techniques for all the 3 data sets cited.

Keywords— Information retrieval, Knowledge representation, Semantic analysis, Text classification, Natural Language Processing, Fact-checking, Verification, Credibility, Authenticity, Source evaluation

# =======================================================================

1. **INTRODUCTION**

Suspicious users find Twitter, the most popular micro-blogging website among OSMs, an easy target to disseminate rumors with illicit and wrong intentions to unroll and deliberately air the misleading facts to all types of crowds spanning masses to prolific personalities and public figures of great stature and big commercial enterprises. Twitter is a micro blogging online social media where the users post short text or multimedia notes or communication called tweets. The characteristic and speciality of these messages is that they are restricted to a maximum length of characters to 280, which later announced and rebirthed 140 characters. There are some terminologies that are specifically used in context of Twitter.

The two primary terms are "follower" and "followee." To receive messages from another user, one must become a "follower" of that user, while the user being followed is referred to as the "followee". Besides there are certain actions that users can perform on tweets of some other user. These include 'favourite', 'reply' , 'retweet' and 'mention'. Users who like a particular post can 'favourite' the tweet. Users can express their opinions on the post by commenting on it through the 'reply' option. A retweet may be considered as a repost or forward of the same tweet by another user. A mention is used when a user wishes to specifically reply to a particular user.

Therefore, it is evident why Twitter has gained popularity as a platform for enabling the generation and dissemination of information, ideas, professional interests, and various types of expression through virtual communities and networks.

It is due to this typical ‘follower-followee’ relationship, that information flow on this social media is relatively much faster as compared to other social media. This also means that misinformation can spread relatively faster on Twitter. Since Twitter is not just used an online social media but also as a popular source of news by many people, the problem of preventing the misinformation spread becomes even more important on Twitter than on other social media. Our system provides a solution to this social issue by building a Google Chrome extension that informs users about the possible credibility of each tweet using Data mining, Natural Language Processing and Ensemble Learning, in real time on the official Twitter website itself. As per the researchers comprehend, most of the prior research paper works related to this topic had a relatively limited scope, since they worked only on some specific events and detection of misinformation was not real time on the official Twitter Website.

The rest of this paper is structured as follows: Section II covers the literature reviewed for this work, and Section III outlines the problem definition while Section IV describes the proposed system methodology along with the block diagram to indicate the workflow as well as detailed explanation of the tools and algorithms used by the system. Section V provides the details of implementation, the performance measures, statistical data for visualisation along with the results. Section VII provides the conclusion of the research paper and outlines potential areas for future work.

1. **LITERATURE REVIEW**

# The research paper by Shu et al. (2022) [1] offers an extensive overview of methods for identifying and mitigating fake news. The authors examine multiple techniques, such as content-based, social context-based, and knowledge-based approaches along with their strengths and weaknesses. They also present a list of challenges and opportunities for future research in this area.

# This paper by Gilda et al. (2022), proposes a data mining-based approach for detecting fake news on social media. The authors use various features, such as linguistic, network, and user-based features, and apply machine learning techniques to classify news articles as fake or real. The experimental results show that their approach achieves high accuracy in detecting fake news. This paper by Horne et al. (2023) provides a survey of fake news detection tools, including both commercial and academic solutions. The authors discuss various techniques used by these tools, such as fact-checking, source verification, and sentiment analysis, and evaluate their strengths and weaknesses. They also identify opportunities for future research in this area. This paper by Xinyi et al. (2022) proposes a data mining-based approach for identifying fabricated news on public platforms.

# The authors utilize a range of features, such as linguistic, network, and user-based features, and employ machine learning methods to categorize news articles as authentic or fake. The results of the experiment demonstrate that their approach achieves exceptional accuracy in detecting fabricated news. Yang et al. (2022) present FAKENEWSNET, a data repository that includes news articles, social context, and spatiotemporal information to investigate fabricated news on social media.

# The authors describe the dataset and its features, and provide some examples of how it can be used for fake news detection research. They also discuss some of the limitations of the dataset and identify opportunities for future research. This paper by Dat Quoc Nguyen, et al. (2021) introduces a pre-trained language model specifically designed for sentiment analysis on English tweets. The model, called BERTweet, is based on the BERT architecture and is fine-tuned on a large dataset of labeled tweets. This survey paper by Rui Liu, et al. (2021) provides an overview of the state-of-the-art in emotion detection from multimodal communication, including text, speech, facial expressions, and physiological signals. It also discusses the challenges and future directions for this field. This paper by Hongliang Yu, et al. (2021) proposes a novel dual learning approach for multilingual sentiment analysis, which leverages unlabelled data from multiple languages to improve the performance of sentiment analysis models. The authors demonstrate the effectiveness of their approach on several benchmark datasets.

# Lei Zhang et al. (2021) present a thorough analysis of deep learning techniques for sentiment analysis, encompassing both supervised and unsupervised methods. The authors additionally examine the constraints and potential research avenues for this domain.This paper by Hao Wang, et al. (2021) proposes a generative model for sentimental text generation, called SentiGAN, which uses a mixture of adversarial networks to capture the sentiment and content of input texts. The authors validate the efficacy of their method on multiple benchmark datasets.

Shobha Tyagi et al. [11] proposed a machine learning solution for preventing misinformation propagation by providing credibility of a tweet on Twitter. The proposed system would take keywords or hashtags of an event from the user and use the Tweepy API to extract related information for the model. The tweet analysis would be done using python’s Natural Language Tool Kit (NLTK) module and the cleaned text and other details would be given to the machine learning model for processing. The system uses the supervised machine learning algorithms of Naïve Bayes and Decision Tree for

classification. The output of the classifier is any one credibility class out of Fake, Seems Fake, Seems Credible and Highly Credible. Confusion matrix is used as the performance evaluation metric for the classifier. The authors propose that the same notion can be expanded to other online social media platforms.

According to Sivasangari V et al. [12], news tends to surface initially on microblogs such as Twitter before appearing in traditional media outlets. The authors build a rumor identification framework for recognizing if data is verified or not on Twitter. Twitter Streaming API was used to fetch Tweet details. Dataset was created using PHEME dataset. The approach followed was to collect tweets related to specific events of PHEME dataset using Twitter Streaming API, perform manual annotation of tweets and obtain a label, and finally use the machine learning classification algorithm for prediction of label. To improve the accuracy, the authors have checked tweets with real news channel to find the original news and fake news. Tweets were classified as Rumour and Non rumour. The Naïve Bayes classification algorithm was used and has provided the authors with around 90% accuracy. In their research paper, Samah M. Alzanin et al. [13] reviewed the various techniques employed to identify rumors on social media, including supervised, unsupervised, and hybrid approaches.

The authors observed that most of the approaches used to deal with the problem of rumors was based on Supervised Learning approaches. However, Unsupervised Learning approaches like Clustering were used as well. Both approaches had some advantages and some drawbacks as well. Nonetheless, the authors noted that certain investigations merge supervised and unsupervised learning in their analysis, which they refer to as a hybrid approach. The research paper concludes that due to the extensive and swift dissemination of information and the lack of measures to guarantee its credibility, multiple studies have emerged to devise systems for identifying rumors. Nevertheless, there remains an imperative requirement to widen the scope of inquiry.

Raveena Dayani and colleagues [14] conducted an analysis of Twitter data by considering various parameters such as the date of tweet posting, user ID, tweet content, tweet label, tweet ID, and other related features. They collected the Twitter data using the Twitter Search API and stored it in a MySQL database. The authors applied their own pre-processing algorithm before using classification algorithms, such as K-Nearest Neighbour (k-NN) and Naïve Bayes. For k-NN, they calculated the Euclidean distance over user-based features but achieved only 73.8% prediction accuracy for endorses and as low as 40.9% for denies. The authors attribute the low accuracy of k-NN to the lack of correlation between user-based features and rumour detection. They applied Naïve Bayes' algorithm based on word frequencies present in the tweets and considered two categories of factors: user-based factors, such as the user's account creation time, the number of followers and followees, the number of tweets posted by the user, and the total number of favorites obtained for each user; and content-based factors. The Naïve Bayes method achieved higher prediction accuracy than k-NN.

Sardar Hamidian et al. [15] utilized three datasets, namely Obama, Palin, and MIX (mixed dataset

consisting of data from five selected rumors) to achieve better prediction accuracy. They considered various parameters such as lexical features, unigrams, bigrams, parts of speech, sentiments, emoticons, replies, retweets, user ID, hashtags, and the time of the tweet. Different levels of pre-processing such as lemmatization, removal of punctuation, lowercase conversion, and stop-word removal were applied to the tweet content. The authors observed that pre-processing did not improve accuracy significantly, but may result in loss of valuable information. Classification was performed using the J48 Decision Tree algorithm and the WEKA platform for training and testing.

Qiao Zhang et al. [16] proposed a binary classification method for rumour detection using a combination of shallow and implicit features. Shallow features are extracted from basic user or content attributes, while implicit features are generated by mining deep information from user or content. Their proposed system consisted of three steps: data cleaning, feature extraction, and training the model. Content-based implicit features considered included popularity orientation, sentiment polarity, opinion of comments, and internal-external consistency, which provides interrelation between message content and the corresponding external page content. User-based implicit features such as social influence, opinion re-tweet influence, and match degree of messages were also considered. The authors used Support Vector Machine (SVM) for classification and concluded that user credibility is an important factor that directly or indirectly impacts the credibility of information.

Aditi Gupta et al. [17] collected Twitter data related to two events, the Boston Marathon Blasts (2013) and the Hurricane Sandy (2012), using the Twitter Streaming API. They analyzed the temporal distribution of tweets by considering the number of tweets posted in each hour after the event occurred. The authors discovered that false information spreads faster than true information and is more prevalent at the start of the event. In addition to analyzing posted tweets, they also analyzed tweets from suspended users. User-based features such as the number of followers and following were also taken into account. The authors used Naïve Bayes and Decision Tree algorithms for classification, and found that Decision Tree provided higher prediction accuracy of 96.65%, compared to Naïve Bayes which achieved a prediction accuracy of 91.52%.

Aditi Gupta and her team, in their study [18], specifically investigated the role of Twitter in disseminating fake images during Hurricane Sandy (2012). The authors conducted a characterization analysis to examine temporal, social reputation, and influence patterns for the spread of fake images. They discovered that only a small percentage of tweets (around 14%) containing fake images were original tweets, while the vast majority (86%) were retweets. The authors used a decision tree classifier to achieve the best results in identifying real images from fake ones. They concluded that automated techniques can be employed to detect fake content posted on Twitter during real-world events. The authors also expressed their interest in expanding their study to detect rumors and other malicious content spread via images and other media during such events.

Vahed Qazvinian and colleagues [19] conducted their analysis in two main steps. Firstly, they extracted tweets related to the controversial aspects of the story using Twitter Search API, and secondly, they identified users who believe the misinformation versus those who refuse or question the rumor (Belief classification). Over 10,400 tweets were annotated, and to calculate the annotation accuracy, 500 annotations were annotated twice, and the comparison was done using the Kappa coefficient. In the first task, the aim was to classify tweets as rumor or non-rumor, while in the second task, the marked rumorous tweets were used to identify users who trust the rumor versus those who deny or question it. The first task involved building various Bayes classifiers for high-level features and then learning a linear function of the classifiers. Content-based, Network-based, and Twitter-specific features were considered, such as lexical patterns, parts of speech, unigrams and bigrams. Moreover, hashtags, which are a vital terminology in Twitter, were analyzed to understand whether those used in rumour-related tweets were similar or different from the other tweets. For the Belief classification task, the Bayes classifier was employed again.

# The data for analysis in the study conducted by Carlos Castillo et al. [20] was collected through the use of a Twitter monitor, which detects sudden spikes in the frequency of a set of keywords in bursts of messages. The tweets were then classified as either news or chats. The study considered various features such as message-based features (e.g. length, symbols, hashtags, re-tweets), user-based features (e.g. registration details, age, number of followers, number of tweets), topic-based features (e.g. sentiments of tweets, URLs), and propagation-based features (e.g. building a propagation tree from the re-tweets of the message). The classification algorithms used in the study were SVM, Decision Tree, and Bayes network, with Decision Tree producing the best results.

# The spread of misinformation, whether intentional or accidental, through social media, especially during sensitive real-world events, can have harmful effects on individuals and society as a whole. Twitter, with its diverse user base including the general public, celebrities, politicians, and organizations, is a popular target for malicious users to spread fake news. The authors identified two main issues related to the spread of misinformation on social media. The first issue is that fake content spreads faster than real content, especially during the initial stages of an event. This leads to people believing and sharing the news without verifying its authenticity. The second issue is the absence of central moderation on social media due to the use of crowd-sourcing, allowing anyone registered on the platform to post any content from their account. This makes it challenging to control the spread of misinformation and trace the source of the information.

1. **METHODOLOGY & PROPOSED SOULTION**

The proposed model is an extended piece of our machine learning paradigm which sufficiently streamlines to expose and unmask such misguiding bits of news on Twitter and identify potential steps that social media companies could take to curb the dissemination of false information, as well as actions that users can adopt to avoid spreading content without verifying its accuracy and truthfulness. Classifying content as real or fake can involve using various techniques and methods to categorize articles into different groups based on their topics, themes, or other characteristics.

Here are some related terms in news content to classify data, as per proposed model:

* Content taxonomy: A hierarchical classification system that categorizes news articles into different topics or subjects, such as politics, sports, or entertainment.
* Content analysis: A research method used to systematically analyze the content of news articles and identify patterns or themes.
* Clustering: A technique used to group similar news articles together based on their content, using algorithms such as k-means clustering or hierarchical clustering.
* Topic modeling: A statistical technique used to identify topics or themes that occur frequently in a set of news articles.
* Keyword extraction: A method for automatically identifying and extracting the most important keywords or phrases in a news article.
* Named entity recognition (NER): A process of identifying and extracting entities such as people, organizations, and locations from news articles, which can be used to classify articles based on the entities they mention.
* Sentiment analysis: A method for determining the emotional tone of a news article, such as positive, negative, or neutral, which can be used to classify articles based on their sentiment.
* Machine learning: A type of artificial intelligence that can be trained to automatically classify news articles based on patterns or features in the data.
* Natural language processing (NLP): A field of study concerned with understanding and analyzing human language, which can be used to classify news articles based on their linguistic features.
* Classification algorithms: Methods like decision trees, random forests, or support vector machines can be employed to automatically categorize news articles based on their features or attributes.
* The proposed model encapsulates that the entire presented document or content which is to be classified as truth and fake as represented in eqn. (1):-

A= { x| x ε (Y Ս Z): Y ε {a document whose data to be to be ascertained as truth or fake }; Z ε {set of all the documents in the repository may be containing fake and true data}} ------(1)

To classify the document as containing real or fake data, a hypothetical function is characterized as

Cɸ(A)= ------(2)

Equation (2) describes the hyper tuning of parameters, and thus helps in classifying the presented document as containing fake or true elements.

The eqn (2) is optimized further in terms of its conditional probability with respect to its cost and complexity function based on its sentiments and contextual embeddings, using logistic regression as

-------(3)

In order to prove the importance of the words in a document researchers proposed the concept of fusion of Term Frequency – Inverse Document Frequency (TF-IDF) method and Probabilistic topic modelling based on sentiment analysis, feature extractions and contextual embeddings of the presented document.

Term Frequency - Inverse Document Frequency (TF-IDF) is a highly popular approach for fetching and retrieving information from a corpus of records and documents. The two most prevalent and interrelated concepts associated with this technique are Term Frequency and Inverse Document Frequency.

Hence term frequency is evident from equation (4), and suggests the frequency in which the word appear in current document to that of the frequency of word appearance in collection of documents in the corpus.

These words in a document are assigned scores which indicate their significance or importance in deducing the meaning or context of the sentence through a vectorization process.

Essentially, IDF represents the proportion of the total number of documents in the corpus to the number of documents in the corpus that contain a particular term, and hence can be empirically stated as eqn. (5)

In order to optimize the eqn (6) computation and to avoid some logical errors, it can be rewritten as

In this entire process a word obtains its importance score, in the form of TF-IDF, by multiplying the scores of TF (from eqn (5)) and Inverse Document Frequency (IDF) (from eqn (6):-

Higher is the value of TF-IDF, greater the significance of the term within the document or across the collection of documents and thus balances the commonality in document and rarity between the documents in the corpus.

In the proposed model, eqn (7) will primarily be used to cluster the text based on its relevance or based on its similarity index. To detect the fakeness in the data Benford’s law has been very prevalent in the natural sciences and behavioral sciences. This theory accentuates on the fact that distribution of initial digits determine and dictates a trend in the content and is represented as

Where d represents the non-zero digits starting from 1,2,…9. The concept highlights the fact that lower numbers occur with more probability in data as compared to higher digits.

To have good Benford’s law predictions a fitness model is created, to closely grab the movement of data instances and determine their statistical directions

ɸ = 100 ------- (9)

where nd stands to data instance taken into consideration and pd stands for the probability of data with initial digits. The Naïve Bayes algorithm employs the principles of probabilistic Bayes' theorem as a machine learning technique. It belongs to a group of probabilistic algorithms that leverage probability theory and Bayes' theorem to classify samples. The equation (10) can express Bayes' theorem, which is the foundation of this algorithm, making it both simple and powerful.

Where A stands for document to be tested and B is repository of documents which are already tested and authenticated, and helps in getting the model trained and thus helps in increasing the efficiency of the proposed model to detect the document as real or fake. Since words are the reflections of the document, hence the orientation of the document in terms of its words referral can be shaped empirically as:-

Where n*domain* = No. of documents related to a particular domain or topic and N*doc\_of all*\_*domains* = Total no. of documents of all domains in the repository.

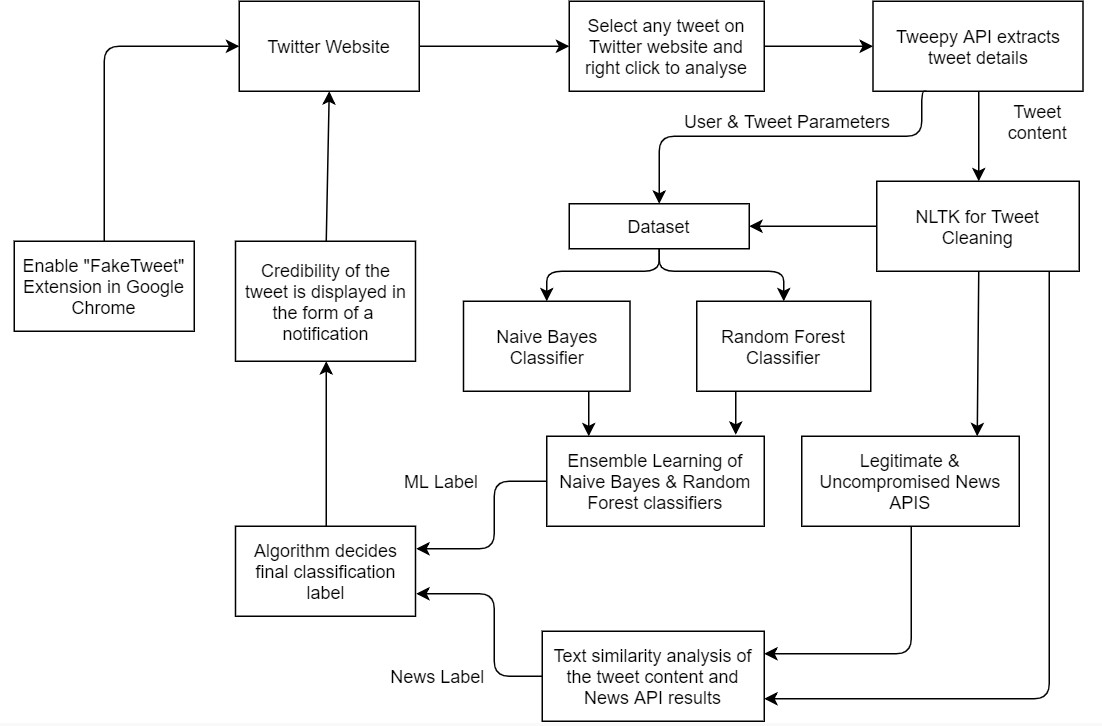
To simplify the suggested model's complexity, only unique words and its associated domains are studied with great detail and deep concentration. Hence to understand their impact on the context and its orientation towards truth or fakeness the probability is further refined as:-

Where w stands for word which might be checked with a topic*j* and UCi denotes the uniqueness and act as

the modulator in deciding the news data as real or fake. To determine the importance of a word or to declare the word as a keyword in a context or situation in a paragraph (p) to decide its orientation as fake or real, equation (13) is illustrated as

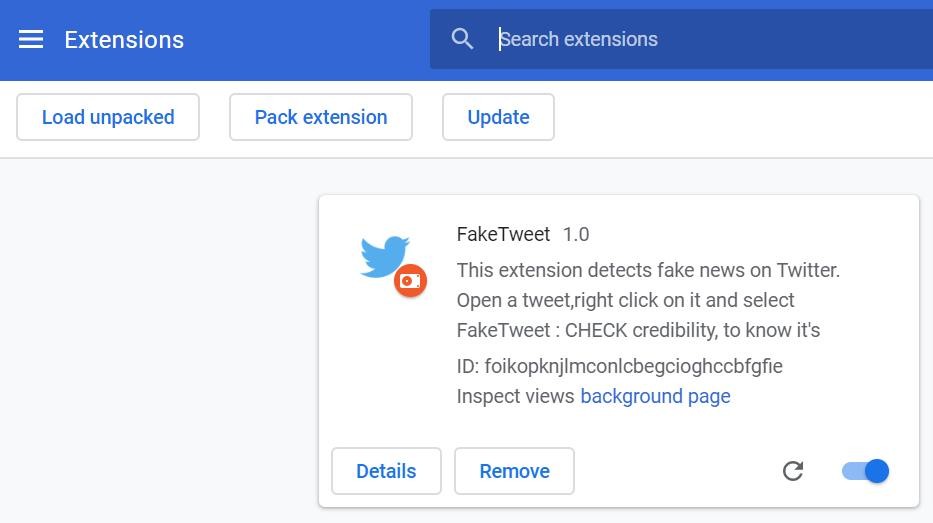
Swp denotes credibility of word in a paragraph and n denotes the no. of paragraphs in document. The system is a Google chrome extension to predict credibility of a tweet using the concept of Ensemble Learning which is a combination of many machine learning algorithms. The user has to visit the official Twitter website, open any tweet, whose credibility is to be found and right click on it to analyse. The system will work on the tweet using Tweepy API, NLTK and Ensemble Learning algorithms to predict a class for the tweet. The tweet will be classified in any of the four classes - Fake, Seems fake, Seems Credible & Highly Credible and the same would be notified to the user. The block diagram of the proposed model is shown in as shown in figure 1.

Extensions are compact software applications that personalize the browsing experience by allowing users to customize browser features and behavior to their specific preferences. These components are developed using web development tools such as HTML, CSS, and JavaScript. Google Chrome provides a range of extensions that enable developers to extend the functionality of the browser. Each extension comprises different yet coherent components, including background and content scripts, an options page, UI elements, and logic files. Depending on the extension's intended purpose, not all components may be necessary.



**Fig.1** Block diagram to indicate the workflow for real time detection of misinformation on Twitter using Ensemble Learning and Natural Language Processing.

Extensions can take various shapes of a user interaction, but the one in our project uses a notification message. The extension will act as the user interface to our system and will take the selected tweet as input and pass it to the system as well as display the classification output obtained from the system.



**Fig.2** Google Chrome Extension for real time detection of misinformation on Twitter.

Since our Google chrome extension is local to our machine, we need a local server on our machine to communicate between the extension and the model as shown in figure 2. This is what the python’s Flask will help us to achieve. Flask, a Python-based micro web framework, does not come with a database abstraction layer, form validation, or other pre-existing components. Instead, these common functions are typically provided by third-party libraries. By storing the data sent by the extension, the web server will be capable of responding to user input with dynamic content that is then relayed to the system.

Twitter provides Application Package Interfaces (APIs) to enable developers to integrate tweet analysis with their programs. Most programming languages offer a wrapper for this API and Tweepy for Python is one of them, which is being used in research proposal. Tweepy, the official Python API for Twitter, is a user-friendly library that employs the REST 1.1 API and supports both OAuth and Streaming API. Tweepy can obtain user and tweet characteristics, which can then be used to analyze data with a machine learning model.

The News API is a straightforward HTTP REST API that enables users to search and retrieve live news articles from the internet. This API can fetch current top stories from any news website or locate top news related to specific topics or keywords, allowing users to retrieve news based on specific criteria. GNews is another such API for extracting relevant news form given keywords.

The tweet content will be analysed with respect to the results of both the News API to get a quantitative analysis of the support of uncompromised sources in favour of the tweet. To understand the support provided by News API results in favour of the tweet, we perform Text similarity using the popular TF- IDF algorithm.TF-FIDF, i.e. Term Frequency-Inverse Document Frequency (TF-IDF) is a numerical metric designed to measure a word's importance to a document in a corpus or collection. It is among the most widely used term-weighting techniques and is frequently employed as a weighting factor in information retrieval and text mining searches. The approach involves two terms: Term Frequency (TF) and Inverse Document Frequency (IDF), with each word having its corresponding TF and IDF score. The product of a term's TF and IDF scores produces the TF-IDF weight. The rarer the term, the higher the TF-IDF score or weight, and vice versa.

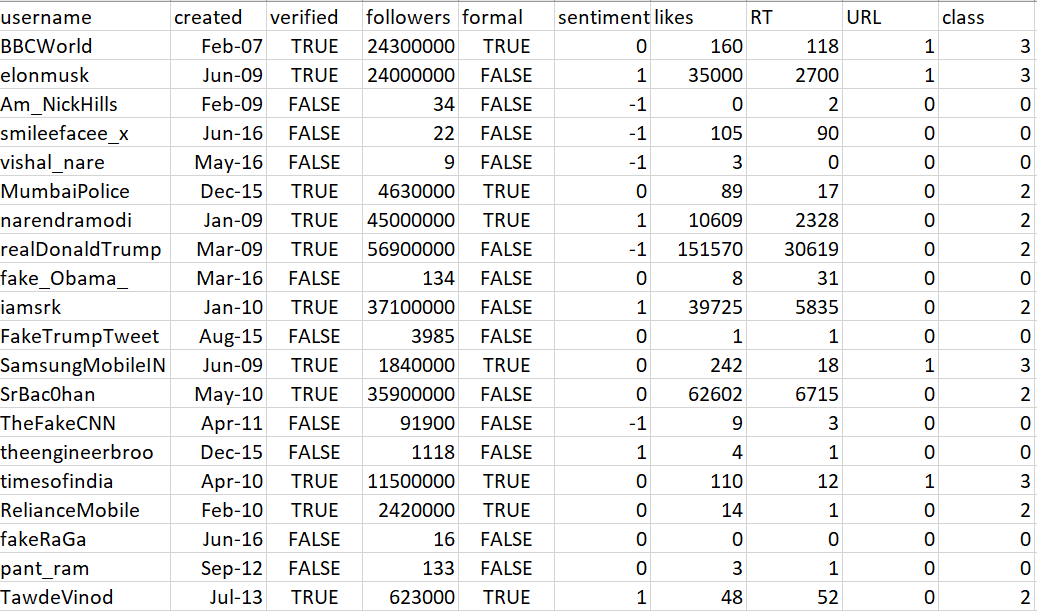
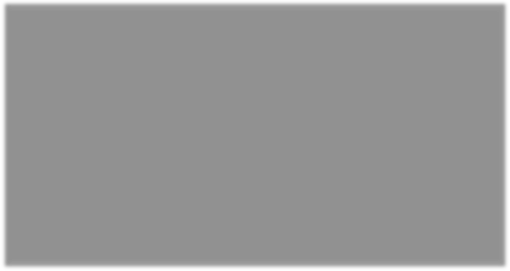
The frequency of a word in a document, or term frequency (TF), is calculated as the number of times the word appears in the document divided by the total number of words in the document. For example, if the word 'student' appears 16 times in a 100-word document, then the TF for ' student ' is 16/100 = 0.16. In contrast, the inverse document frequency (IDF) of a word measures the significance of the term in the entire corpus. For instance, if the term ' student ' appears 'x' times in a corpus of 10,000,000 documents and there are 0.3 million documents containing the term, the IDF (i.e. log {DF}) is calculated as the total number of documents (10,000,000) divided by the number of documents containing the term 'student' (300,000). IDF will be calculated as log (10,000,000/300,000) which is 1.52. Now the TF- IDF score of 'dog' is calculated as 0.16 \* 1.52 = 0.2432. TF-IDF score will be used to estimate the similarity of the selected tweet text with the results of both the News APIs each. This provides a classification label according to the News API.

Tweets are composed of Natural Language, which is the way humans communicate with each other. However, Natural Language is often difficult for computers to interpret, as it can be messy and full of nuances. Natural Language Processing (NLP) is a field of Artificial Intelligence that aims to enable computers to understand, interpret, and manipulate human language. Python provides a valuable module for NLP called Natural Language Tool Kit (NLTK), which offers a range of functionalities such as splitting paragraphs into sentences, splitting words, identifying the part of speech of those words, highlighting important subjects, and aiding in the machine's understanding of the text. This will be embedded to analyse the tweet content in detail.

Machine Learning is a field of computer science that enables computers to learn from experience without being explicitly programmed. This learning can be categorized into three types: Supervised learning, Unsupervised learning and Reinforcement Learning. The proposed system is a supervised machine learning model which means the model will first learn through examples provided through a dataset and then predict an outcome with a better accuracy. Supervised machine learning algorithms need a dataset to understand the classification outcomes based on input features. Data set is taken from Kagge.com (https://www.kaggle.com/datasets/elvinagammed/covid19-fake-news-dataset-nlp?select=Constraint\_Test.csv and https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews and http://ai.stanford.edu/~amaas/data/sentiment/)

A database has primarily many features & each feature may or may not contribute to the final classification outcome. As an example, the dataset used in our system contains both user-based and tweet-based features. User-based features comprise the username, the date and time the account was created, whether the user is verified, and the number of followers. Meanwhile, tweet-based features consist of the formality of the tweet's content, the sentiment expressed, and the number of likes and

retweets garnered and whether the tweet contains a valid URL about the event mentioned to support the user’s assertion. Since there was no such dataset available that involves all these parameters, we created a mini chrome extension and python code for creating our own customized dataset. This automation also sped up the operation of dataset creation. The dataset being prepared, the next step is to train the classifier models on the dataset and predict the results on unknown samples, as shown in figure 3.



**Fig.3** Snapshot of the dataset used by the system’s machine learning model.

There are various supervised machine learning algorithms for classification like Logistic regression, K- nearest neighbors, Support vector machine, Naïve Bayes, Decision Tree, etc. However, there is no ‘best classification algorithm’ which will work great for all problems. An algorithm that works great on one problem may or may not work good on another. Hence to select algorithm for our system, we implemented all the above mentioned algorithms on our dataset and chose the ones that were not only providing better accuracy but were also less affected by changes in the dataset. We selected the machine learning algorithms of Naïve Bayes and Decision Tree for our classification problem.

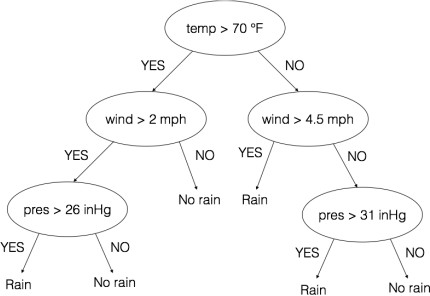
Naive Bayes classifier calculates the probability of each category for a given sample and selects the category with the highest probability. There are several advantages of using this classifier. It can be used for both binary and multiclass classification problems, is highly scalable, and can make probabilistic predictions that are less sensitive to changes in data. Additionally, it can handle both continuous and discrete data, is not affected by irrelevant features, and requires less training data. When the conditional independence assumption holds, it converges faster than discriminative models such as logistic regression. However, in practice, it often works well even when the independence assumption is violated. To understand the working of Naïve Bayes, consider for instance that a Tweet has been identified with following features after data mining –

X = {Verified = False, Followers = 80, Formal = False, Sentiment = 0, Likes = 8, Retweets = 1, URL = 0 } Naïve Bayes finds the probability of tweet for falling in each class. Say the probabilities obtained are - P ( Fake / X ) = 0.68 , P ( Seems Fake / X ) = 0.21,

P ( Seems Credible / X ) = 0.08 P ( Highly Credible / X) =0.03

Clearly the probability of tweet being fake is more than the other three classes and hence there are maximum chances that the tweet is fake. This is intuitive as well. An unverified user with less followers posting an informal content without any URL and that hasn’t got much response from other users is most likely to be an irrelevant or fake information.

The second algorithm that our system would be using is the Decision Tree. Decision tree is a tool used in decision analysis that visually and explicitly represents decision-making. It constructs decision rules for prediction using a tree-like model of decisions. The decision tree is typically drawn with its root at the top and is inverted in orientation. In simple terms the decision tree algorithm works by analysing a dataset to obtain the feature that best splits the dataset for classification. This forms the root of the tree and the process is recursively repeated till all the samples of the dataset fall under some rule. From the tree, the algorithm constructs a set of if-then rules. Hence once the tree is constructed, any input sample will just be tested against these rules and a class label will be assigned accordingly. Decision trees are more intuitive as well as work good for non-linear data as well. Besides, the data preparation required is minimum. Most of the times, data pre-processing is not even needed. Among the various significance of Decision tree, the very prominent benefit is that it eliminates redundant features as well which might be responsible for reducing the accuracy For instance after a tree is created, the algorithms might construct a rule that ‘If the user is not verified and number of followers is less than some threshold value ‘x’ and sentiment is negative and no legitimate URL is provided then the tweet is more likely to be fake.



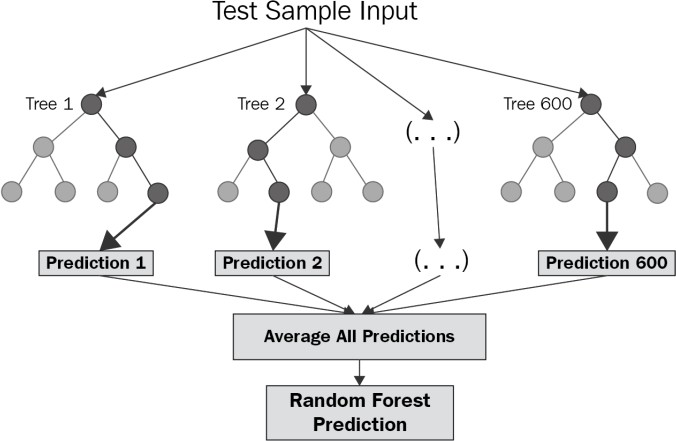
**Fig. 4** An example of a decision tree constructed by some decision tree classifier to predict the possibility of rain based on temperature, wind speed and pressure. [12]

Although Naïve Bayes and Decision Tree algorithm provide good results, the problem of fake news

being sensitive in real time situation, demands as much accuracy as possible. Although we understand

that the accuracy for real time problems like this, would be limited, we came up with the concept of Ensemble Learning to boost the efficiency, as shown in figure 4. Ensemble learning is a machine learning technique that involves creating multiple models and then combining them to produce better results than a single model. The use of ensemble learning can often lead to more accurate solutions. An example of an ensemble learning algorithm is Random Forest, which is composed of multiple decision trees. The algorithm randomly selects a subset of k-data points from the training set and constructs a decision tree based on these points.

This process is repeated for a selected number of trees which means there will be n-decision trees created, each working on random data points forming a ‘forest’ of n-trees, hence the name Random Forest. For each new data point, each of the n-trees will predict an outcome class. The data point will be assigned to the class that has received majority of the votes. For instance, if we decided to create 100 decision trees and if 84 out of 100 trees vote for class ‘A’ and the rest vote for class ‘B’ then the outcome will be class ’A’.



**Fig. 5** An example of Random forest classifier as an ensemble learning of multiple Decision tree classifiers [13]

However, Ensemble learning doesn’t always means combining the votes of same type of classifiers. In or system we are implementing an ensemble learning of Naive Bayes and Random Forest (which itself is an ensemble learning of Decision trees), which has helped boost accuracy of the model. Hence the voting of both Naïve Bayes and Random forest is considered to decide the final class, as shown in figure 5. Moreover, Voting need not always be on the majority basis. There are two types of voting in ensemble learning algorithms viz. Hard voting and Soft voting.

Hard voting predictions are made directly on voting basis whereas Soft voting predictions will be done based on probabilities. For instance, consider we have 3 classifiers and 2 classes ‘A’ and ‘B’ to predict. In hard voting, classifiers 1, 2 and 3 predict outcomes as classes ‘A’, ‘B’ and ‘B’ respectively. Since 2/3 classifiers predict class ‘B’ as the outcome, the ensemble learning decision is class ‘B’. In

case of soft voting, the approach will be probabilistic. Classifier 1 predicts that outcome is ‘A’ with probability of 99%, Classifier 2 predicts that outcome is ‘A’ with probability of 49% and Classifier 3 predicts that outcome is ‘A’ with probability of 49%, The average probability of the output class being class ‘A’ will be (99 + 49 + 49)/3 = 65.67%. Similarly, the average probability of the output class being class ‘B’ will be calculated and the class that has the maximum value will be the final ensemble learning decision. It is clear that soft voting can enhance hard voting as it considers more information by utilizing the uncertainty of each classifier for every class while making the final decision. Thus, the outcome of the ensemble learning model of our system will provide another classification label.

Now the researchers analyzed the outcome label of the machine learning model and the outcome label that was earlier obtained from the News API to obtain the final credibility of the tweet. This will be passed to the chrome extension and the output will be displayed to the user immediately in the form of a notification.

1. **RESULTS AND DISCUSSIONS**

As already mentioned, most of the prior research papers related to this topic had a relatively limited scope, since they worked only on some specific events and detection of misinformation was not real time on the official Twitter Website. In contrast to those, our system requires users to only enable the extension and then obtain credibility of any tweet in real time on the official Twitter website, as shown in figure 6 and figure 7. The system after implementation is found to be able to predict a suitable class for a selected tweet based on its’ credibility (Fake, Seems Fake, Seems Credible or Highly credible) in real time.

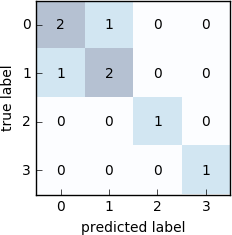


**Fig. 6** Select the Tweet whose credibility is to be determined and select FakeTweet option on right click



**Fig. 7:** Credibility of the tweet is informed to the user by a notification message

As our machine learning model deals with a classification problem, we utilize a confusion matrix to evaluate its performance. A confusion matrix provides a summary of the prediction outcomes on a specific classification problem. It displays the count values of correct and incorrect predictions for each class. By analyzing the confusion matrix, we can gain an understanding of how the classification model becomes confused while making predictions, hence the name. The advantage of confusion matrix is that it not only provides the error rate and accuracy, but also the types of errors being made. For instance, from a confusion matrix of 2 classes ‘X’ and ‘Y’, we can understand how many tuples belonging to class ‘X’ are predicted as class’ Y’ and vice versa. Moreover, a confusion matrix can be applied to multi class problems as well. The accuracy will be calculated as the sum of counts in the main diagonal divided by sum of all the counts, as shown in figure 8.



**Fig. 8:** Confusion matrix for a multi class classification problem. [14]

For instance, for the sample confusion matrix mentioned above, the accuracy is calculated as (2+2+1+1)/(2+2+1+1+1+1) = 0.75 i.e. 75%. Additionally, it can be observed that a single tuple, which

actually belongs to class 1, is incorrectly predicted as class 0, and one tuple that belongs to class 0 is incorrectly predicted as class 1. Table 1: Represents critical analysis of contemporary machine learning algorithms with respect to accuracy, precision and characteristic curve

Table 1: Represents critical analysis of contemporary machine learning algorithms

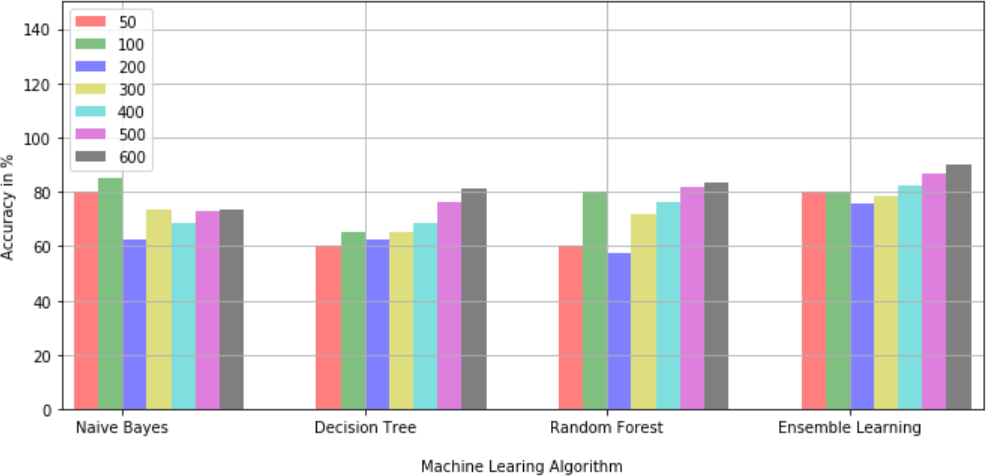
|  |  |  |  |
| --- | --- | --- | --- |
| ***Machine Learning Algorithms ↓/ Parameters →*** | **Accuracy%** | **Precision%** | **Characteristic Curve%** |
| ***Naïve Bayes Algorithm*** | **85** | **80** | **82** |
| ***Decision Tree Algorithm*** | **75** | **78** | **75.8** |
| ***Random Forest Algorithm*** | **81.5** | **83** | **88** |
| ***Ensemble Learning Algorithm*** | **96.6** | **95.74** | **97** |

Figure 9: Represents accuracy% performance of contemporary machine learning algorithms

Figure 10: Represents precision% performance of contemporary machine learning algorithms

Figure 11: Represents characteristic curve% performance of contemporary machine learning algorithms

The inclusion of the concept of Ensemble Learning instead of using a single classification algorithm has improved the accuracy of the system, particularly as the dataset size increases, which is evident from the following plot, as shown in figure 12.



**Fig. 12:** Performance of Naïve Bayes, Decision tree and Random Forest classifiers versus the Ensemble Learning of Naïve Bayes and random Forest classifiers, with increasing dataset size.

Although it is difficult for an algorithm to accurately predict the credibility of any random tweet particularly in real time, on un-moderated online social media like Twitter where information spreads rapidly. However, using the techniques of Data Mining, Natural Language Processing and Ensemble Learning, our model could achieve an accuracy of around 96.44% for detection of misinformation on Twitter.

**V. CONCLUSION AND FUTURE SCOPE**

# The impact of social media on individuals and society at large is significant and cannot be refuted and disregarded. While Twitter and other social media platforms have been successful in connecting millions of people around the world, they are often used to disseminate fake news and malicious content, which can be dangerous, especially in real-world emergencies. To address this monstrous issue, the researchers proposed to detect the news authenticity in real time and hence contribute significantly to the development of a hoax free and better society, the designed system employs data mining, ensemble learning, and natural language processing algorithms and is critically compared with other contemporary techniques like Naïve bayes, Random Forest and Decision Tree to establish its positive outcomes and gains. Furthermore, this concept can be extended to other popular social media platforms such as Facebook and instant messaging applications like WhatsApp. A system can be proposed and developed to verify the veracity of information by taking inputs from multiple sources, thereby enhancing the accuracy of predictions and preventing the spread of misinformation more efficiently.

# REFERENCES

1. Feng Qian Shu, He Jiang, Natali Ruchansky, Ming Zhang, Yan Liu, Hide, "Combating Fake News: A Survey on Identification and Mitigation Techniques", ACM Transactions on Intelligent Systems and Technology 10(3):1-42, DOI:10.1145/3305260, 2022.

2. Kai Shu Gilda, Amy Sliva, Suhang Wang, Jiliang Tang,"Detecting Fake News on Social Media: A Data Mining Perspective", ACM SIGKDD Explorations Newsletter, 19(1), DOI:10.1145/3137597.3137600, 2022.

3. Xinyi Zhou, Reza Zafarani, "A Survey of Fake News Detection Tools: Techniques, Methods, and Opportunities", A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities, Vol. 12, Issue 2, 2023.

4. R. Zellers et al., "Defending Against Neural Fake News," arXiv:1905.12616 [cs], Dec. 2020, Accessed: Mar. 19, 2021. [Online]. Available: http://arxiv.org/abs/1905.12616.

5. Kai Shu1, Deepak Mahudeswaran, Suhang Wang, "FAKENEWSNET: A Data Repository with News Content, Social Context and Spatiotemporal Information for Studying Fake News on Social Media", Journal of Big Data (Mary Ann Liebert Inc.)-Vol. 8, Iss: 3, pp 171-188, 2022.

6. Dat Quoc Nguyen1 , Thanh Vu2, Anh Tuan Nguyen,"BERTweet: A Pre-trained Language Model for English Tweets" Proceedings of the 2020 EMNLP (Systems Demonstrations), pages 9–14 November 16-20,Association for Computational Linguistics, 2021.

7. Kun Zhou a, Berrak Sisman b, Rui Liu b, Haizhou Li, "Emotion Detection in Multimodal Communication: A Survey", Speech Communication, Volume 137, February 2022, Pages 1-18, 2021.

8. Hong Liang, Xiao Sun, Yunlei Sun & Yuan Gao,"Learning Multilingual Sentiment Representations with Dual Learning", EURASIP Journal on Wireless Communications and Networking volume 2017, Article number: 211,

2021.

9. Lei Zhang, Shuai Wang, Bing Liu, "Deep Learning for Sentiment Analysis: A Survey", Computation and Language (cs.CL), Information Retrieval (cs.IR); Machine Learning (cs.LG); Machine Learning (stat.ML), https://doi.org/10.48550/arXiv.1801.07883, 2021.

10. Ke Wang, Xiaojun Wan,"SentiGAN: Generating Sentimental Texts via Mixture Adversarial Networks", Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18), 4446-4452, 2021.

11. Shobha Tyagi, Adarsh Pai, Jeson Pegado, Ajinkya Kamath,“A Proposed Model for Preventing the spread of misinformation on Online Social Media using Machine Learning”. IEEE Amity International Conference on Artificial Intelligence, 2019 (AICIA’19), Dubai, United Arab Emirates, 2019.

12. Sivasangari V, Vijay Anand Pandian, Santhya R.,“A Modern approach to identify the fake news using machine learning”. International Journal of Pure and Applied Mathematics, Volume 118, 2018.

13. Samah M. Alzanin, Aqil M. Azmi.,“Detecting rumours in social media : A survey”. The 4th International Conference on Arabic Computational Linguistics (ACLing 2018), November 17-19, Dubai, United Arab Emirates,2018.

14. Raveena Dayani, Nikita Chhabra, Taruna Kadian and Rishabh Kaushal,“Rumor Detection in Twitter: An Analysis in Retrospect”. IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), 2015.

15.Sardar Hamidian and Mona Diab. (2015). “Rumor Detection and Classification for Twitter Data”. SOTICS 2015: The Fifth International Conference on Social Media Technologies, Communication, and Informatics, 2015.

16. Qiao Zhang, Shuiyuan Zhang, Jian Dong, Jinhua Xiong, Xueqi Cheng,“Automatic Detection of Rumor on Social Network”. NLPCC 2015 Proceedings of the 4th CCF Conference on Natural Language Processing and Chinese Computing - Volume 9362, 2019.

17. Aditi Gupta, Hemank Lamba, Ponnurangam Kumaraguru, Anupam Joshi. (2019) “Analysing and Measuring the Spread of Fake Content on Twitter during High Impact Events”. Security and Privacy Symposium, 2014.

18. Aditi Gupta, Hemank Lamba, Ponnurangam Kumaraguru, Anupam Joshi, “Faking Sandy: Characterizing and Identifying Fake Images on Twitter during Hurricane Sandy”. Second International Workshop on Privacy and Security in Online Social Media (PSOSM), Rio De Janeiro, Brazil, 2018.

19. Vahed Qazvinian, Emily Rosengren, Dragomir R. Radev, Qiaozhu Mei,"Rumor has it: Identifying Misinformation in Micro blogs”. Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, 2019.

20. Carlos Castillo, Marcelo Mendoza, Barbara Poblete. (2019). "Information Credibility on Twitter”. WWW 2011 – Session: Information Credibility.

21. Bayes’ theorem. Wikipedia, The free Encyclopedia. Retrieved from: <https://wikimedia.org/api/rest_v1/media/math/render/svg/b1078eae6dea894bd826f0b598ff41130ee09c19>.

22. How Are Decision Trees Constructed in Machine Learning. Retrieved from: <https://prateekvjoshi.com/2016/03/22/how-are-decision-trees-constructed-in-machine-learning/>

23. Decision tree-based ensemble methods. Retrieved from: https://subscription.packtpub.com/book/big\_data\_and\_business\_intelligence/9781789132212/2/ch02lvl1sec21/d ecision-tree-based-ensemble-methods.

24. Confusion Matrix. Retrieved from: [http://rasbt.github.io/mlxtend/user\_guide/evaluate/confusion\_matrix/](%20http://rasbt.github.io/mlxtend/user_guide/evaluate/confusion_matrix/)

25. M. Sambridge,1 H. Tkalčić, 1,A. Jackson2, Benford’s law in the natural sciences, GEOPHYSICAL RESEARCH LETTERS, VOL. 37, L22301, doi:10.1029/2010GL044830, 2010.

26. Shubham Bauskar,Vijay Badole, Prajal Jain, Meenu Chawla, Natural Language Processing based Hybrid

Model for Detecting Fake News Using Content-Based Features and Social Features 2019.

27. Bilal Ahmed, Sukkur, Gulsher Ali, Sukkur IBA University Sukkur, Abdul Baseer Buriro, Junaid Ahmed, ‘Analysis of Text Feature Extractors using Deep Learning on Fake News’, Engineering, Technology & Applied Science Research Vol. 11, No. 2,7001-7005, 2021.

28. D. Chopra, N. Joshi, and I. Mathur, "Improving Translation Quality By Using Ensemble Approach," Engineering, Technology & Applied Science Research, vol. 8, no. 6, pp. 3512–3514, Dec. 2018, <https://doi.org/10.48084/etasr.2269>.

29. M. Biniz, S. Boukil, F. Adnani, L. Cherrat, and A. Moutaouakkil, "Arabic Text Classification Using Deep Learning Technics," International Journal of Grid and Distributed Computing, vol. 11, no. 9, pp. 103–114, Sep. 2018, https://doi.org/10.14257/ijgdc.2018.11.9.09.

30 A. Hassan and A. Mahmood, "Efficient Deep Learning Model for Text Classification Based on Recurrent and Convolutional Layers," in 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, Mexico, Dec. 2017, pp. 1108–1113, <https://doi.org/10.1109/ICMLA.2017.00009>.

31. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pretraining of Deep Bidirectional Transformers for Language Understanding," arXiv:1810.04805 [cs], May 2019, Accessed: Mar. 19, 2021. [Online]. Available: <http://arxiv.org/abs/1810.04805>.

32. J. Pennington, R. Socher, and C. Manning, "GloVe: Global Vectors for Word Representation," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, Oct. 2014, pp. 1532–1543, https://doi.org/10.3115/v1/D14-1162.

33. F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," The Journal of Machine Learning Research, vol. 12, pp. 2825–2830, Nov. 2011. [14] "Fake News: Balanced dataset for fake news analysis," Kaggle. https://kaggle.com/hassanamin/textdb3 (accessed Mar. 19, 2021).

34. "Fake news: Fake News Classifier Using Bidirectional LSTM," Kaggle. https://kaggle.com/saratchendra/fake-news (accessed Dec. 19, 2022). [16] H. Christian, M. P. Agus, and D. Suhartono, "Single Document Automatic Text Summarization using Term Frequency-Inverse Document Frequency (TF-IDF)," ComTech: Computer, Mathematics and Engineering Applications, vol. 7, no. 4, pp. 285–294, Dec. 2016, https://doi.org/10.21512/comtech.v7i4.3746.

35. B. Trstenjak, S. Mikac, and D. Donko, "KNN with TF-IDF based Framework for Text Categorization," Procedia Engineering, vol. 69, pp. 1356–1364, Jan. 2014, https://doi.org/10.1016/j.proeng.2014.03.129.

36. W. K. Sari, D. P. Rini, and R. F. Malik, "Text Classification Using Long Short-Term Memory With GloVe Features," Jurnal Ilmiah Teknik Elektro Komputer dan Informatika, vol. 5, no. 2, pp. 85–100, Dec. 2019, https://doi.org/10.26555/jiteki.v5i2.15021.