Fake News Detection Using Machine Learning Algorithm

Ms. Adlin Belshiya (AP/IT), Ajitha Kumari N, Mahumood Sameema S, Manju P S
Department of Information Technology, Arunachala College of Engineering for Women

Abstract- The rapid development of digital communication platforms and social media sites has dramatically changed the pattern of information generation, dissemination, and consumption. This, in turn, has resulted in an increase in fake news dissemination, which is a serious threat to society. Fake news is a serious threat to society and affects various aspects of life, including public opinion, political manipulation, dissemination of misinformation during critical times, and social stability. Thus, there is a need to develop efficient and reliable fake news detection systems, and this is an important research area.

In this paper, an efficient fake news detection system is proposed, which is based on Natural Language Processing and Machine Learning algorithms. This proposed system utilizes a Doc2Vec model, which is a distributed representation model of words and phrases, to convert text data into dense vector representations, which capture semantic and contextual relationships between words in a text document. This vector representation of text data is further utilized to classify fake and real news articles using a Support Vector Machine algorithm.

In addition to classification, this system proposes a cosine similarity-based consistency analysis between the headline and body of the news article. This feature is useful in identifying misleading headlines, which is a key feature of fake news. Moreover, this system incorporates Explainable Artificial Intelligence (XAI) based on LIME (Local Interpretable Model-Agnostic Explanations), which provides insights into the prediction of the model by identifying the most influential words in classification. The entire system is implemented in Python, and a web application is developed using Flask, a Python framework, to enable user interaction with the system in real time. From the experimental results, it is clear that this proposed system achieves better accuracy, contextual understanding, and interpretability compared to other traditional methods.

The proposed system is reliable and efficient in identifying fake news, and this is attributed to various features incorporated in this system, such as classification, consistency, and interpretability.

Keywords- Fake News Detection, Natural Language Processing, Doc2Vec, Support Vector Machine, Cosine Similarity, Explainable AI, LIME, Machine Learning

I. INTRODUCTION

In the modern world, the internet has become the primary source of information for millions of people around the world. Social media, online news portals, and digital communication channels have provided users with the facility to access information and share it instantly. However, this instant sharing of information has also given rise to the global phenomenon of fake news, which refers to false information being shared as legitimate news.

The problem of fake news has become a significant challenge in the modern world due to its potential to impact social harmony, public opinion, and critical decision-making processes. Fake news has become a significant problem due to its potential to manipulate people, earn money through advertisements, and promote political agendas. The instant sharing of information has made it difficult to control the spread of fake news.

The traditional methods of identifying and controlling the spread of fake news include manual verification of the information. Although this has proven to be effective, it has also been found to be time-consuming. In order to control the spread of fake news, there is a growing need to develop automated systems to identify and control the spread of fake news.

Recent advances in Machine Learning and Natural Language Processing have made it possible for intelligent systems to be created that can process and analyze text data and identify patterns that are associated with fake news. However, most of these systems use multiple algorithms or rely on shallow text features, which make them complex and not interpretable. Furthermore, most of these systems are

only focused on classification and do not consider other factors such as misleading headlines and a lack of transparency in predictions.

To overcome these challenges and limitations, this paper proposes a system for detecting fake news that includes semantic feature extraction, efficient classification, consistency analysis, and explainability in a unified system. The system proposed in this paper uses Doc2Vec for efficient classification of the extracted features from the text data. Doc2Vec is a type of word2vec that can be used for generating meaningful vector representations of the text data and can identify both syntactic and semantic features in the text data. The system also uses a Support Vector Machine (SVM) for classification, as SVM is known for its efficiency in handling high-dimensional features and providing robust classification results.

Furthermore, the system incorporates a consistency analysis tool that utilizes the cosine similarity technique to measure the association between the headline and the body of the news article. This tool is effective in identifying articles that have misleading headlines that do not reflect the content of the article, a characteristic that is typical of fake news.

Another notable contribution of the study is the utilization of the Explainable Artificial Intelligence (XAI) tool that utilizes the LIME technique. This tool is effective in providing transparency by explaining the factors that influence the classification process.

The proposed system utilizes Python programming and the Flask web application platform for the implementation of the system. This allows the system to analyze the news article in real-time.

This study contributes to the field by providing a simple yet effective approach towards the detection of fake news articles. The proposed system is effective in providing an accurate and understandable approach towards the classification of news articles.

The proposed system can be expanded for real-world applications by incorporating the following features into the system: social media monitoring systems, news article verification systems, and misinformation control systems.

II. LITERATURE REVIEW

The detection of fake news has emerged as a major research domain due to the widespread occurrence of misinformation on social media platforms. Several approaches based on machine learning algorithms have been developed to tackle this issue.

Traditional approaches to detecting fake news involved machine learning algorithms such as Naive Bayes, Decision Trees, K-Nearest Neighbor (KNN), etc. These algorithms were based on statistical attributes such as word frequency. Even though these approaches were simple and computationally efficient, they were not capable of understanding the context of the text. This limited the accuracy of the results obtained from such approaches.

Recently, several approaches based on Natural Language Processing algorithms such as Word2Vec and Doc2Vec have been developed. Word2Vec generates vectors based on word relationships. This approach was found to be efficient in detecting fake news. Doc2Vec generates vectors based on document relationships. This approach was found to be efficient in detecting fake news.

Support Vector Machine (SVM) is one of the most popular algorithms for text classification problems, owing to its ability to handle high-dimensional data. Several research studies have proved that SVM achieves better accuracy in comparison to traditional classifiers by applying an efficient feature extraction mechanism. However, most existing solutions utilize a combination of classifiers, which makes them more complex without achieving any significant improvement in accuracy.

Recently, researchers have also tried to solve the problem of identifying inconsistencies between a news headline and its body. They applied a technique such as cosine similarity to calculate semantic similarity between different parts of a document. This technique is useful in identifying misleading headlines, which is one of the characteristics of fake news.

Another prominent advancement in this field is the application of Explainable Artificial Intelligence (XAI). This technique is useful in providing explanations about a prediction made by a machine

learning algorithm, such as LIME (Local Interpretable Model-Agnostic Explanations).

However, the existing systems face the following limitations: lack of interpretability, complexity, and the inability to effectively identify misleading information.

To overcome the aforementioned challenges, the proposed system incorporates a hybrid approach that utilizes the Doc2Vec technique for feature extraction and the SVM technique for classification. Additionally, the system incorporates the cosine similarity technique for consistency analysis and the LIME technique for interpretability.

III. METHODOLOGY

The proposed system for detecting fake news has been designed as a structured pipeline that includes various aspects of Natural Language Processing (NLP), Machine Learning, and Explainable Artificial Intelligence. The methodology for the proposed system includes various stages that are significant for the detection of fake news in an accurate and reliable manner.

Data Collection and Preparation

The dataset for the current study is collected from the Kaggle platform and consists of two separate files: Fake.csv and True.csv. The first file includes fake news articles, and the second file includes real news articles. The attributes for both datasets are title, text, subject, and date.

The dataset is prepared as follows: the two files are combined to create one dataset. A new column, label, is added to the combined dataset to indicate the class of the articles, where fake news is given the class 1 and real news is given the class 0. The dataset is then shuffled to include randomness and prevent bias during training.

Text Preprocessing

Text preprocessing is done to improve the quality of the input data. This includes the following steps:

- Lowercasing of text data
- Removal of punctuation marks and special characters
- Tokenization of text data
- Removal of stop words
- Lemmatization

These steps are necessary to improve the quality of the input data.

Feature Extraction using Doc2Vec

After this, the text data is transformed into numerical vectors using the Doc2Vec model. In this model, dense vectors are created for documents, considering their semantic meaning and contextual relationships.

The infer_vector() method is applied to create vector representations for each of the news articles. These vectors are then used as features in the classification model. In comparison to other techniques, such as Bag of Words and TF-IDF, the use of Doc2Vec provides better contextual understanding.

Classification using Support Vector Machine (SVM)

The classification of the news articles is done using a Support Vector Machine classifier. Support Vector Machine is a supervised learning algorithm that finds the best hyperplane to split the data points into different classes.

The vectors generated by the Doc2Vec model are given as input to the Support Vector Machine classifier. This classifier is then used to classify the input articles as real or fake. Support Vector Machine has been chosen as the classifier due to its high accuracy.

Consistency Analysis using Cosine Similarity

Apart from classification, the system also performs a consistency analysis between the headline and the body of the news article. This is done by utilizing cosine similarity. Cosine similarity is a measure used to calculate the similarity between two vectors.

The vectors representing the headline and the body of the news article are created using the Doc2Vec model. Then, the cosine similarity between two vectors is calculated. This similarity score represents the level of consistency:

- High similarity: This represents a match between the headline and the content
- Low similarity: This represents misleading or clickbait content

This step helps improve the system's ability to identify fake news.

Explainable AI using LIME

To increase transparency and interpretability, this system also incorporates LIME, or Local Interpretable Model-Agnostic Explanations. LIME is responsible for explaining the prediction of the SVM model by determining which words in the text are most influential to the prediction.

The explainer modifies the text and observes the change in prediction by the model and assigns a score to words based on their influence on the prediction, enabling users to comprehend the prediction of the model.

Model Deployment using Flask

The entire system is implemented as a web application, and it utilizes the Flask framework. The trained model is stored in a serialized form.

The user can provide the input to the system, and it is processed through all the stages of the pipeline in the backend. The final output, which includes the prediction, consistency score, and explanation, is shown to the user in real time.

Workflow Summary

The overall workflow of the system can be summarized as follows:

1. User inputs headline and article text
2. Text is pre-processed and cleaned
3. Doc2Vec converts text into vectors
4. SVM classifies the news as real or fake
5. Cosine similarity calculates consistency score
6. LIME explains the prediction
7. Results are displayed to the user

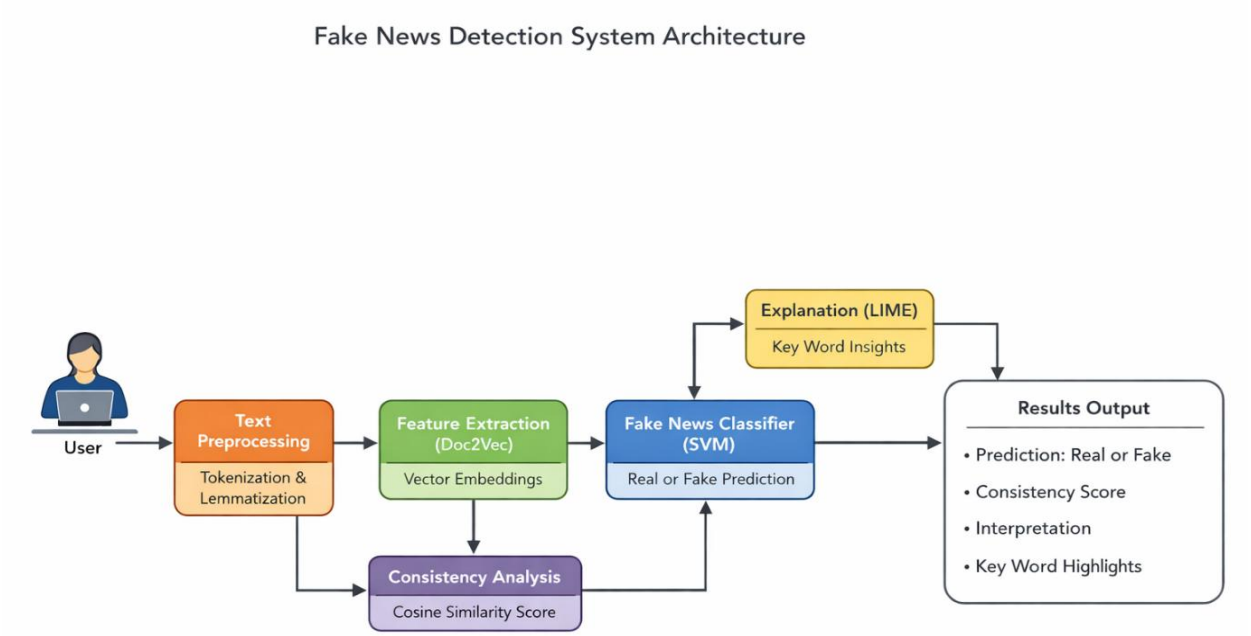
IV. ARCHITECTURE

The proposed fake news detection system utilizes a modular and pipeline-based architecture that is best for handling textual data and making predictions with high accuracy. The system starts with the input layer, which requires the user to input the news headline and the body of the news article using a web interface. The input data is then fed into the preprocessing layer, which performs a series of preprocessing tasks such as converting the text to lowercase, removing special characters and punctuation marks, tokenizing the text into words, removing stopwords, and lemmatizing the words to obtain the root word.

After the preprocessing stage, the cleaned data is fed into the feature extraction layer, which utilizes the Doc2Vec model to convert the textual data into numerical vectors that can be analyzed effectively by the system. The numerical vectors generated by the Doc2Vec model are then fed into the classification layer, which utilizes a Support Vector Machine (SVM) classification model to classify the news article as real or fake. This classification module forms the core part of the proposed system.

Parallel to this classification, the system also performs a consistency analysis using the cosine similarity. In this case, the article's title and body are transformed into vectors, and the similarity between them is computed. This would then determine whether the title of the article correctly represents the content or is misleading, which is a typical feature of fake news.

For the system to be interpretable, it has been enhanced to include an explainability layer. In this case, the LIME (Local Interpretable Model-Agnostic Explanations) algorithm has been included. LIME analyzes the prediction made by the SVM model and identifies the words that have the most impact on the classification. Lastly, the output layer simply presents the results to the user, indicating whether the article is real or fake. This design ensures that the system is not only effective but also interpretable.



V. IMPLEMENTATION

The proposed system for the detection of fake news has been implemented using Python, incorporating Natural Language Processing, Machine Learning, and Explainable Artificial Intelligence. The proposed system has been implemented in a unified framework to provide the functionality of processing the text, classifying the text, and providing results in an interpretable form.

The implementation of the proposed system has been initiated by importing necessary libraries, including NumPy, Pandas, NLTK, Gensim, Scikit-learn, and LIME. NLTK has been implemented to provide the functionality of text preprocessing. Text preprocessing has been implemented to clean the text by converting it to lowercase, removing punctuation using regular expressions, splitting the text into words, and removing stopwords and non-alphabetic words. Lemmatization has also been implemented to reduce the words to their root form.

For feature extraction, the system utilizes the Doc2Vec model provided by the Gensim library. The preprocessed text data is transformed into dense vectors using the infer_vector() function. These vectors represent the semantic meaning and contextual associations of the provided text. This same Doc2Vec model has been used to create vectors for both the training data and the user's input.

For the classification feature, the system utilizes a Support Vector Machine (SVM) model provided by the Scikit-learn library. The SVM model has been trained on the labeled data and stored in a serialized file format using the pickle module. At runtime, the model has been loaded directly from the serialized file, bypassing the need to retrain the model. The vectors generated by the Doc2Vec model for the input text have been provided to the SVM model to determine whether the news article is real or fake.

Apart from classifying, the system also calculates the consistency score between the headline and body of the article using cosine similarity. For calculating the cosine similarity, both the headline and body are preprocessed and converted to their respective vector representations using the Doc2Vec model. Then, the cosine similarity is obtained using Scikit-learn's pairwise similarity function.

The LIME (Local Interpretable Model-Agnostic Explanations) method is also incorporated into the system to increase the interpretability of the model. LIME is a technique for explaining the predictions made by the model. LIME generates explanations for the predictions made by the model by creating variations of the input and examining the predictions made by the model for the new variations. It also provides importance scores for the most influential words in the article, and the importance is shown to the user.

The system is implemented as a web application using the Flask framework. The backend processes HTTP requests and user input using routes. The home page route loads the interface and checks if the trained models are available. The analysis route processes input data in JSON format, calls the analysis function, and returns the prediction results. The frontend is implemented using HTML and Tailwind CSS for a clean and responsive interface. JavaScript is used for asynchronous requests to dynamically update the results on the webpage.

The final output displayed to the user includes the classification output (real or fake), the consistency score along with its interpretation, and the explanation of the prediction generated by LIME. This implementation ensures that the system is efficient, scalable, and able to provide accurate results in real-time.

VI. RESULTS & DISCUSSION

The proposed system for detecting fake news was implemented and tested using a set of real and fake news articles. The system was tested and evaluated according to the accuracy in classifying the news content, detecting inconsistencies in the headlines and bodies of the articles, and the ability to provide explanations for the predictions made.

The results obtained from the classification show that the proposed system combining Doc2Vec and SVM provides effective performance in distinguishing between real and fake news. The Doc2Vec model was able to capture the relationships between the words in the text, allowing the SVM classifier to make accurate predictions. The proposed system also outperforms traditional text representation methods in that it provides better results in understanding the document structure, thereby improving the classification results.

The consistency analysis conducted through the cosine similarity measure was also an important addition to the system. This measure was effective in detecting inconsistencies between the headlines and the article content. This is a characteristic feature of fake news and clickbait. The system was also able to detect articles with low similarity scores, indicating that the headline did not represent the content.

The inclusion of Explainable Artificial Intelligence through the LIME technique increases the interpretability of the model. The system gives the user a list of words and their impact on the prediction. This way, the user can easily understand why a particular article is classified as real or fake. This addresses one of the major limitations associated with traditional machine learning models, as these models are considered to be black boxes.

The system was also tested in a real-world scenario by creating a web application using the Flask framework. The application was able to process user input and provide results in real time. The results were displayed in a clear and organized manner, making it easy for users with varying levels of technical expertise to understand.

Despite the effectiveness of the system, there are a few limitations. The performance of the model depends on the dataset. Moreover, even though the Doc2Vec model can capture the meaning of words, it may not be able to capture complex linguistic patterns and sarcasm, which are sometimes seen in news articles. Similarly, even though the cosine similarity measure provides an idea about the results, it may not be entirely accurate.

Overall, the suggested system shows a balance in terms of classification, consistency, and explainability. The results show that the suggested system is capable of providing accurate, interpretable, and reliable fake news detection, and it is ready to be applied in real-world scenarios.

VII. CONCLUSION

In this paper, a complete fake news detection system has been proposed by combining the concepts of Natural Language Processing and Machine Learning approaches. The proposed system effectively handles the issues that arise from the widespread dissemination of misinformation by providing a tool that can classify news articles into real and fake news.

The proposed system uses the Doc2Vec approach for extracting features from the articles and a Support Vector Machine for classification. The classification process is enhanced by the addition of a consistency analysis tool that uses the cosine similarity approach to verify the consistency of the headline and the body of the article. This tool is useful for detecting misleading articles such as clickbait articles. The inclusion of the LIME tool provides an explanation for the prediction by highlighting the most impacting words in the article.

The usage of the system as a web application using the Flask framework also proves the practicality of the system in real-time applications. From the obtained results, it is clear that the proposed system is reliable and can be used to perform predictions, which is beneficial for applications like news verification systems and social media monitoring systems.

Even though the system is working well, there are certain limitations to the system, which can be considered to take it to the next level in the future.

The proposed system offers a balanced view of a system, where accuracy, consistency, and interpretability are considered, which is beneficial to fight against the menace of fake news in the digital world.

REFERENCES

[1] Tao Jiang, Jian Ping Li, Amin Ul Haq, Abdus Saboor and Amjad Ali, " A Novel Stacking Approach for Accurate Detection of Fake News", IEEE Access, Vol. 9, pp. 22626 – 22639, 2021

[2] Veronica Perez-Rosas, Bennett Kleinberg, Alexandra Lefevre and Rada Mihalcea, "Automatic Detection of Fake News", Proceedings of the 27th International Conference on Computational Linguistics, pp. 3391–3401, 2018

[3] Shital Lathiya and M B Chaudhari, "Rumour Detection from Social Media : A Review", International Journal of Scientific Research in Computer Science, Engineering and Information Technology, Vol. 3, pp. 384 – 389, 2018

[4] Jing Ma, Wei Gao and Kam-Fai Wong, "Rumor Detection on Twitter with Treestructured recursive Neural Networks", Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Long Papers), pp. 1980–1989, 2018

[5] Ankit Kumar Srivastava, Georg Rehm and Julian Moreno Schneider, "Rumour Detection and Classification using Cascading Heuristics", Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017) - Association for Computational Linguistics, pp. 486–490, 2017

[6] A. Louni, S. Anand and K. P. Subbalakshmi, "Identification of Source of Rumors in Social Networks with Incomplete Information", arXiv:1509.00557v1, 2015

[7] M.Narmatha and S.Aruna Devi, "Rumour Source Identification in Social Networks with Time-Varying Topology", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 7, pp. 98 – 102, 2018

[8] S. Nivetha, R. Priyadharshini, P. Balakumar and R.K. Kapilavani, "Detecting root of the Rumor in Social Network using GSSS", International Research Journal of Engineering and Technology, Vol. 4, pp. 1253 – 1257, 2017

[9] Eunsoo Seoa, Prasant Mohapatrab and Tarek Abdelzahera, "Identifying Rumors and their Sources in Social Networks", Proceedings of SPIE - The International Society for Optical Engineering, Vol. 8389, 2012

[10] Raveena Dayani, Nikita Chhabra, Taruna Kadian and Rishabh Kaushal, "Rumor detection in twitter: An analysis in retrospect", IEEE International Conference on Advanced Networks and Telecommunications Systems, 2015

[11] Samah M. Alzanin and Aqil M. Azmi, "Detecting rumors in social media: A survey", The 4th International Conference on Arabic

Computational Linguistics (ACLing 2018), pp. 294 – 300, 2018

[12] Hardeo Kumar Thakur, Anand Gupta, Ayushi Bhardwaj and Devanshi Verma, "Rumor Detection on Twitter Using a Supervised Machine Learning Framework", International Journal of Information Retrieval Research, Vol. 8, 2018

[13] Dr. Dinesh B. Vaghela and Divya M. Patel, "Rumor Detection with Twitter and News Channel Data Using Sentiment Analysis and Classification", International Journal of Advance Engineering and Research Development, Vol. 5, pp. 924 – 930, 2018

[14] Tetsuro Takahashi and Nobuyuki Igata, "Rumor detection on twitter", The 6th International Conference on Soft Computing and Intelligent Systems, and The 13th International Symposium on Advanced Intelligence Systems, 2012

[15] Rosa Sicilia, Stella Lo Giudice, Yulong Pei, Mykola Pechenizkiy and Paolo Soda, "Twitter Rumour Detection in the Health Domain", Expert Systems With Applications - An International journal, Vol. 110, pp. 33 – 40, 2018

[16] Sardar Hamidian and Mona Diab, "Rumor Detection and Classification for Twitter Data", The Fifth International Conference on Social Media Technologies, Communication, and Informatics, pp. 71 – 77, 2015

[17] Sardar Hamidian and Mona T Diab, "Rumor Identification and Belief Investigation on Twitter", Proceedings of NAACL-HLT - Association for Computational Linguistics, pp. 3–8, 2016

[18] Sarah A. Alkhodair, Steven H.H. Dingb, Benjamin C.M. Fung, Junqiang Liu, "Detecting breaking news rumors of emerging topics in social media", Information Processing and Management, Vol. 57, 2019

[19] Rehana Moin, Zahoor-ur-Rehman, Khalid Mahmood, Mohammad Eid Alzahrani and Muhammad Qaiser Saleem, "Framework for Rumors Detection in Social Media", International

Journal of Advanced Computer Science and Applications, Vol. 9, pp. 439 – 444, 2018 .

[20] Quoc Le and Tomas Mikolov, “Distributed Representations of Sentences and Documents”, 2014 Cortes, C., Vapnik, V., “Support Vector Networks”

[21] Marco Tulio Ribeiro et al., “Why Should I Trust You? Explaining the Predictions of Any Classifier”, 2016 Kaggle Fake News Dataset.