Chrome Extension for Misinformation Detection

Tanya Lillian Borges
Computer Science and Engineering
Sharda University
Greater Noida, India
tanya.l.borges@gmail.com

Abstract– **Misinformation can be stories, hoaxes, or news deliberately created to spread false news and deceive readers. Fake news has always been a part of our lives. However, it has become a topic of interest only recently. Majorly due to the rise of SOCIAL MEDIA. As stated in news articles the Supreme Court condemns these actions and advice a regulatory mechanism. As the use of social media has increased so, has the number of unreliable sources. During covid, there was a variety of misinformation floating around like, applying cow dung can cure covid. In reality, cow dung can't cure covid but can cause black fungal infection. Some other examples include inciting religious sentiments, causing chaos, and harming someone's reputation. Therefore, detecting fake news and stopping it from spreading is necessary. In this model, I have applied TF-IDF vectorizer and Passive Aggressive Classifiers to train my model. After the training and testing have been completed, a local server using Flask has been set up to assist in the development of the second phase of the project, which is Chrome Extension. The Chrome Extension is called Gossip Checker and sends selected data to the model and returns a predicted score of 0 and 1, where 0 means the data is reliable and 1 is not.**

Keywords–NLP, Tokenization, Misinformation, Chrome Extension, Flask, TF-IDF Vectorizer

I. INTRODUCTION

This section of the research paper gives us an insight into misinformation, NLP, application of NLP in misinformation. Moreover, it covers the application of chrome extension in this model.

A. What is misinformation?

Misinformation is problematic and can be defined as a false statement that hides some important information in order to lead people. It tends to generate mistrust, it can weaken relations [1]. Misinformation is also known as falsehood, ambiguity and so on. For example, there was a Facebook discussion about a recently published product. Both, real and fake users were involved. While the real users discussed and voiced their opinions honestly, fake users praised their product regardless of their true beliefs.

B. Types of misinformation

The term "Fake News" used to be defined as misinformation designed to look as though it is legitimate and real news, but due to its political weaponization and overuse, the term has lost its meaning [2]. There has been a significant growth in the spread of false information due to the wide expansion of social media [3]. Social network users tend to rely on fake news over traditional news, as the former is generally shared by friends and people tend to believe the information sent to them by their close ones. Moreover, it is hard to differentiate between fake and real news, while the user is overwhelmed by the misleading information they have been receiving continuously. People often misinterpret the meaning of misinformation, they simply believe it is equivalent to false information, but that is not true, there are different types of misinformation, each with a different intent, tactic and impact. There are five types of misinformation as shown below:
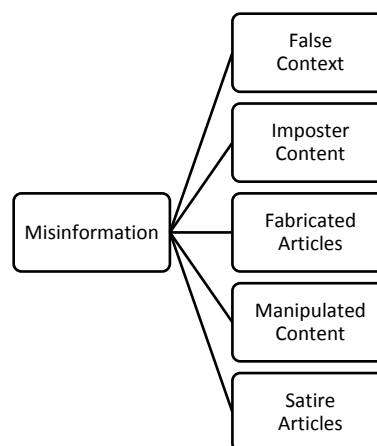

Fig. 1. Key types of misinformation

1. False Context: It takes a piece of content from one place and adds it to another article that may or may not have the same domain, in order to change its meaning [4].

2. Imposter Content: It includes using a well-known name and image so as to fool the people into believing it is authentic.

3. Fabricated Articles: Unlike false context, this type of misinformation is completely made up to deceive you into thinking it is real.

4. Manipulated Content: It involves making changes in the original material, like using editing tools to a photo or video, to affect your opinion.

5. Satire Articles: As the name suggests, it uses humour or exaggeration in order to critique or mock an organization, policy or a person. But sometimes it can be mistaken to be genuine

C. Why is misinformation detection important?

In today's world, anyone and everyone is a journalist, you just need a phone, internet connection and a valid email id. With this you can go on the internet and post anything for the world to see. Unfortunately, this misinformation, fake news or rumours gain a lot of attraction, mostly through web-based applications [5]. Individuals can fall into this trap as they might not double check if the piece of information, they are reading is true or false, they simply read it and forward it. Such activities are not good for society as it can spark

rumours, or build up negative feels against a group of people. Technology moves at a fast pace; we need the preventive measures also to move at the same pace. But this is rarely true, which more often than not leads to exploitation of technology.

There are numerous sites easily available to the general public that provide false data. They are made to intentionally deceit the reader and control the data. There oodles of cases of such sites throughout the world. Due to this, counterfeit news ends up controlling the brains of many individuals across the world [6]. Fake news detection needs to be implemented in order to stop the spread of false information. Misinformation can lead to chaos, mob lynching, killings, and hence it is a great motivation to stop the spread of misinformation [7].
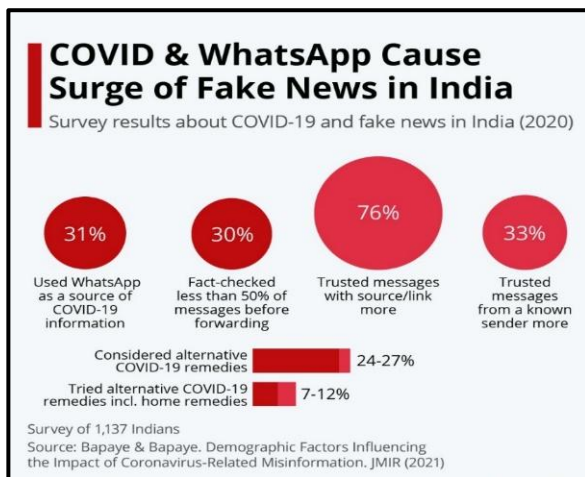

Fig. 2.   COVID and Fake News in India 2020 [8]

This figure explains the role played by WhatsApp and COVID in the spreading of misinformation related to COVID-19.

D. What is a Chrome Extension?

Chrome extensions are programs that are meant to modify and add functionality to a chrome browser [9]. Generally, they are created using web technology like HTML, CSS, JavaScript, etc. The main objective of chrome extensions is to serve the purpose it was built around, in this case the chrome extension will help detect if an article is real or fake. They might have multiple components but all the components finally help in accomplishing the main purpose of the program. An extension's main focus is to provide good functionality with less overhead. An extension usually has minimal interface or is extended to a web page. Some examples of chrome extensions are as follows:

- Ad blocker [10]: They help in blocking ads

- Honey: It checks and applies if a coupon is available while shopping from a site.

- Grammarly [11]: They have a chrome extension to check your grammar and provide writing suggestions.

- Fake new detection: To detect if a piece of article is real or fake.

This project is divided into several phases. The first phase while building this project is to start with the ML model. To create an ML model, we start by pre-processing to

do so the words are broken into tokens through a process called tokenization [12] and we will then remove stop words. Stop words are words like "the", "a", "and" are a few examples of stop words [13]. Such words contribute little to no meaning to the text. Therefore, they are filtered before further processing. We will then move ahead and apply vectorization. Vectorization is the process of mapping phrases of words from our vocabulary to a corresponding vector of real numbers. This helps in finding similarities and word predictions. For this, we have used two famous vectorizers, Count Vectorizer and TF-IDF vectorizer. Now that the data has been pre-processed, we will model the data using three algorithms, Naïve-Bayes, Logistic Regression and Passive Aggressive Classifier and decide the better one. The next phase includes deploying the model into an external web server by creating a flask server. To set up the flask server, two routes have been created, one to train the model and another to give us the result. After setting up the flask server, a url will be generated which is used in our chrome extension to route the text from the web page to our server and return the prediction.

II.   LITERATURE REVIEW

In this section of the research paper, I will present an overview of the developments of NLP and AI methods in the domain of misinformation detection. This section is as important as any other as it will be the foundation for the development of the algorithms that will be used later. Initially, NLP algorithms lacked generalizability and flexibility, they relied on human-engineered and were based on rules. Moreover, experts determine the important linguistic features for each task, the discover of patterns or trends are limited. Even though, rule-based systems are still prevalent in NLP systems, the growth of ML has aided major advances in the field.

Techniques used for Misinformation Analysis

Discussed below are the two types of methods researched in the field of misinformation analysis.

A. Machine Learning Based Methods

Gilda [14] proposed a technique where in TF-IDF (Term Frequency-Inverse Document Frequency) of bi-grams and PCFG (Probabilistic Context Free Grammar) detection to their dataset, acquired from Signal Media and a listing of re-assets from OpenSources.co, of 11,000 articles. He then applied various classification algorithms like Decision Trees, SVM (support vector machine), Random Forests, Gradient Boosting and Stochastic Gradient Descent. He achieved an accuracy of 77.2 and a promising potential for the predictive power. Jain et al. [15] applied the Naïve-Bayes classification algorithm to a Facebook dataset in order to classify a post as fake or real, while doing so they achieved an accuracy of 80.7%. Khan et al. [16] proposed that in a dataset with less that 100k news articles, Naïve-Bayes with n-gram attains a result equivalent to that of a model based on neural networks. They also noted that the performance of Long-Short Term Memory (LSTM) model usually depends on the information in the news article and the length of the dataset, when the model is provided with adequate information, they can overcome overfitting due to a higher probability. Manzoor et al. [17] studied the ML approaches that have been implemented and suggested that more emphasis should be paid to DL methods so as to produce an efficient fake news detector. Jain et al. [18] proposed a model where SVM,

Naïve-Bayes and NLP were used, at the end they received an accuracy of 93.6%. Ahmad et al. [19] developed a model where they used ensemble methods in ML. They extracted the different features from the text using Linguistic Inquiry and Word Count (LIWC) took and they trained and parameter-tuned the model. They received an accuracy of 95.6%. Ibrishimova et al. [20] proposed a hybrid model, that consisted of 5 NLP features, Google NLP API for sentiment analysis, title length, stopwords percentage, ARI readability, and ratio of proper nouns to nouns, combined with three knowledge verification features, such as whether similar titles appear in more sources, whether the title is based on facts or opinions, and whether the title is similar to a title from a reputable and trusted source Oliveira et al. [21] proposed using latent semantic analysis to apply

dimensionality reduction techniques and data compression (LSA). They used three distinct news categorization strategies after reducing the amount of variables, two of which used cascade or unique configurations of learning algorithms and the other statistically analysing the difference between the types of news. Propaganda Spotting in Online Urdu Language (ProSOUL) was proposed by Kausar et al. [22]. It is, as the name implies, a framework for identifying propaganda content and sources in Urdu. They created the LIWC dictionary to aid in the extraction of psycholinguistic characteristics. Ahmed et al. [23] studied three research questions, why is ML required to detect fake news; which ML classifiers must be used to detect fake news; and how are these classifiers trained.

TABLE I. ML-BASED METHODS FOR MISINFORMATION ANALYSIS

Author	Year	Objective	Techniques	Benefits
Gilda [14]	2017	Application of natural language processing techniques for the detection of fake news	TF-IDF of bi-grams and PCFG detection. The model was tested on multiple classification algorithms - Gradient Boosting, SVM, Random Forests, Stochastic Gradient Descent, and Bounded Decision Trees.	Promising potential predictive power, even when ignoring named entities
Jain et al. [15]	2018	Apply fake news detection on Facebook	Naive Bayes classification model to predict whether a post on Facebook will be labelled as REAL or FAKE.	Detects fake news, can be made better by omitting stop words
Khan et al. [16]	2019	Assess performance of different approaches on datasets.	Overall performance analysis of different approaches on three different datasets. Developed a dataset.	Gives clarity about working of different techniques on different size datasets.
Manzoor et al. [17]	2019	Review ML approach to detect fake news, limitation, improvisation by implementing DL.	Systematic review of different machine learning approach to detect fake news.	Provides information about different techniques and suggests deep learning.
Jain et al. [18]	2019	Detecting fake news and stopping them	Feature extraction: counter-vectorizer generation, train: Naive-Bayes, Support vector machine.	Detected fake news and suggested news on that topic which is very useful for any use
Ahmad et al. [19]	2020	Identify patterns in text to differentiate fake articles from true news	Extracted unique textual functions from articles the usage of LIWC tool, used the feature, set an enter to the fashions. Learning fashions skilled and parameter-tuned for ultimate accuracy	Improved results, tested on 4 datasets
Ibrishimova et al. [20]	2020	Defined what constitutes fake news in the context of information warfare, and to propose an automated method for fake news detection	A hybrid framework for faux information detection that makes use of know-how verification functions further to NLP functions.	Defines and detects fake news
Oliveira et al. [21]	2020	Discriminate between real and fake news	Computational-stylistic analysis on NLP, efficient application ML algorithms	Differentiates between fake news and real news
Kausar et al. [22]	2020	Identify content and sources of propaganda spread in the Urdu language.	Linguistic dictionary of LIWC: extract psycho-linguistic features of Urdu. Detailed analysis of classifiers with Word2Vec, BERT, n-gram, and NELA features	Identifies propaganda content in Urdu leangueag
Ahmed et al. [23]	2021	Reviewed different machine learning techniques for fake news detection	Papers accumulated from different sources, and then discussed.	Answered important research questions related to the topic

B. Deep Learning Based Methods

Liu et al. [24] proposed a neural network for timely detection of misinformation. Their model consisted of three parts: a position-aware attention mechanism that highlights user responses, a status-sensitive crowd response feature extractor that extracts text and user features, and a multi-region mean-pooling mechanism that aggregates features across multiple window sizes. A state sensitive crowd reaction feature extractor, CNN-based news classifier, and PU-Learning framework were the three components of their

model. Islam et al. [25] looked at user connections and past activities to see how they contribute to fake news diffusion. They combined the user's human mental state with historical data to better analyse the user's behaviour, as the tendency to spread misleading information is linked to the user's human mental state. Jiang et al. [26] compared the performance of five machine learning models and three deep learning models on two different fake and real news datasets of varying sizes. Aside from that, they used approaches like phrase frequency, tf-idf, and embedding to obtain text representations for the models.

TABLE II.   DL-BASED METHODS FOR MISINFORMATION ANALYSIS

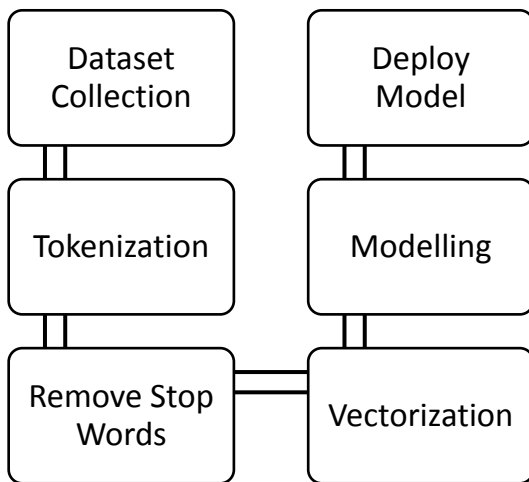| Author	Year	Objective	Techniques	Benefits
Yang Liu et al. [24]	2020	Create a neural network to detect fake news	Input is Status-sensitive crowd reaction, CNN classification module, incorporates position-aware attention mechanism and multi-region mean pooling, to facilitate detection.	Novel deep neural network to detect fake news.
Islam et al. [25]	2020	Reviewed different techniques, suggested deep learning is the best	They analysed person connections and their historic pastime and mirror how they relate to the unfold of faux news. They included human intellectual circumstance with the person's historic data, that could higher examine the person's pastime, because the tendency to unfold fake records is associated with the person's human intellectual circumstance.	Incorporates human mental condition with user's data, and analyses the user's activity.
Hassan et al. [27]	2020	Classify the veracity labels of a statement made by politicians into fine-grained classes	Combination of twin GRU layers and lexical capabilities which have been similarly optimise via way of means of the use of Gaussian Noise.	Classified the labels of statements made by a politician
Xie et al. [28]	2020	Identify if a frame of a video is fake or real.	Utilized a changed AlexNet built of an association of 6 layers: convolution2d, max pooling, dense, flatten, activation and dropout layers	Managed to identify if video was real or fake
Jiang et al. [26]	2021	Fake news detection	ML fashions like, logistic regression, SVM, k-nearest neighbour, choice tree, random forest, and DL fashions like CNN, gated recurrent network (GRU), LSTM. Finally stacking approach of schooling some other RF version primarily based totally at the prediction effects of all man or woman fashions	Detects fake news with high accuracy

III.   METHODOLOGY

Fig. 3.   Workflow Diagram

A. Dataset

In this python project, we have used a CSV dataset will be used. The dataset contains 7796 rows and 4 columns. The dataset has two different CSV files, one to train the model and the other to test it. The main aim of this specific dataset is to differentiate between real and fake instances.

The dataset has the following attributes:

- id: provides the unique id of the given news article

- title: provides the title of the given news article

- author: provides author of the given news article

- text: provides the content of the given news article

- label: marks the given news article according to reliability, 1 refers reliable and 0 is unreliable.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20800 entries, 0 to 20799
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   id      20800 non-null  int64
 1   title   20242 non-null  object
 2   author  18843 non-null  object
 3   text    20761 non-null  object
 4   label   20800 non-null  int64
```

Fig. 4.   Attributes of the dataset

B. Tokenization

Tokenization, as previously stated, is the act of dividing raw text into little words or sentences known as tokens. Tokens aid in the comprehension of text and the development of an NLP model. Tokenization analyses the order in which the words appear in the text to determine its meaning. It's termed text tokenization if the separation is done to split the text into words, and it's called sentence tokenization if it's done to break the text into sentences. There are a variety of tokenization strategies that we can apply based on language and modelling. Natural Language Toolkit (NLTK), Textblob, and Gensim are some of the libraries that have been utilised to perform this task. To achieve tokenization in the current model, NLTK, a python library that aids in NLP, developed by Microsoft, was employed.

C. Removing Stop Words

Stop words are words that contribute little to no meaning to the text. Therefore, to simplify the text and make the detection of misinformation easier, we have attempted to remove all the stop words. Their examples include 'a', 'the', 'and', and so on.

D. Vectorization

- Count Vectorization:

The count vectorizer simply counts the amount of times a given token or text appears in the text and assigns the resultant value as the weight [29]. This is subsequently turned into a token count vector. Count Vectorizer also aids in the pre-processing of text data prior to its conversion to vector form. As a result, count vectorizer is a text feature representation module with a lot of flexibility.

- TF-IDF Vectorization:

The TF-IDF algorithm combines two algorithms: phrase frequency (which displays how frequently a word appears in a document divided by the total number of terms in the document) and inverse document frequency (which shows how unique and rare the words are) [30]. The TF-IDF method is used to count the number of words in a collection of documents. The weight given to tokens is determined not just by their frequency, but also by their recurrence. We employed TF-IDF vectorization in this model.

E. Modelling

After vectorization the next step is modelling. Here three models are tested:

1) Naïve Bayes:

It is a Bayes Theorem-based supervised learning algorithm. It's employed in text categorization with a large number of dimensions, like ours. It makes predictions based on an object's probability. It is presumptively assumed that the occurrence of one feature is unrelated to the occurrence of other traits [21]. It transforms the data into frequency tables and then uses probability to build the likelihood table. Finally, P(A|B) is calculated.

With prior knowledge, the Bayes Theorem is used to calculate the probability of a hypothesis. Its conditional probability that determines this.

Bayes Theorem is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.

Formula:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \qquad (1)$$

Where, P(A|B) = probability of A on observed event B

P(B|A) = probability of B given A is true

P(A) = probability of A

P(B) = probability of B

2) Logistic Regression

logistic regression is used when the dependent variable or target is categorical, that is, either true or false [32]. Some examples where logistic regression can be used is when we need to classify if a piece of information is true or false, or if the sentiment of a text is sad or happy.

3) Passive Aggressive Classifier:

They are a family of ML algorithms that are used in large-scale learning. Here, if the classification is correct, we will keep the model, and if the classification is not correct,

we will update the model so as to adjust it for the more misclassified examples [33]. This algorithm is different from the others as it doesn't converge but updates itself to correct losses.

F. Deploy Model as Chrome Extension

After building the ML model on the local machine, the model hosted to an external web server. To do so Flask has been used, and then a chrome extension is created and deployed the entire project.

Flask is a micro-framework written in Python for python applications. It helps to implement ML applications so that they can be easily plugged, extended and deployed as a web application [34]. Flask provides a glue layer to any python application. It makes it easier to embed and existing python application. Flask is a collection of software packages that help create a web application and extend it further.

To set up the flask server, two routes have been created, one to train the model and another to give us the result. In the result route, a post request is created to send the selected text data to our server that will help in predicting our result. After setting up the flask server, a url will be generated which is used in our chrome extension to route the text from the web page to our server and return the prediction.

The next step was setting up the Chrome Extension. The Chrome Extension consists of a few files, these files have been discussed below:

1) Manifest file:

A manifest file is a file in the format json, and is used to tell chrome important information that is used to define your extension, like its name, version and the permissions it requires. Without a manifest file you can't have any chrome extension. Another thing to note is that when setting version, we will always set it as 2 and not 1 as the latter is unsupported as of January 2014.

2) Popup UI:

This consists of two different files, popup.html and popup.js. The popup.html file is used to create the structure of the popup box whereas the .js file helps to control the behaviour and working of the server. It is through the .js file that we will send the selected text to our model and help in the prediction.

IV. RESULTS

After testing the three models, it was found that PAC is the best choice, as it corrects its errors rather than converging. Moreover, PAC is better at correcting an error when there is some misclassification.

The ML model achieved an accuracy of 96.63%, which is greater than the existing models. So, we used this model to create a chrome extension as discussed above.

V. CONCLUSION

This research paper discusses the development of Gossip Checker, a chrome extension that helps in the detection of Fake News. To build the basic ML model, three algorithms have been tested, Naïve-Bayes, Logistic Regression and Passive Aggressive Classifiers, in the end it was concluded that Passive Aggressive Classifiers are the best option, due to its efficiency and ability to rectify its mistakes. After the development of the model, a chrome extension was

established using HTML and JavaScript and used Flask to deploy the ML model on the local machine.

REFERENCES

[1] T. Tran, P. Rad, R. Valecha and H. R. Rao, "Misinformation Harms During Crises: When the Human and Machine Loops Interact," 2019 IEEE International Conference on Big Data (Big Data), 2019, pp. 4644-4646, doi: 10.1109/BigData47090.2019.9005561.

[2] Zhang H, Alim MA, Li X, Thai MT, Nguyen HT (2016) Misinformation in online social networks: detect them all with a limited budget. ACM Trans Inf Syst 34:1–24

[3] Saxena, A., Hsu, W., Lee, M. L., Leong Chieu, H., Ng, L., & Teow, L. N. (2020). Mitigating misinformation in online social network with top-K debunkers and evolving user opinions. Companion Proceedings of the Web Conference 2020. https://doi.org/10.1145/3366424.3383297

[4] Introduction and background. Disinformation and 'fake news': Interim Report - Digital, Culture, Media and Sport Committee - House of Commons. (n.d.). Retrieved November 29, 2021, from https://publications.parliament.uk/pa/cm201719/cmselect/cmcumeds/363/36304.htm.

[5] Meel, P., & Vishwakarma, D. K. (2020). Fake news, rumor, information pollution in social media and web: A contemporary survey of state-of-the-arts, challenges and opportunities. Expert Systems with Applications, 153, 112986.

https://doi.org/10.1016/j.eswa.2019.112986

[6] Van Bavel, J. J., & Pereira, A. (2018). The Partisan Brain: An identity-based model of political belief. Trends in Cognitive Sciences, 22(3), 213–224.

https://doi.org/10.1016/j.tics.2018.01.004

[7] Haque, M. M., Yousuf, M., Alam, A. S., Saha, P., Ahmed, S. I., & Hassan, N. (2020). Combating misinformation in Bangladesh. Proceedings of the ACM on HumanComputer Interaction, 4(CSCW2), 1–32. https://doi.org/10.1145/3415201

[8] https://www.statista.com/chart/25031/covid-19-misinformation-on-whatsapp-india/

[9] Mehta, P. (2016). Creating google chrome extensions. Apress.

[10] A. H. Lashkari, A. Seo, G. D. Gil and A. Ghorbani, "CIC-AB: Online ad blocker for browsers,"2017 International Carnahan Conference on Security Technology (ICCST), 2017, pp. 1-7, doi: 10.1109/CCST.2017.8167846.

[11] Fitria, T. N. (2021). Grammarly as AI-powered English writing assistant: Students' alternative for writing English. Metathesis: Journal of English Language, Literature, and Teaching, 5(1), 65. https://doi.org/10.31002/metathesis.v5i1.3519

[12] Webster, J. J., & Kit, C. (1992). Tokenization as the initial phase in NLP. Proceedings of the 14th Conference on Computational Linguistics -. https://doi.org/10.3115/992424.992434

[13] Othman, M., Hassan, H., Moawad, R., & Idrees, A. M. (2016). Using NLP approach for opinion types classifier. Journal of Computers, 400–410. https://doi.org/10.17706/jcp.11.5.400-410

[14] IEEE Malaysia Section, Universiti Putra Malaysia. Department of Electrical and Electronic Engineering, Universiti Putra Malaysia. IEEE Student Branch, and Institute of Electrical and Electronics Engineers, IEEE Student Conference on Research and Development (SCOReD) 2017 : Inspiring technology for humanity : proceedings : 13-14 December, Putrajaya, Malaysia.

[15] J. Singhai, S. Gupta, Maulana Azad National Institute of Technology, IEEE Asia-Pacific Region. MP Subsection, IEEE Asia-Pacific Region. Bombay Section, and Institute of Electrical and Electronics Engineers, 2018 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS) : February 24-25, 2018, Maulana Azad National Institute of Technology, Bhopal.

[16] J. Y. Khan, Md. T. I. Khondaker, S. Afroz, G. Uddin, and A. Iqbal, "A Benchmark Study of Machine Learning Models for Online Fake News Detection," May 2019, doi:

10.1016/j.mlwa.2021.100032.

[17] SCAD College of Engineering and Technology and Institute of Electrical and Electronics Engineers, Proceedings of the International Conference on Trends in Electronics and Informatics (ICOEI 2019) : 23-25, April 2019.

[18] IEEE Staff, 2019 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT). IEEE, 2019.

[19] I. Ahmad, M. Yousaf, S. Yousaf, and M. O. Ahmad, "Fake News Detection Using Machine Learning Ensemble Methods," Complexity, vol. 2020, 2020, doi: 10.1155/2020/8885861.

[20] M. D. Ibrishimova and K. F. Li, "A machine learning approach to fake news detection using knowledge verification and natural language processing," in Advances in Intelligent Systems and Computing, 2020, vol. 1035, pp. 223–234. doi: 10.1007/978-3-030-29035-1_22.

[21] N. R. de Oliveira, D. S. V. Medeiros and D. M. F. Mattos, "A Sensitive Stylistic Approach to Identify Fake News on Social Networking," in IEEE Signal Processing Letters, vol. 27, pp. 1250-1254, 2020, doi: 10.1109/LSP.2020.3008087.

[22] S. Kausar, B. Tahir and M. A. Mehmood, "ProSOUL: A Framework to Identify Propaganda From Online Urdu Content," in IEEE Access, vol. 8, pp. 186039-186054, 2020, doi:10.1109/ACCESS.2020.3028131.

[23] Ahmed, Alim Al Ayub & Aljabouh, Ayman & Donepudi, Praveen & Choi, Myung. (2021). Detecting Fake News Using Machine Learning : A Systematic Literature Review.

[24] Y. Liu and Y. F. B. Wu, "FNED: A Deep Network for Fake News Early Detection on Social Media," ACM Transactions on Information Systems, vol. 38, no. 3. Association for Computing Machinery, Jun. 01,2020. doi: 10.1145/3386253.

[25] M. R. Islam, S. Liu, X. Wang, and G. Xu, "Deep learning for misinformation detection on online social networks: a survey and new perspectives," Social Network Analysis and Mining, vol. 10, no. 1. Springer, Dec. 01, 2020. doi: 10.1007/s13278-020-00696-x.

[26] T. Jiang, J. P. Li, A. U. Haq, A. Saboor and A. Ali, "A Novel Stacking Approach for Accurate Detection of Fake News," in IEEE Access, vol. 9, pp. 22626-22639, 2021, doi: 10.1109/ACCESS.2021.3056079.

[27] F. M. Hassan and M. Lee, "Political Fake Statement Detection via Multistage Feature-assisted Neural Modeling," 2020 IEEE International Conference on Intelligence and Security Informatics (ISI), 2020, pp.1-6, doi: 10.1109/ISI49825.2020.9280531.

[28] D. Xie, P. Chatterjee, Z. Liu, K. Roy and E. Kossi, "DeepFake Detection on Publicly Available Datasets using Modified AlexNet," 2020 IEEE Symposium Series on Computational Intelligence (SSCI), 2020, pp. 1866-1871, doi: 10.1109/SSCI47803.2020.9308428.

[29] Singh P. (2019) Natural Language Processing. In: Machine Learning with PySpark. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-4131-8_9

[30] T. K. Tran and H. Sato, "NLP-based approaches for malware classification from API sequences," 2017 21st Asia Pacific Symposium on Intelligent and Evolutionary Systems (IES), 2017, pp. 101-105, doi: 10.1109/IESYS.2017.8233569.

[31] G. Wenfeng, Z. Hong and C. Ruoyi, "A Study on Online Detection of micro-blog Rumors Based on Naive Bayes Algorithm," 2020 Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC), 2020, pp. 22-25, doi: 10.1109/IPEC49694.2020.9115171.

[32] M. A. Shafin, M. M. Hasan, M. R. Alam, M. A. Mithu, A. U. Nur and M. O. Faruk, "Product Review Sentiment Analysis by Using NLP and Machine Learning in Bangla Language," 2020 23rd International Conference on Computer and Information Technology (ICCIT), 2020, pp. 1-5, doi: 10.1109/ICCIT51783.2020.9392733.

[33] S. Mhatre and A. Masurkar, "A Hybrid Method for Fake News Detection using Cosine Similarity Scores,"2021 International Conference on Communication information and Computing Technology (ICCICT), 2021, pp. 1-6, doi: 10.1109/ICCICT50803.2021.9510134.

[34] A. Yaganteeswarudu, "Multi Disease Prediction Model by using Machine Learning and Flask API," 2020 5th International Conference on Communication and Electronics Systems (ICCES), 2020, pp. 1242-1246, doi: 10.1109/ICCES48766.2020.9137896.