

# **CSE1901 - Technical Answers to Real World Problems (TARP)**

## **Project Report**

### **Fake News Detection**

*By*

19BAI1118

Ashwin U Iyer

19BAI1121

Rahul Dash

19BAI1128

Rishma Mukhopadhyay

19BAI1129

Hitesh Goyal

19BAI1157

Yash Tripathi

**B. Tech Computer Science and Engineering**

*Submitted to*

**Dr. Nayeemulla Khan**

**School of Computer Science and Engineering**



**VIT<sup>®</sup>**  
**Vellore Institute of Technology**  
(Deemed to be University under section 3 of UGC Act, 1956)

*April 2022*

## **DECLARATION**

I hereby declare that the report titled “**Fake News Detection**” submitted by me to VIT Chennai is a record of bona-fide work undertaken by me under the supervision of **Dr. Nayeemulla Khan**, School of Computer Science and Engineering, Vellore Institute of Technology, Chennai.

A handwritten signature in black ink, consisting of a circle with a stylized 'H' inside, followed by a series of loops and a final flourish.

Signature of the Candidate

**Hitesh Goyal**

**19BAI1129**

## **CERTIFICATE**

Certified that this project report entitled “**Fake News Detection**” is a bonafide work of **Ashwin U Iyer (19BAI1118), Rahul Dash (19BAI1121), Rishma Mukhopadhyay (19BAI1128), Hitesh Goyal (19BAI1129), Yash Tripathi (19BAI1157)** and they carried out the Project work under my supervision and guidance for CSE1901 - Technical Answers to Real World Problems (TARP).

**Dr. Nayeemulla Khan**  
SCOPE, VIT Chennai

A handwritten signature in black ink, consisting of a circle with a stylized 'H' inside, followed by a series of loops and a long horizontal stroke.

Signature of the Candidate

**Hitesh Goyal**  
**19BAI1129**

## **ACKNOWLEDGEMENT**

We wish to express our sincere thanks and deep sense of gratitude to our project guide, **Dr. Nayeemulla Khan**, School of Computer Science and Engineering for her consistent encouragement and valuable guidance offered to us throughout the course of the project work.

We are extremely grateful to **Dr. R. Ganesan**, Dean, School of Computer Science and Engineering (SCOPE), Vellore Institute of Technology, Chennai, for extending the facilities of the School towards our project and for his unstinting support.

We express our thanks to our Head of the Department, **Dr.J.Priyadarshini** for his support throughout the course of this project.

We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the courses.

We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

## **ABSTRACT**

Fake news, hoaxes and superstitions based on this news have been there since before the arrival of the Internet. Fake over news on the Internet can be defined as ‘fictitious articles deliberately fabricated to deceive readers.’ Social media and some news outlets publish fake news to increase readership or as part of psychological warfare. The idea behind fake news is profiting through clickbaits. Clickbaits increase curiosity and lure users with flashy headlines that are designed to click links increasing advertisement revenues. There is a sharp increase in the volume of fake news, made possible by the emergence of social networking sites. This project aims to create a solution to determine the authenticity of a given news statement. We train our model using word embeddings and use a neural network architecture to classify the news. The final application is deployed as an extension that fetches the username of the Twitter account and hence the set of tweets. We provide an authenticity report about each tweet.

## **CONTENTS**

Declaration	i
Certificate	ii
Acknowledgement	iii
Abstract	iv
1 Introduction	1
1.1 Objective and Goal of the Project	1
1.2 Problem Statement	1
1.3 Motivation	1
1.4 Challenges	1
2 Literature Survey	1
3 Requirements Specification	3
3.1 Hardware Requirements	3
3.2 Software Requirements	3
4 System Design	3
5 Implementation of System	4
6 Results and Discussion	7
7 Conclusion and Future Work	9
8 References	10

# **1. INTRODUCTION**

## **1.1 OBJECTIVE AND GOAL OF THE PROJECT**

Create a model that will detect if an article is authentic, or fake based on the embeddings of the actual content. Using these GloVe embeddings, we train our neural network model with the goal of providing an authenticity score for each news article passed to the model.

Our goal is to build a web-based application that can profile news channel's Twitter accounts and the articles posted on their tweets to check the authenticity of the posts. This can help fact check the news in a much more effective manner, thus helping the common man and average social media user stay informed on how trustworthy the content they're reading can be.

## **1.2 PROBLEM STATEMENT**

Fake News Detection based on Twitter Accounts

## **1.3 MOTIVATION**

Fake news is false or misleading information presented in the form of news. It is often created with the aim of damaging the reputation of a person or other entity. Deliberately misinforming or deceive the reader can have adverse effect on the reader and may lead to potential monetary damage. There is, hence, a need to check the truthfulness and authenticity of major news articles to decide the news reliability. The project aims to provide a one-click solution to check the correctness of a given news.

## **1.4 CHALLENGES**

Fake news detection is quite a difficult project area. The major challenge is the unavailability or shortage of labeled data for training the detection models. It is challenging to scale well with the bulks of newly created news, specifically on social media.

# **2. LITERATURE SURVEY**

[1] pointed out numerous media sources and conducted appropriate research to determine whether the given article is genuine or a forgery. Models based on speech features and prediction models that do not fit with other existing models are used in the article. In [2] Naive Bayes was used to detect bogus news using the naive Bayes

classifier. This strategy was tested as a software framework using diverse records from Facebook, Twitter, and other sources, with a 74 percent accuracy. The punctuation problems in the document were overlooked, resulting in poor accuracy. [3] Estimated several machine learning algorithms and conducted research on the percentage of prediction accuracy. Bounded decision trees, gradient improvement, and support vector machine were used to test the accuracy of various predicting patterns. The patterns are calculated with 85-91 percent accuracy using an unstable probability threshold. [4] discuss how to implement false news detection on several social media platforms using the Naive Bayes classifier. They gathered information from Facebook, Twitter, and other social media platforms. Because the material on this site is not 100 percent reliable, accuracy is extremely poor. [5] Used to identify spammers on Twitter. The naive Bayes algorithms, clustering, and decision tree are some of the models employed. Spammers are detected with an average accuracy of 70%, while fraudsters are detected with an average accuracy of 71.2 percent. The models used to distinguish spammers from non-spam have a poor level of intermediate accuracy. In [6] fake news was recognized in a variety of methods. As a language model, the accuracy is restricted to 76%. When a predictive model is utilized, greater accuracy may be gained. [7] aims to identify bogus news using machine learning approaches They use three popular methodologies in their research: Nave Bayes, Neural Networks, and Support Vector Machines (SVM). Prior to using machine learning to categorize the data, the normalization process is an important step in data purification. The results showed that Nave Bayes can detect false communications with a 96.08 percent accuracy. The accuracy of two more sophisticated approaches, the neural network, and the machine vector (SVM), was 99.90%. In [8] Fake news identification has been discovered to be a predictive analytic application. The three processes of processing, feature extraction, and classification are used to detect fraudulent texts. Show bogus news is the focus of this study's hybrid categorization approach. A mix of KNN and random forests is used to classify the data. The accuracy and recall of the recommended model's execution are examined. Using a mixed false message detection model, the end results increased by up to 8%. On Twitter, [9][10] researchers looked at how fake news was utilized in the 2012 Dutch elections. In the Twitter dataset, she investigates the performance of eight supervised machine learning classifiers.[11]. For the data set with a F score of 88 percent, we suppose that the decision tree method performs best. There were 613,033 tweets assessed, with 328,897 deemed real and 284,136 deemed fake. Features and



attributes of the incorrect material were discovered and classified into six categories after studying the qualitative content of bogus. [12] provided a model for detecting counterfeits based on N-gram analysis via the lenses of multiple characteristic extraction approaches. In addition, we looked at various feature extraction strategies as well as six distinct machine learning methodologies. The proposed model has the best accuracy in practice. A unigram and a linear SVM worksheet are included. The maximum level of precision is 92%.

### 3. REQUIREMENTS SPECIFICATION

#### 3.1 HARDWARE REQUIREMENTS

2GB GPU min 4GB rec

4gb min 8gb rec

I5 8th gen min, i7 10th rec

#### 3.2 SOFTWARE REQUIREMENTS

Tensorflow2+, Python3+, Flask, Tweepy, Google Chrome

### 4. SYSTEM DESIGN



*Figure 1 DATA FLOW PIPELINE*

1. Input URL - Taking the input in the form of a URL to identify the source of the data.
2. Get data from URL - Extracting the relevant data from the provided source. For example, if the source is a twitter account, scraping all the different tweets sent through that account.
3. Send data to analysis function - Send the extracted and preprocessed data to the analysis function. This step provides the summarized analysis.
4. Provide Report- Whatever analysis has been generated by the algorithm is shown as a tabulated display

This entire data flow will be a part of a simple web application which will take an input as a URL or ID and provide the analysis in the tabulated form after the entire processing.

## 5. IMPLEMENTATION OF SYSTEM

The aim with data sourcing is to look for news article based data that looks to classify fake or biased news. The biggest source for such datasets is Kaggle. Sourcing of data will be a process that will require some more in depth searching, but currently, there are few resources we have shortlisted which can be used for fake and biased news:

1. [Ratings of Bias in Electronic Media](#)
2. [Getting Real about Fake News](#)
3. [Fake and Real News Dataset](#)

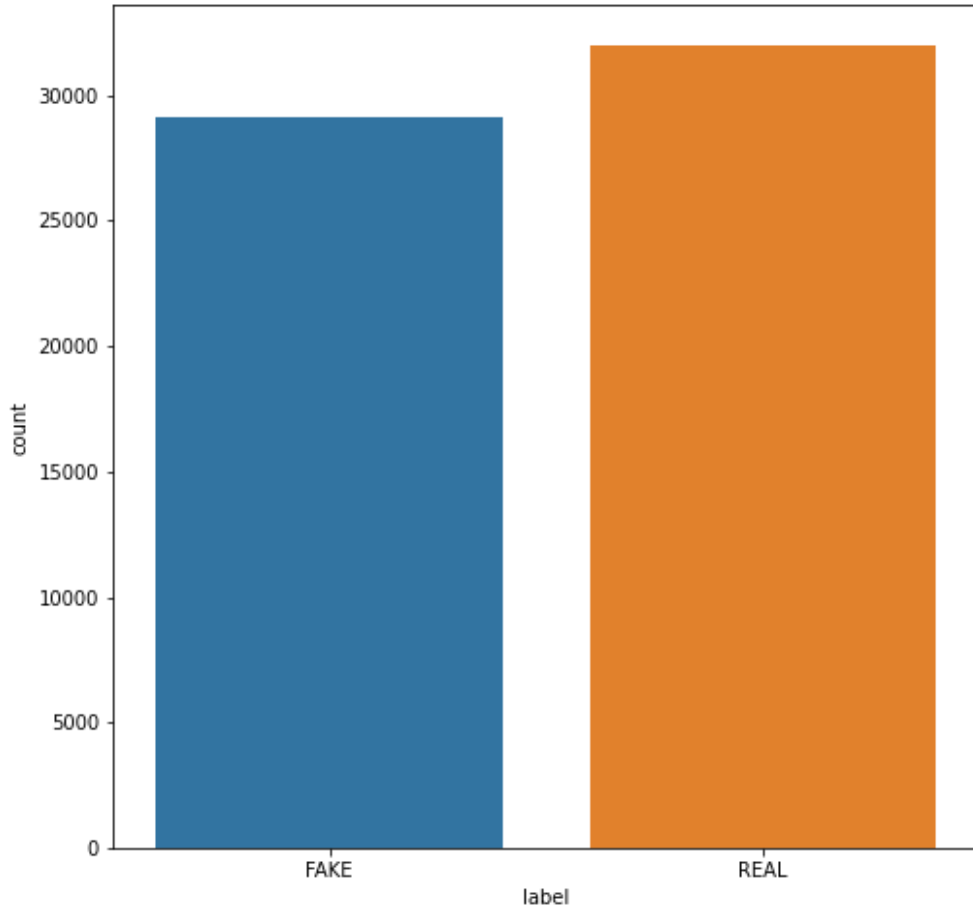
Based on a little bit of initial domain research it was found that most Twitter accounts that are associated with news channels generally tend to post the headlines of the news in the tweet followed by the link to the webpage of the article. Moreover, having more samples of real and fake/biased news in various contexts can help build a master dataset if the need arises. Thus, aggregating data containing news headlines could be a helpful approach and to that effect a few resources found online were:

1. [A Million News Headlines](#)
2. [News Aggregator Dataset](#)
3. [Fake News](#)

The dataset was created by aggregating the source datasets in various ratios to get the required format. Columns like author, source etc. were avoided to not introduce any bias into the classification process.

	title	text	label
0	You Can Smell Hillary's Fear	Daniel Greenfield, a Shillman Journalism Fello...	FAKE
1	Watch The Exact Moment Paul Ryan Committed Pol...	Google Pinterest Digg Linkedin Reddit Stumbleu...	FAKE
2	Kerry to go to Paris in gesture of sympathy	U.S. Secretary of State John F. Kerry said Mon...	REAL
3	Bernie supporters on Twitter erupt in anger ag...	— Kaydee King (@KaydeeKing) November 9, 2016 T...	FAKE
4	The Battle of New York: Why This Primary Matters	It's primary day in New York and front-runners...	REAL

*Table 1* GENERATED DATASET



*Figure 2* DISTRIBUTION OF CLASSES

The final balanced dataset has 61,144 data points i.e., fake and real news articles. 31,992 real news articles, constituting 52.32% of the dataset and 29,152 fake news articles, constituting 47.68% of the dataset.

We must first load the word-embedding dictionary (GloVe by Stanford) which can convert every word to its respective vector form. To do so, we use the GloVe's embedding dictionary which has key value pairs with the keys as tokens (words) and values as the n-dimensional vector corresponding to its numerical interpretation. Then, pass the text through a preprocessing function. The preprocessing function first takes in the text and cleans it. Cleaning includes setting all words to lowercase, removal of unnecessary things like URLs, HTML tags and punctuations. Then, it tokenizes the remaining text and passes a list of tokens to the GloVe embedding function. The embedding function finds the vector for every token and takes an element-wise sum of the feature vectors. This feature vector is then passed on to the model for predictions. After this the model takes an n-dimensional feature vector as input and passes it through 2 hidden layers - the first layer has 128 nodes, second has 64. Both layers use ReLU

activation, have a dropout of 0.2 and are batch normalized. The final layer's output is the probability of the text being real. It is a single node activated by sigmoid.

The web application is created on flask so that it is easier for the team to work with the python code and would ensure faster development of the application. The flask application also allows us to have flexibility in functionality as it can host the application along with it the backend code for the application to function such as the processing of the request from the API module. Requesting the tweets from twitter and sending those tweets to the prediction function is handled by the flask application. It is the glue which holds everything together. We have 3 main pages for the website portion of the application. The Landing page which allows us to inform the user about the application and its purpose. Then we have a search page which allows you to enter some twitter user id's and allows you to search their tweets and get their authenticity score. Third page is the page which records all the scores and displays them with their respective tweets.

The chrome extension fetches the URL from the current tab, from which the username of the twitter account is extracted. This username is used to call the API of the website which is running on localhost. Using fetch, we carry out a POST request. From the API we get the prediction in the form of tweet, and its associated authenticity. The API returns us the recent 10 tweets from the twitter profile. When the extension is clicked, the popup.html file is loaded and using DOM tree a table is created in which the tweet and the authenticity is displayed. All of this is carried out in popup.js which the script attached to popup.html.

The extension starts with a "manifest.json" file. It provides various important information such as manifest version, name of the extension, version name, various actions that need to be performed, language of the extension, description, various icons that are used, and many more.

1. permissions - All the APIs must be registered under this field.
  - a. storage - this API is used to store, retrieve, and track changes to user data
  - b. activeTab - gives the extension temporary access to the currently active tab when the user invokes the extension.
  - c. scripting - execute scripts in various contexts.
2. icons - Extensions also display images on the extension management page, the permissions warning, and favicon. The images are registered in the manifest under icons.

Extensions can have many kinds of forms of user interface such as a popup, tooltip, omnibox and many more. We will use a popup user interface. A button is used to set the background color of the popup-styled extension.

Using `chrome.tabs.query` we fetch the URL of the current tab in the browser. Using `substr()` the username is extracted. The username is stored in the local storage of chrome as a key-value pair, using `chrome.storage.local.set`. The API call is defined in a function named `apicall()` which takes the username as a parameter. The username is fetched from the local storage using `chrome.storage.local.get()`. A POST request is executed to the website `http://127.0.0.1:5000/predict`. We get the response from the API as a json. It contains the 10 recent tweets and their authenticity. Using DOM-tree the report is displayed in a tabular format.

## 6. RESULTS AND DISCUSSION

Results on the website



*Figure 2 Homepage of the Website*



Figure 3 Page to input the username to fetch the tweets



Figure 4 Final report for <https://www.twitter.com/BarackObama>

## Results on the extension

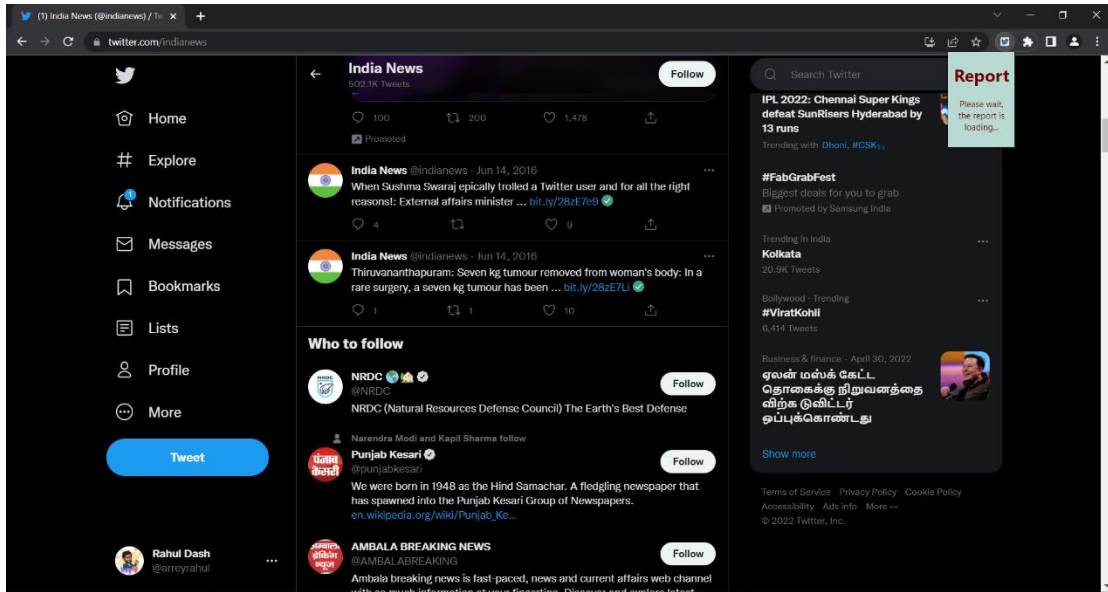


Figure 5 Clicked the extension when in any twitter profile

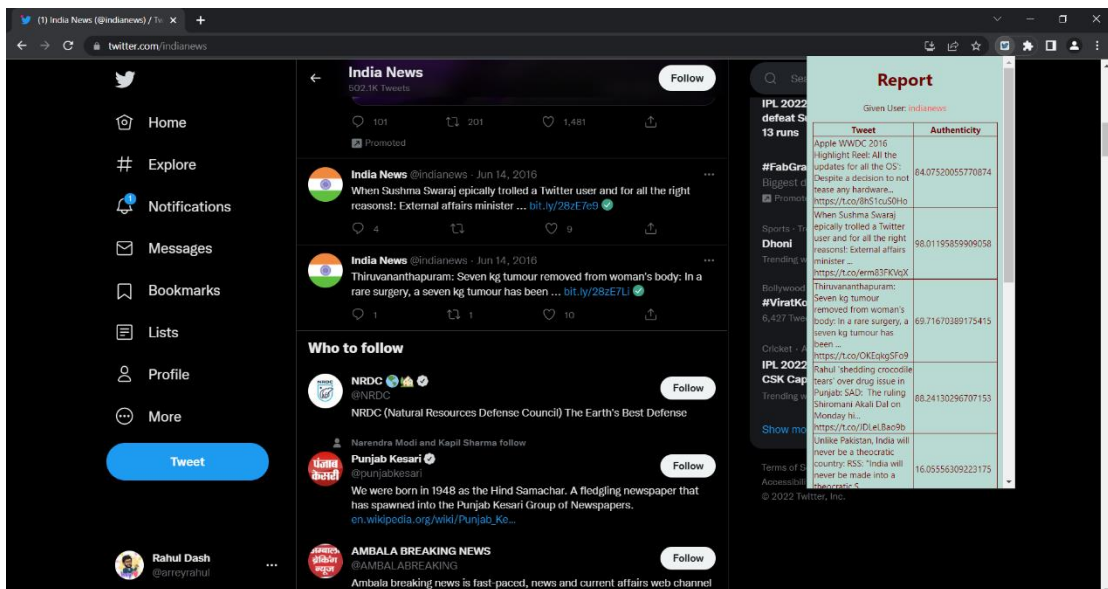


Figure 6 Final Report

## 7. CONCLUSION AND FUTURE WORK

The problem of detecting fake news circulation in Twitter is now a major concern as there are increased reports of organizations and people in positions of power using platforms such as twitter to spread their own propaganda and ideologies. Whereas no clear solution exists, open source or proprietary, our solution to this problem proves that the problem is solvable with Deep Learning and Natural Language Processing techniques. While Machine Learning models may still require monitoring to a certain extent, the ability it has to understand the authenticity of the news can be improved more over further iterations. What we have managed to provide is a simple solution in

the form of a chrome extension as well as a website which can help understand the authenticity of a news headline or even profile a particular news channel's Twitter account. The addition of an API endpoint to query the model by sending Twitter usernames makes our model accessible to other software developers, who can easily make a request and validate a news item. Thus, our solution is versatile, easily reusable and an effective first step towards completely automating the process of detecting fake news on social media platforms.

Future work would include improving the model to handle data drift and concept drift, as the types of fake news and the topics themselves will differ over time, and thus the model needs to be able to keep up with these trends. Currently, the solution only uses Twitter, which can be scaled up to scrape data from all social media platforms and build a better solution overall.

## **8. REFERENCES**

- [1] MykhailoGranik and VolodymyrMesyura. “Fake news detection using naive Bayes classifier.” First Ukraine Conference on Electrical and Computer Engineering (UKRCON). Ukraine : IEEE. 2017.
- [2] Gilda, S. “Evaluating machine learning algorithms for fake news detection.” 15th Student Conference on Research and Development (SCOREd) (pp. 110-115). IEEE. 2017.
- [3] Akshay Jain and AmeyKasbe. “Fake News Detection.” 2018 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS). Bhopal, India: IEEE. 2018.
- [4] Yumeng Qin et al. “Predicting Future Rumours.” Chinese Journal of Electronics Volume: 27 , Issue: 3 , 5 2018, 514 - 520.
- [5] Veronica Perez-Rosas et al. Available at:  
[https://www.researchgate.net/publication/319255985\\_Automatic\\_Detection\\_of\\_Fake\\_News](https://www.researchgate.net/publication/319255985_Automatic_Detection_of_Fake_News) August, 2017.
- [6] Supanya Aphiwongsophon et al. “ Detecting Fake News with Machine Learning Method.” 2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON). Chiang Rai, Thailand, Thailand: IEEE . 2018.



- [7] Prabhjot Kaur et al. “Hybrid Text Classification Method for Fake News Detection.” *International Journal of Engineering and Advanced Technology (IJEAT)* , 2388-2392. 2019.
- [8] Looijenga, M. S. “The Detection of Fake Messages using Machine Learning.” 29 Twente Student Conference on IT, Jun. 6th, 2018, Enschede, The Netherlands. Netherlands: [essay.utwente.nl](http://essay.utwente.nl). 2018.
- [9] I. Traore et al. “Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques.” *International Conference on Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments* (pp. 127–138). Springer International Publishing . 2017.
- [10] Khanam Z., Alkhaldi S. “An Intelligent Recommendation Engine for Selecting the University for Graduate Courses in KSA: SARS Student Admission Recommender System.” In: Smys S., Bestak R., Rocha Á. (eds) *Inventive Computation Technologies. ICICIT 2019. Lecture Notes in Networks and Systems*, vol 98. Springer, Cham. 2019.
- [11] Khanam Z. and Ahsan M.N. “Implementation of the pHash algorithm for face recognition in secured remote online examination system.” *International Journal of Advances in Scientific Research and Engineering (ijasre)* Volume 4, Issue 11 November. 2018.
- [12] Sharma, K., Qian, F., Jiang, H., Ruchansky, N., Zhang, M., & Liu, Y. (2019). Combating fake news: A survey on identification and mitigation techniques. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(3), 1-42