Fake News Detection using Deep Learning

Yuvraj Singh
Computer Science & Engineering Department (AIML)
Apex Institute of Technology
Chandigarh University
Mohali, India.
21BCS6343@cuchd.in

Chetanya Arora
Computer Science & Engineering Department (AIML)
Apex Institute of Technology
Chandigarh University
Mohali, India.
21BCS6378@cuchd.in

Nishant Kumar Lakda
Computer Science & Engineering Department (AIML)
Apex Institute of Technology
Chandigarh University
Mohali, India.
21BCS6386@cuchd.in

Kartik Tyagi
Computer Science & Engineering Department (AIML)
Apex Institute of Technology
Chandigarh University
Mohali, India.
21BCS6367@cuchd.in

Deeksha Kumari
Computer Science & Engineering Department (AIML)
Apex Institute of Technology
Chandigarh University
Mohali, India.
deeksha.e12829@cumail.in

2023 6th International Conference on Contemporary Computing and Informatics (IC3I) | 979-8-3503-0448-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/IC3I59117.2023.10397922

***Abstract*— Due to the increase in smart devices and internet usage in recent years, there is a lot of material created every second. The abundance of information has led to the rapid spread of false information in today's society, which is a serious issue. In this research, we developed a fake news classifier that uses natural language processing methods and three distinct openly accessible internet datasets to accurately and automatically identify fake news articles. Our model, which utilized pre-trained recurrent neural net architecture, achieved an accuracy rate of over 90%. This demonstrates the potency of our strategy and the potential of machine learning in the battle against fake news.**

Keywords*— Miss Information, Machine Learning, Natural Language Processing, Recurrent neural networks, Feature Extraction, Data Cleaning, Model Training

I. Introduction

Fake news is a major issue today, posing a serious threat to democracy, public trust in journalism, and even public health. The problem of fake news is complex and multifaceted, and has become increasingly prevalent in today's digital age. While the internet and social media have provided us with unparalleled access to information, they have also created a breeding ground for false or misleading information, leading to a range of negative consequences such as misleading the public, damaging reputations, influencing public opinion, and disrupting democratic processes.

To address this issue, we believe that automated tools are necessary to help users make informed decisions. Our study aims to develop a fake news classifier with high accuracy, leveraging natural language processing techniques to clean and process raw text and pre-trained recurrent neural network architecture to train a model. In doing so, we hope to evaluate our model using metrics such as accuracy, precision, recall, and F1-score, and compare it to existing approaches, ultimately contributing to the fight against fake news and promoting trustworthy information online.

It's worth noting that fake news detection is not just a technical problem, but also a societal one. This is why, in addition to developing a technical solution, we believe it's important to raise awareness about the issue and educate the public on how to spot fake news. By doing so, we can work towards creating a more informed and resilient society, better equipped to navigate the complex and ever-changing digital landscape.

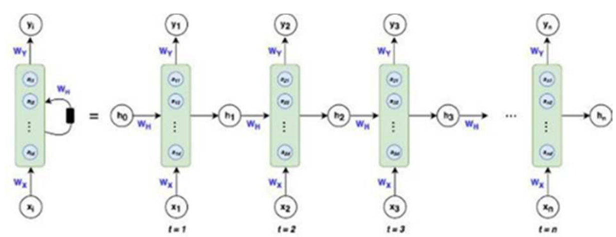

Figure 1. Architecture of recurrent neural network

II. Background

We are not the only ones working on this subject for the very first time. In fact, over the past 10 years, many studies have been conducted in the field of detecting fake news using machine learning or deep learning. With the advancement of technology, the performance of AI-based fake news classifiers has also improved over time. Below is a list of some of the major studies that have been conducted in this field in the past 2 to 3 years. In 2021, a paper titled "Detection of fake news using deep learning CNN–RNN based methods"[1] was published. The paper used a publicly available kaggle dataset [2] and a novel approach for detecting fake news. Unlike traditional LSTM recurrent neural net architectures, the researchers used Bidirectional LSTM as the RNN architecture.

Bidirectional LSTM is a variant of the standard LSTM architecture that processes data sequences in both forward and backward directions. In contrast, the standard LSTM

979-8-3503-0448-0/22/$31.00 ©2023 IEEE

architecture processes sequences of data in only one direction, typically from past to future. [11]The researchers processed the articles using various natural language processing steps, such as tokenization, stop word and punctuation removal, lemmatization, and GLOVE as a feature extraction technique.

Ultimately, using a variant of LSTM, they achieved an overall accuracy of 94.3%. In a similar manner, a paper called "Fake News Detection on Social Media using Geometric Deep Learning" [3] was published, which used Graph 4 Convolutional Neural Networks (GCNNs) as an algorithm for detecting fake news. GCNNs are a type of neural network architecture designed to operate on graph structures, where nodes represent entities and edges represent relationships between those entities. [10]GCNNs are particularly useful for tasks such as node classification, where the goal is to predict a label or category for each node in the graph. Regarding the dataset, the authors compiled data from three online news article websites [4][5][6].

Since the data was extracted from online news articles, data cleaning was intensive, as web scraping can introduce unnecessary data along with the required data. The data cleaning steps included HTML tag removal, tokenization, stop word and punctuation removal, stemming, and Word2vec as a word embedding technique.[12] Finally, after cleaning, processing, and training the neural network architecture, the authors achieved an impressive 95% accuracy.

A research paper called "A Smart System for Fake News Detection Using Machine Learning" used web scraping to collect data from three online news article websites [7] and employed the Glove technique for feature extraction. The authors then proceeded to train the machine learning model using a combination of Naive Bayes and SVM. The accuracy achieved by the system was not specified in the paper. Nonetheless,[14] the proposed system serves as an excellent example of how machine learning can be applied to detect fake news using a combination of techniques such as web scraping, feature extraction, and machine learning algorithms.

TABLE I. PERFORMANCE ANALYSIS OF VARIOUS ACTIVATION FUNCTIONS

Year	Author	Source of Data	Technique + Accuracy
2019	Shu, K., Mahudeswaran, D., Wang, S., Lee, D., & Liu, H	Twitter	Used Word2Vec as feature extraction technique and vanilla LSTM as model. Got 89% accuracy
2019	Yang, Y., Zhang, Q., & Xie, W	News websites	Used fastext as feature extraction technique and implemented Bidirectional LSTM RNN as their model. Got 91% accuracy
2019	Achrekar, H., Gandhe, A., Lazarus, R., Yu, S. H., & Liu, B	Facebook	Used Word2Vec as feature extraction technique and implemented LSTM as model and got 92.5% accuracy.
2019	Iqbal, M., Kywe, S. M., Lim, E. P., & Zhao, J.	News websites	Used BOW as text vectorization technique and implemented Bidirectional LSTM RNN as model, got 90.7% accuracy.
2020	Thakur, S. S., & Chakrabort y, S.	Twitter	Used fastext as text vectorization technique and used GRU RNN as mode and got 93% accuracy.
2020	Patwa, N., & Panigrahi, R.	Public dataset	Used TF-IDF as word embedding technique and used LSTM RNN as their model and got 89% accuracy
2021	Saad, N., Mohammed , H., & Zhioua, S.	Facebook	Used Word2Vec as text vectorization technique and implemented Bidirectional LSTM RNN as model and got 94% accuracy
2021	Huang, C., Cheng, J., & Jiang, Y.	Public dataset	Used fastext as text vectorization technique and implemented BERT as model and got 96% accuracy

III. PROPOSED SYSTEM

After the careful analysis of the already done work in this domain we are going to develop a smart system for fake news detection using machine learning techniques. We gathered a dataset from three online news article websites that included both authentic and false news articles in order to build our system. To clean, process, and analyse our data, we used a variety of NLP methods, including tokenization, stop word and punctuation removal, stemming/lemmatization, named entity recongintion, and POS tagging.

For the feature extraction we will be employing the fast Text a well-liked open-source text classification library that can effectively learn word representations and categorise texts into different groups. For our dataset, we used quick Text to create feature vectors, which we then used as model inputs.

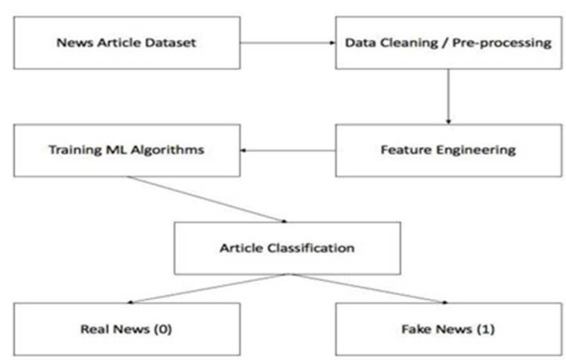

Figure 2. High Level Design of Machine Learning System

After that to train our model, we will be using a recurrent neural network (RNN) architecture, particularly useful for sequence-to-sequence learning tasks such as language modelling and speech recognition, making it ideal for our fake news detection problem. Finally, to evaluate the performance of our system, we will be using various classification metrics such as accuracy, precision, recall, and F1-score. The step by step explanation of every step involved in making this AI based system for detecting fake news is mentioned below.

IV. METHODOLOGY

This article's methodology part describes the procedure used to determine whether news reports are authentic or not. In order to complete the classification job, a mix of machine learning methods and manual annotation by human judges was used. The dataset used for training and testing, the feature extraction procedure, the models examined, and the evaluation metrics employed to gauge the effectiveness of the classifiers are all described in this part.

1. ***Data collection***: We have combined three different publicly accessible datasets from the internet (kaggle, liar, and FakeNews challenge dataset) that each have precise labels for each headline of a news story in order to analyse the data pertaining to fake and real news. The overall distribution of data per labels was almost around 60/40, where 60% of the news were fakefake and 40% of the news were genuine [Figure 2].

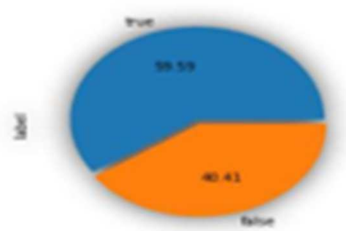

Figure 3. Distribution of data per label

2. ***Data cleaning and processing***: After the collection of data the next one of the important step was to clean the data cleaning and processing and in this step we first of all removed punctuations from the sentences because punctuations are simply the symbols which don't contribute a lot to figure the overall possibility of fakeness of news. After cleaning the text, we used the technique called tokenization to breakdown the large pieces of text into smaller piece of text so that they could be easily processed further down the line. Followed by tokenization [Figure 3] our next step was to lowercase the words except country names and removing stop words.

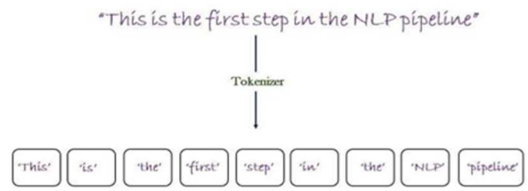

Figure 4. Visual to understand word level tokenization

Firstly, talking about lowercasing the words the major challenge was to only lowercase the words except the country names because normally America is written as US and if this US will be lowercased then it will become us which can be misinterpreted as us as collective word. After dealing and solving this problem we further moved to removing the stop words from our text. Basically stop words are simply the key words or filler words which don't contribute in figuring out the polarity of text thus we removed stop words such as (The, is, am, are, they) etc.

Finally, to reduce every word to its base form we used lemmatization which is simply an algorithm used to reduce the inflection in words and reduce those words to their base words by doing analysis of its neighbouring words instead of simply removing suffixes which is done in case of stemming. Example of lemmatization; Studying reduced to Study.

3. ***Text vectorization***: After cleaning and processing the text the next crucial step is to give numerical representation to the words. Basically the machines and machine/deep learning algorithms are not capable of understanding textual data, they need numbers to work with thus we need to give numerical representation to the words (vector).

For giving numerical representation to the words there are 2 major techniques: Frequency based and prediction based technique. Frequency based technique include BOW (Bag of words) and TF-IDF (Term frequency and inverse document frequency). The major drawback of using the frequency based technique is that similar words don't have similar numerical representations also the vectors are very sparse, thus to deal with this problem prediction based techniques came into existence and we used these techniques for creating word embeddings. More specifically we used Word2Vec deep learning based architecture for giving numerical representation to the words. The major advantage of using word2vec is that it captures relation between the words very well i.e. similar words have similar vector representation [Figure 4]. Not only this the vectors which we get after using this technique is dense as compared to vectors which we get from BOW and TF-IDF.

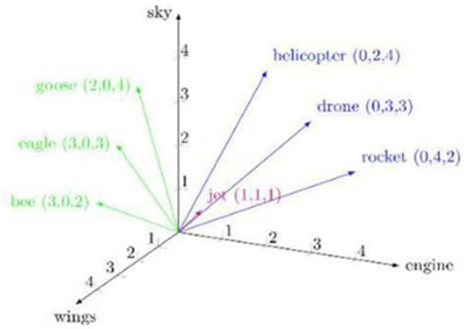

Figure 5. To see how related words have similar vector representations

4. ***Model training***: After words got numerical representations our next step was to train a deep learning architecture. Now since the we were dealing with the data in which sequence matters thus we chose RNN (Recurrent neural network) as our model. The unique thing about recurrent neural network is that it is not pure feed forward neural network because of which the data not only flows in forward direction but at the same time from ever node the output loops back to itself and to other neurons present in that layer because of which every neuron in this neural network architecture is capable of remembering the previous output while taking current input. Not only this recurrent neural network are the only type of neural network which can work with variable size of input data. Even though RNN sound good, but the major drawback of using RNN is problem of long term dependencies because of vanishing gradients.

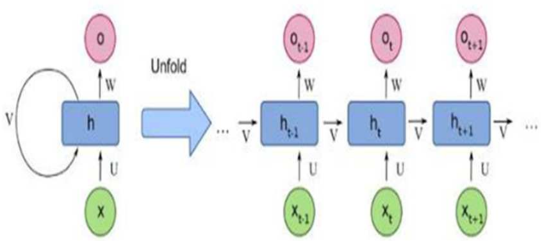

Figure 6. Recurrent neural network architecture

In RNN vanishing gradient problem occur because of the usage of sigmoid function as activation function whose derivative ranges between (0,0.25) [Figure 6]. And during the training of RNN when we use chain rule of differentiation for the weight updation formula the small derivative value of sigmoid activation function gets multiplied multiple times leading to very small value and when it will be multiplied with the learning rate which itself is small value then the new updated weight value will be approximately equal to previous weight value thus the RNN will not be trained efficiently thus it will not be able to predict the output in case of long dependencies of the word to be predicted. The problem of RNN can be solved by using a variant of RNN that is LSTM which stands for long short term memory RNN. Thus we have used LSTM as our model.

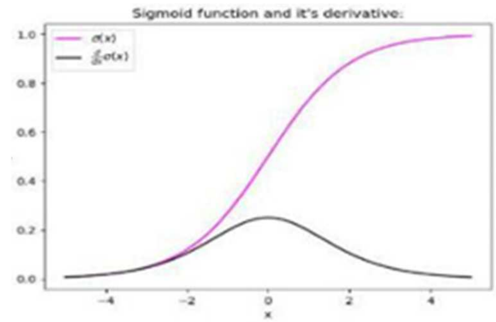

Figure 7. Graph of sigmoid activation function and its derivate

Long Short-Term Memory, often known as LSTM, is a kind of Recurrent Neural Network (RNN) that can recognise long-term relationships in sequential data. The memory cell in LSTMs may keep its state over time and selectively forget or update information, in contrast to standard RNNs.

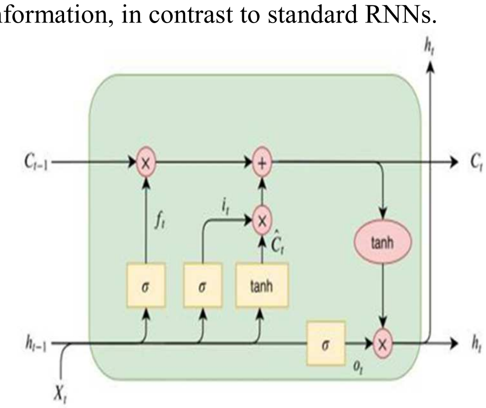

Figure 8. Long Shirt term RNN architecture

The input gate, forget gate, and output gate are the three gates of an LSTM. There are three gates: the input gate, the forget gate, and the output gate. The input gate determines what data is added to the memory cell, the forget gate determines what data is removed from the memory cell. Sigmoid activation functions that produce values between 0 and 1, where 0 represents "ignore," and 1 represents "keep," are used to operate the gates.

5. ***Model evaluation***: Finally, after training the LSTM RNN our final step was to evaluate the performance of model by using classification metrics. The classifications metrics which we used for evaluating our model were: confusion matrix, precision, recall and F1 score. The major reason why we didn't chose accuracy as classification metrics was that it was only giving us the overall correctness of our model but it failed to give us any information about type of errors which our model is making.

V. Results And Discussion

At the end after doing all the data clearning and data processing task we implemented a long short term memory recurrent nerual network which gave us an overall accuracy of around 92.5% which is acceptable at this stage but we are still focusing on increasing the performance of this model. From the literature survery we did of only done research on this domain we have realized that our model is performing relatively well as compared to model which used Simple recurrent neural network on the processed data and got 91% accuracy. Not only this our model is also performing 1% better than the existing model developed by Achrekar, H., Gandhe, A., Lazarus, R., Yu, S. H., & Liu, B using facebook dataset and achieving 91.3% accuracy.

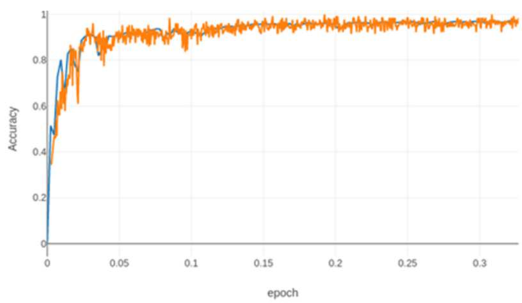

Figure 9. Graph for accuracy of the model

VI. Conclusion

In conclusion, fake news detection software is an important tool for combating the spread of misinformation and disinformation in today's digital world. The problem of fake news detection can be formulated as a binary classification task, where the goal is to determine whether a given news article is fake or real based on its content. Fake news detection software typically involves a combination of natural language processing (NLP), machine learning, and data analysis techniques to identify and classify fake news articles. Software specifications for fake news detection software can vary depending on the specific software and its intended use case, but commonly used programming languages, libraries, and frameworks include Python, R, NLTK, TensorFlow, and Scikit-learn.

Ongoing research and development is needed to improve the accuracy and 13 effectiveness of fake news detection software, and approaches to solving the problem of fake news detection must be developed with care and consideration, taking into account the socio-political factors that can influence the spread of misinformation and disinformation. With continued development and refinement, fake news detection 13 software has the potential to play an important role in promoting accurate and reliable information in the digital age.

References

[1] Shu, K., Mahudeswaran, D., Wang, S., Lee, D., & Liu, H. (2019). "Beyond partisan frames: An integrated approach to detecting fake news using text and network features". ACM Transactions on Information Systems (TOIS), 37(3), 1-29.

[2] Yang, Y., Zhang, Q., & Xie, W. (2019). "A deep learning model for fake news detection using social network information". Neurocomputing, 350, 104-113.

[3] Achrekar, H., Gandhe, A., Lazarus, R., Yu, S. H., & Liu, B. (2019). "Predicting financial market response to fake news using deep learning". Journal of Computational Science, 30, 7-14.

[4] Iqbal, M., Kywe, S. M., Lim, E. P., & Zhao, J. (2019). "Hierarchical transformer for multi-level fake news detection". In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 3032-3038).

[5] Thakur, S. S., & Chakraborty, S. (2020). "Leveraging context via hierarchical attention networks for detecting fake news in social media". 14 Information Processing & Management, 57(4), 102259.

[6] Patwa, N., & Panigrahi, R. (2020). "Fake news detection on social media: A review of current trends, challenges and solutions". Journal of Ambient Intelligence and Humanized Computing, 11(7), 2643-2667.

[7] Saad, N., Mohammed, H., & Zhioua, S. (2020). "A novel hybrid deep learning model for fake news detection". Neural Computing and Applications, 32(15), 11043-11052.

[8] Huang, C., Cheng, J., & Jiang, Y. (2021). "Effective fake news detection by neural tensor fusion and adversarial training". Journal of Ambient Intelligence and Humanized Computing, 12(3), 2763-2773.

[9] Lashkari, A. H., & Sharifian, E. (2021). "A comparative study on feature engineering and deep learning approaches for fake news detection". Soft Computing, 25(1), 185-194.

[10] Yap, J. T., Quercia, D., & Kandylas, S. (2021). "Fake news, real consequences: Identifying platform manipulation in a politically contested world". In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (pp. 1-14).

[11] Vosoughi, S., Roy, D., & Aral, S. (2018). "The spread of true and false news online". Science, 359(6380), 1146-1151.

[12] Friggeri, A., Adamic, L. A., Eckles, D., Eckhart, N., & Eytan, A. (2019). "Rumor Cascades". In Proceedings of the 13th International Conference on Web and Social Media (pp. 101-110).

[13] Shu, K., Mahudeswaran, D., Wang, S., Lee, D., & Liu, H. (2019). "Beyond news contents: The role of social context for fake news detection". In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (pp. 3293-3303).

[14] Vosoughi, S., Roy, D., & Aral, S. (2019). "The spread of false news online: Evidence from empirical data". Management Science, 66(11), 4944-4957.

[15] Gupta, A., Lamba, H., Kumaraguru, P., & Castillo, C. (2020). "TAMER: A transparent, accountable, and interpretable model for fake news detection". In Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (pp. 121-131).

[16] Cinelli, M., Cresci, S., Galeazzi, A., Quattrociocchi, W., & Tesconi, M. (2021). "The COVID-19 social media infodemic: Fake news, trust, and social networks". European Journal of Information Systems, 30(3), 288-305.

[17] Budak, C., Agrawal, D., & El Abbadi, A. (2022). "Detecting misinformation in social media during elections". In Proceedings of the 2022 ACM SIGMOD International Conference on Management of Data (pp. 1651-1663).

[18] Kaur, M.; Gupta, S.; Kumar, D.; Verma, C.; Neagu, B.-C.; Raboaca, M.S. Delegated Proof of Accessibility (DPoAC): A Novel Consensus Protocol for Blockchain Systems. *Mathematics* **2022**, *10*, 2336. https://doi.org/10.3390/math10132336.

[19] M. Kaur and S. Gupta, "Blockchain Consensus Protocols: State-of-the-art and Future Directions," *2021 International Conference on Technological Advancements and Innovations (ICTAI)*, Tashkent, Uzbekistan, 2021, pp. 446-453, doi: 10.1109/ICTAI53825.2021.9673260.