International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Strengthening Fake News Detection: A Resilient Model with TweetTruth

Archana Nanade[*1], Dr. Arun Kumar, Dr. Ashutosh Gupta[3], Dr. Arvind Sharma[3]

Submitted: 25/01/2024 **Revised**: 03/03/2024 **Accepted**: 11/03/2024

Abstract: This study addresses the urgent need to combat misinformation by leveraging the capabilities of BERTweet for advanced fake news detection. The study begins with the pretraining of BERTweet on a diverse corpus, harnessing its ability to comprehend contextual relationships in social media texts. Fine-tuning follows using a meticulously curated dataset representing a variety of sources and deceptive writing styles commonly found in fake news. To enhance the model's resilience, external knowledge sources such as fact-checking databases and reputable news outlets are integrated during both pretraining and fine-tuning. In addition, the study employs data augmentation techniques to address potential imbalances, exposing the model to a broader linguistic spectrum present in fake news on social media platforms.

Keywords: *Fakes News, BERT, BERTweet, TweetTruth, Algorithm*

1. Introduction

The ubiquity of digital platforms and social media has ushered in an era of unprecedented information flow, yet it has also given rise to a pressing concern—fake news [1]. Public discourse, political stability, and individual decision-making processes are all at risk due to the quick and extensive dissemination of false or completely created information, also known as misinformation. Thus, reliable and effective techniques to identify and lessen the impact of false information are critical. This study investigates the use of the cutting-edge natural language processing (NLP) model bidirectional encoder representations from transformers (BERT) in the field of fake news detection[1].

Effective techniques for distinguishing between reliable and misleading information are more important than ever due to the spread of fake news. Traditional approaches often struggle to navigate the intricacies of language, context, and the dynamic nature of online content [2]. In response to these limitations, machine learning techniques, especially those binding advanced NLP models, have attracted attention as a potential solution.

Detecting factual news from a sea of misleading digital information is a challenging task in this era. Taking the help of machine learning to build models that would help in solving such tasks is an hour of the need. In addition, the challenge lies in developing models that are capable of understanding the context, language, subtleties of language, and intent. The Tweetruth model focuses on not only classifying the tweet as fact or not but also providing a significant level of confidence of the tweet either being fact or not.

[1] *Sir Padampat Singhania University, Udaipur, Rajasthan-*
ORCID ID : 0000-0002-2886-5810
[2] *Sir Padampat Singhania University, Udaipur, Rajasthan*
ORCID ID : 0000-0003-1972-5717
[3] *Sir Padampat Singhania University, Udaipur, Rajasthan*
ORCID ID : 0000-0001-6291-1318
[4] *Sir Padampat Singhania University, Udaipur, Rajasthan*
ORCID ID : 0000-0003-3257-0836
* *Corresponding Author Email: archana.nanade@spsu.ac.in*

2. Literature Survey

The literature survey underscores the significance of BERT and BERTweet models in fake news detection, showcasing their effectiveness in capturing contextual nuances and addressing the unique challenges posed by social media platforms. Below is a rigors survey discussing ablut Fake news detection:

Table 1:

Hadeer Ahmed, Issa Traore, and Sherif Saad Year 2017	Detection of Online Fake News Using N-Gram Analysis and Machine Learning	N-gram modeling is a widely used technique for identifying and analyzing features in the language modeling and natural language processing disciplines. In word-based n-gram data, modifications such as stop-word removal, tokenization, sentence segmentation, and punctuation removal are required. Here, a simple classifier based on n-grams can distinguish between fake and genuine news articles.
Reema Aswani, Arpan Kumar Kar & P. Vigne	Detection of Spammers in Twitter Marketing: A Hybrid Approach Using Social Media Analytics and Bio-Inspired Computing	This study employs a mixed research technique that merges insights from social media analytics with bio-inspired computers to simulate Twitter spammers. A statistical t-test identified thirteen significant characteristics, including

swara Ilavarasan Year 2017		emotion diversity, polarity diversity, hashing frequency, unique words, user mentions, linguistic diversity, tweet count, follower count, like count and friend count. The performance of content and semantics analysis is limited by satire and the use of non-English terminology, particularly millennial language, which is a drawback of the current study.
Shu K, Wang S, Liu H Year - 2019	Beyond news contents: Role of social context for fake news detection	Using a TRI-RELATIONSHIP EMBEDDING FRAMEWORK, TriFN can extract beneficial features from news publishers and user encounters independently, while concurrently capturing the interrelationship. However, it is only worthwhile to investigate useful features and models for the early identification of bogus news.
Jwa H, Oh D, Park K, Kang JM, Lim H Year - 2019	exBAKE: Automatic Fake News Detection Model Based on Bidirectional Encoder Representations from Transformers (BERT)	In this study, the authors explain how BERT outperforms previous models. For appropriate analysis, it is essential to grasp the link between words. BERT is intended to determine the precise link between words in a phrase. Only three out of four news categories achieved state-of-the-art performance using this method.
Andri Fachrur Rozie, Andria Arisal, Devi Munandar Year - 2018	Transferring Multi-Channel Convolutional NeuralNetwork Model for Cross-Domain Sentiment Analysis	Using Sentiment Analysis This paper focused mostly on Twitter datasets from the telecommunications, government, and food & beverages industries. Primarily focused on communications. Using a source-based model for weight initialization improves the performance of sentiment analysis significantly. Furthermore, it is intriguing to discover that the source model is not necessarily the most accurate. In this instance, the telecommunications domain is less precise than the Food & Beverage domain. Using sentiment analysis This paper focused mostly on Twitter datasets from the telecommunications, government, and food & beverages industries. Primarily focused on communications. Using a source-based model for weight initialization improves the performance of sentiment analysis significantly. Furthermore, it is intriguing to discover that the source model is not necessarily the most accurate. In this instance, the telecommunications domain is less precise than the Food & Beverage domain. Using sentiment analysis This article focused mostly on Twitter datasets from the telecommunications, government, and food & drinks industries. Primarily devoted to communications. Using a source-based model for weight initialization improves the performance of sentiment analysis significantly. Furthermore, it is intriguing to discover that the source model is not necessarily the most accurate. In this instance, the telecommunications domain is less precise than the food and beverage domain.
Tenney I, Das D, Pavlick E Year - 2019	BERT Rediscovers the Classical NLP Pipeline	This study describes how various levels of the BERT network may resolve syntactic and semantic sentence structure. BERT eliminates the unidirectional limitation by using a mask language model. This work focuses mostly on encoder–BERT and edge probing experiments and this

		model has drawbacks compared with inspection-based probing.
Rohit Kumar Kaliyar, Anurag Goswami, Pratik Narang Year - 2021	FakeBERT: Fake news detection in social media using a BERT-based deep learning approach	Classification findings indicate that the power of autonomous feature extraction using deep learning models is crucial for the precise identification of false news. Deep learning models are renowned for producing state-of-the-art outcomes in various artificial intelligence applications.
Vladislav Kolev, Gerhard Weiss, and Gerasimos Spanakis Year 2022	FOREAL: RoBERTa Model for Fake News Detection Based on Emotions	Embedding for RoBERTa is identical to that for BERT; however, the RoBERTa vocabulary is somewhat broader, and hence the model employs more parameters. This article focuses on the FOREAL-Fake or Real Emotion Analyzer model. The findings of Emotion Classification using the REAL model are encouraging.
Hadeer Ahmed, Issa Traore, and Sherif Saad Year 2017	Detection of Online Fake News Using N-Gram Analysis and Machine Learning	N-gram modeling is a widely used technique for identifying and analyzing features in the language modeling and natural language processing disciplines. In word-based n-gram data, modifications such as stop-word removal, tokenization, sentence segmentation, and punctuation removal are required. Here, a simple classifier based on n-grams can distinguish between fake and genuine news articles.
Reema Aswani, Arpan Kumar Kar & P. Vigneswara Ilavarasa Year 2017	Detection of Spammers in Twitter Marketing: A Hybrid Approach Using Social Media Analytics and Bio-Inspired Computing	This study employs a mixed research technique that merges insights from social media analytics with bio-inspired computers to simulate Twitter spammers. A statistical t-test identified thirteen significant characteristics, including emotion diversity, polarity diversity, hashing frequency, unique words, user mentions, linguistic diversity, tweet count, follower count, like count and friend count. The performance of content and semantics analysis is limited by satire and the use of non-English terminology, particularly millennial language, which is a drawback of the current study.

3. Proposed Methodology

The proposed model employs the traditional BERT architecture with additional tests to obtain the highest level of precision. In [23], we had already explained the workings of the Tweetruth model, and like any other classifier, it predicts whether the tweet is fake or not. This is the traditional way of building a classifier and leaving it to do the predictions; however, there is a requirement of continuously updating the model with recent news and trends, checking the source, etc. Depending on Twitter data or pre-trained data would result in a bias toward tweets on which the model has not been trained. Hence, taking our research ahead we propose a robust way of fact-checking the tweets via news API as well as Twitter API now known as X. This would definitely help us in getting a higher confidence and also provide us with a support level for the entered tweet.

We are initially examining a single domain/type of false tweet. The classification of tweets will be based on different parameters such as name, post, activities, followers, following, account date creation, and others. The system will use the BERTweet algorithm, which will compare the suspected accounts with the set of standard accounts.

In phase 1 of our study, we will validate disaster-related tweets. Later, a second domain of sports-related tweets will be added. By using a domain-based approach, the accuracy of fake tweet detection will be much higher when compared to a universal domain.

First, the domain is selected, and then the tweet that needs to be validated is fed. Then, the tweets are processed to be cleaned. These cleaned tweets are then put into the BERTweet model. The BERTweet model is already trained on datasets from Kaggle. This model is continuously trained to improve accuracy. The output from the BERTweet model is also checked using the Worldwide News API.

Using this API, the validity of a tweet is checked across popular news websites. If the text in the tweet is found in trusted news sources, the tweet is considered factual. If the text is not found on a trusted news website, the entered text is considered non-factual.

3.1 Categorization

The output from the pre-trained BERTweet model is categorized into three classes -

- Below 25% - Discarded directly
- Above 25% - Verified on Twitter and Worldwide News API
- Above 75% - Factual tweet

The above percentages are a threshold that we can adjust according to the requirements. The flow of the work will be as follows:

Below 25% - If the output from the BERTweet for the entered tweet or the concerned tweet received a score of less than 25%, then it will be discarded directly. Because the model predicts a score of less than 25%, the tweet is considered non-factual or fake. If the score is less than 25%, the model does not test the tweet's authenticity using the Worldwide News API. This helps in reducing the processing time and results in a much faster user experience.

Above 25% - If the entered tweet has a score above 25%, then further analysis will be required. The text in the tweet will be checked on Twitter. This ensures that the entered text is a tweet and not any random text. Here, we score the Twitter response match.

1. If the Twitter response match is less than or below 25%, the tweet is considered a non-factual/fake tweet.
2. If the Twitter response match is above 75%, the tweet is considered to be factual. These data are stored to re-train the model. As the score from the Twitter model was above 75%, it was not checked on Worldwide News API.
3. If the Twitter response match is between 25% and 75%, it searches throughout the internet on multiple news websites for similar articles. If the text in the tweet is found in news articles, it is considered factual. If the text is not found on any popular news website, the tweet is considered non-factual or fake.

Using these checks, the legitimacy of the entered tweets may be verified with significantly greater precision than with the standard BERT framework. With constant feedback, the accuracy of the system can be enhanced accordingly. This would make the tweets easier to trace. We will be able to identify and discard the concerned tweet more efficiently. This will save users and companies a lot of trouble.

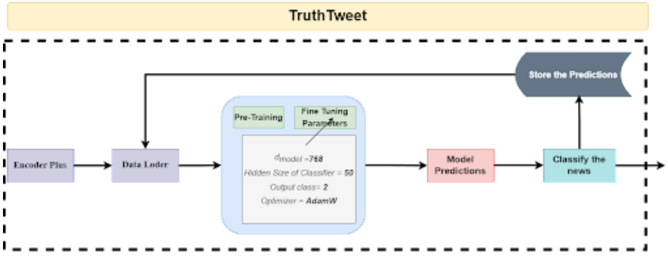

Fig. 1. TweetTruth Architecture

3.2 Acquisition of data

Since we are focusing on disaster management, the collected dataset must be accurate and diverse. With the acquisition out of the way, it is important to read the dataset and pre-process it for further operations that will be executed with the help of Pandas. Pandas is a Python package mainly used for data analysis and associated manipulation of tabular data in data frames.

Furthermore, the dataset is divided into two classes of samples to achieve a diversity of tweet types. One with no real disaster and the other with a real disaster. This will be useful for segregation and training. The acquired and pre-processed dataset will be sent for further clarification.

As far as training is concerned, it is crucial to achieve satisfactory speed in the training model. Hence, we propose using Pytorch to train a model, as it is inadequate to only use the computing power of the CPU. For this, we set the device variable to "CUDA" (GPU), which will make the model train faster.

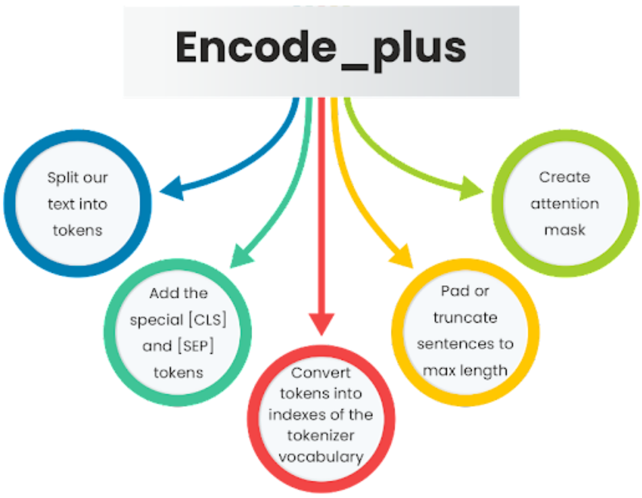

Fig. 2. Steps performed by encode_plus[23]

This is a five-step process. The below figure shows the steps that are used.

1. This will return the input IDs and attention masks. To calculate the max length of the tweets, we use the encode method and set the max length variable to the maximum length of the sentences after adding special tokens.
2. The above-mentioned tokenization and input formatting function is used to tokenize (X_train) and validate (X_val) data of tweets.

3.3 Dataset Iterator

The Pytorch DataLoader class will be used to create an iterator of the dataset to help save memory during training and boost the training speed. Labels will be converted to torch tensors with a batch size of 32.

Next, the tokenized train input IDs are converted, attention masks, and train labels to the DataLoader class using RandomSampler to randomize drawing samples from the dataset. The same operation is performed for data validation. In the end, this step will provide the train_dataloader and val_dataloader.

3.3.1 Creation of the Bert Model

The BERT model is created using the class BertClassifier. We specified the hidden size of BERT to 786, the hidden size of the classifier to 50, and the number of output classes to 2. Initialization of a one-layer feed-forward classifier, this class consists of 1 Linear input layer, Activation function as RELU, and 1 Linear output layer. Next, a forward function is added which takes input as input IDs and attention masks, feeds the same to the BERT model, extracts the last hidden state of the [CLS] token provides it to the classification model, and returns the probabilities.

3.3.2 Initialization of the Parameters of BERT

Below are the steps used for initializing the parameters of BERT

1. Initializing the optimizer using AdamW, learning rate, and default epsilon value.
2. Defining the training steps as the length of train_dataloader times the epochs.
3. Finally, define the learning rate scheduler using

3.3.3 Handling class imbalance

To handle class imbalance, the study uses sklearn's compute_class_weight method using the class_weight param as balanced. Next, convert the class weights to tensor objects and then move them to the GPU. These computed class weights are used in our model.

Defining the training loop

For each epoch, the model is put into the training mode using model. train(). For each step and batch in train_dataloder, the following steps are performed.

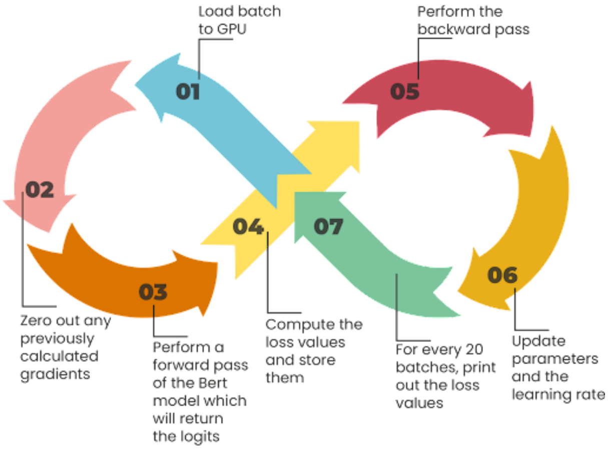

Fig. 3. Steps performed in train_dataloader

After the above steps are performed, the average loss for the training data is calculated. Next, the model is evaluated using the evaluation method defined below. If the calculated validation loss is less than the previous validation loss, the model is saved.

3.3.4 Defining the Evaluation Method

In this step, the mode is put into the evaluation mode using the model. eval(). For each batch in the val_dataloader, the following steps are performed.

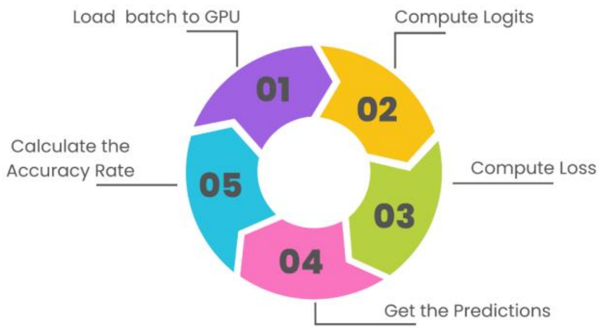

Fig. 4. Steps are performed for each batch in the val_dataloader during an evaluation

Finally, the average accuracy and loss over the validation set are computed.

Train the Model

For the training model, we set the epochs to 5. Next, the BERT model is initialized. The training will then process the data and take some time to compute.

3.3.5 Defining the Predict Method

The prediction step is similar to the evaluation step. A forward pass is performed to compute logits and apply softmax to calculate probabilities.

The model is then put into the evaluation mode using model. eval(). For each batch in the val_dataloader, the following steps are performed.

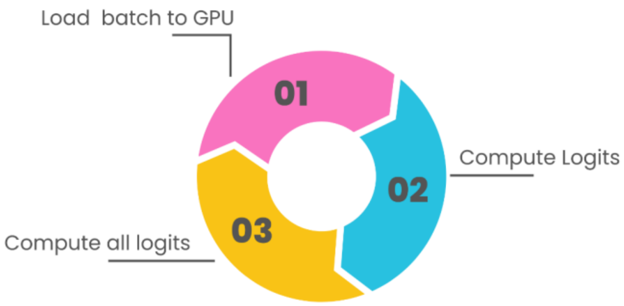

Fig. 5. Steps performed for each batch in val_dataloader during prediction

Next, logits from each batch are concatenated and softmax is applied to calculate the probabilities.

3.3.6 Saving the model

The above-mentioned prediction method is used to determine the accuracy AUC, and ROC curve of the above model. Once the results are satisfactory and meet the accuracy standards, the model is saved for inference.

3.4 Algorithm

Input: tweet text **Output**: tweet type

1. Enter the tweet text
2. Predict the type of text from the BERTweet model with confidence a
3. If a > 25%:
   a. Search the same tweet on Twitter using Twitter API.
   b. Use the match tweet procedure (defined below) to match the entered tweet and tweets returned by Twitter API.
   c. get match percent as m_pc
   d. If m_pc > 75%:
      i. Add the entered tweet to further train the BERTweet
      ii. model
      iii. Mark the entered tweet as Factual
      iv. end
         a. else if m_pc < 75% and m_pc > 25%:
            i. Search the entered tweet on WorldWideNews API.
               1. If the tweet is found:
                  a. Mark the entered tweet as Factual
                  b. end
               2. else if not found:
                  a. Mark the entered tweet as Non-Factual
                  b. end
         b. else if m_pc < 25%:
            i. Mark the entered tweet as Non-Factual
            ii. end
4. else a < 25%:
   a. Mark the entered tweet as Non-Factual
   b. end
5. procedure match_tweet
6. For each tweet in the tweets file do
   a. use Levenshtein Distance to get the distance between the entered tweet and the tweet from the file and save in a list
   b. use the top match and return the distance
   c. end for
7. end procedure

---

4. Performance Metrics for the Proposed vs Existing Model:

4.1 Table with accuracy and other parameters for BERT (vanilla BERT) and BERTweet.

Output Network (Model)	Accuracy	Precision	Recall	F1 Score	AUC
Vanilla BERT	0.8657	0.66	0.785	0.717	**0.931**

BERTweet	**0.885**	**0.675**	**0.795**	**0.719**	0.9254
Passive Aggressive Classifier	0.843	0.62	0.49	0.54	0.71
GPT2	0.45	0.22	0.78	0.34	0.59

Fig 6. Table with accuracy and other parameters for BERT (vanilla BERT) and BERTweet.

The output for every entered text will be different as the confidence and Twitter match scores vary for every tweet.

4.2 Comparison results for the BERT and BERTweet models.

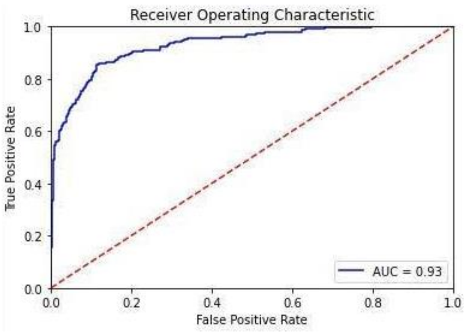

Fig 7.1 Comparison results for the BERT and BERTweet models.

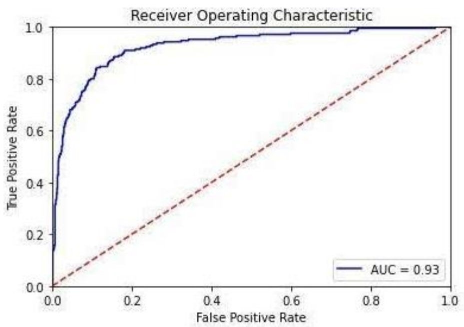

Fig 7.2 Comparison results for the BERT and BERTweet models.

5. Conclusion

Nearly 3,50,000 tweets are generated every minute [18]. The amount of fake news that may spread in a matter of minutes is enormous because of the quick flow of data. This study aims to find fake news tweets among the millions of tweets that are posted every day on the internet by using the BERTweet framework and the worldwide news API. When combined with the worldwide news API, the BERTweet model is much better at detecting fake news tweets than the standalone BERT model. The domain-based way of finding these tweets helps to improve accuracy. The goal of this study is to find these fake tweets and give users a way to check if any tweet on the internet is real.

References

[1] U. Kamath, K. L. Graham, and W. Emara, “Bidirectional encoder representations from Transformers (Bert),” Transformers for Machine Learning, pp. 43–70, 2022. doi:10.1201/9781003170082-3

[2] Ahmed H, Traore I, Saad S (2017) Detection of online fake news using N-gram analysis and machine learning techniques. In: International conference on intelligent, secure, and dependable systems in distributed and cloud environments. Springer, Cham, pp 127–138.

[3] Reema A, Kar AK, Vigneswara Ilavarasan P (2018) Detection of spammers in Twitter marketing: a hybrid approach using social media analytics and bio inspired computing. Information Systems Frontiers 20(3):515–530.

[4] Shu K, Wang S, Liu H (2019) Beyond news contents: The role of social context for fake news detection. In: Proceedings of the twelfth ACM international conference on web search and data mining, pp 312–320. ACM.

[5] Munandar D, Arisal A, Riswantini D, Rozie AF (2018) Text classification for sentiment prediction of social media dataset using multichannel convolution neural network. In: 2018 International conference on computer, control, informatics and its applications (IC3INA). IEEE, pp 104–109.

[6] Tenney I, Das D, Pavlick E (2019) BERT rediscovers the classical NLP pipeline. In: Proceedings of the 57th annual meeting of the association for computational linguistics.

[7] Rohit Kumar Kaliyar;Anurag Goswami;Pratik Narang; (2021). FakeBERT: Fake news detection in social media with a BERT-based deep learning approach. Multimedia Tools and Applications.

[8] Tenney I, Das D, Pavlick E (2019) BERT Rediscovers the Classical NLP Pipeline ICAART 2022 - 14th International Conference on Agents and Artificial Intelligence.

[9] Walker, Mason; Matsa, Katerina Eva (2021-09-20). "News Consumption Across Social Media in 2021". Pew Research Center's Journalism Project. Retrieved 2021-11-03.

[10] Castells, Manuel (2004). The Network Society. doi:10.4337/9781845421663. ISBN 9781845421663.

[11] NCRB data: 214% rise in cases relating to fake news, rumours | India News,The Indian Express

[12] https://medium.com/swlh/fake-news-detection-using-machine-learning-69ff9050351f.

[13] A. Gupta and P. Kumaraguru, “Credibility ranking of tweets during high impact events,” in Proceedings of

the 1st Workshop on Privacy and Security in Online Social Media, ser. PSOSM ’12. New York, NY, USA: ACM, 2012, pp. 2:2–2:8. [Online]. Available: http://doi.acm.org/10.1145/2185354.2185356

[14] [S. Mohd Shariff, X. Zhang, and M. Sanderson, “User perception of information credibility of news on twitter,” in Proceedings of the 36th European Conference on IR Research on Advances in Information Retrieval - Volume 8416, ser. ECIR 2014. New York, NY, USA: Springer-Verlag New York, Inc., 2014, pp. 513–518. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-06028-6 50

[15] S. Sikdar, S. Adali, M. Amin, T. Abdelzaher, K. Chan, J. H. Cho, B. Kang, and J. O’Donovan, “Finding true and credible information on twitter,” in 17th International Conference on Information Fusion (FUSION), July 2014, pp. 1–8.

[16] C. Castillo, M. Mendoza, and B. Poblete, “Information credibility on twitter,” in Proceedings of the 20th International Conference on World Wide Web, ser. WWW ’11. New York, NY, USA: ACM, 2011, pp. 675– 684. [Online]. Available: http://doi.acm.org/10.1145/1963405.1963500.

[17] M. Mendoza, B. Poblete, and C. Castillo, “Twitter Under Crisis: Can We Trust What We RT?” in Proceedings of the First Workshop on Social Media Analytics, ser. SOMA ’10. New York, NY, USA: ACM, 2010, pp. 71–79. [Online]. Available: http://doi.acm.org/10.1145/1964858.196486

[18] Z.-H. Zhou, “A brief introduction to weakly supervised learning

[19] A. Zubiaga, G. W. S. Hoi, M. Liakata, R. Procter, and P. Tolmie, “Analysing How People Orient to and Spread Rumours in Social Media by Looking at Conversational Threads,” PLoS ONE, pp. 1–33, 2015. [Online]. Available: http://arxiv.org/abs/1511.07487

[20] T. Mitra and E. Gilbert, “CREDBANK: A Large-Scale Social Media Corpus With Associated Credibility Annotations,” International AAAI Conference on Web and Social Media (ICWSM), 2015. [Online] Available:http://www.aaai.org/ocs/index.php/ICWSM/ICWSM15/paper/ view/10582

[21] M. Young, The Technical Writer’s Handbook. Mill Valley, CA: Uni C. Silverman, L. Strapagiel, H. Shaban, E. Hall, and J. Singer-Vine, “Hyperpartisan Facebook Pages Are Publishing False And Misleading Information At An Alarming Rate,” oct 2016. [Online]. Available: https://www.buzzfeed.com/craigsilverman/partisan-fb-pages-analysiversity Science, 1989.

[22] Nanade, Archana, and Arun Kumar. "Combating Fake News on Twitter: A Machine Learning Approach for Detection and Classification of Fake Tweets." International Journal of Intelligent Systems and Applications in Engineering 12.1 (2024): 424-436.

[23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.

[24] Nanade, Archana and Bhatnagar, Divya and Kumar, Arun, A Survey on Information Retrieval System for Authenticating a Tweet (March 14, 2019). Proceedings of International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM), Amity University Rajasthan, Jaipur - India, February 26-28, 2019, Available at SSRN: https://ssrn.com/abstract=3352356

[25] Gandhi, H., Bachwani, R., Nanade, A. (2023). Detecting Toxic Comments Using FastText, CNN, and LSTM Models. In: Singh, M., Tyagi, V., Gupta, P., Flusser, J., Ören, T. (eds) Advances in Computing and Data Sciences. ICACDS 2023. Communications in Computer and Information Science, vol 1848. Springer, Cham. https://doi.org/10.1007/978-3-031-37940-6_20

[26] GitHub - scikit-learn/scikit-learn: scikit-learn: machine learning in Python

[27] https://github.com/VinAIResearch/BERTweet

[28] https://www.kaggle.com/datasets/vstepanenko/disaster-tweets

[29] https://www.internetlivestats.com/twitter-statistics/