Fake News Detection using CNN with PSO Optimization and Decision Tree Classifier

Hitesh Lenka
Computer Science and Engineering
CV Raman Global University
Bhubaneswar, India
hitesh.len@gmail.com

Nitish Kumar Rout
Computer Science and Engineering
CV Raman Global University
Bhubaneswar, India
badalnitish300@gmail.com

Omkareswar Hota
Computer Science and Engineering
CV Raman Global University
Bhubaneswar, India
omkareswarhota1234@gmail.com

Abstract*— Fake news refers to false or misleading information presented as if it were real news. The spread of fake news can have serious consequences, including influencing public opinion, shaping political events, and causing harm to individuals or groups. As a result, there is a need for effective methods to detect fake news in order to prevent its spread. In this paper, we explore the potential use of deep learning techniques such as Convolutional Neural Networks (CNNs), and classifiers like Support Vector Machines (SVM), Logistic Regression (LR), and Decision Trees (DT) for fake news detection. These techniques have shown promise in detecting patterns and relationships in large and complex datasets, making them well-suited for the task of identifying fake news.

Keywords- Fake news, False information, Misleading information, public opinion, Political events, Harm Detection, Deep learning techniques, Convolutional Neural Networks (CNNs), Classifiers, Support Vector Machines (SVM), Logistic Regression (LR), Decision Trees (DT), Patter, Relationship, Large datasets Complex dataset Identifying fake news.

I. INTRODUCTION

In today's digital landscape, the unbridled flow of information online has birthed both opportunity and peril. Amidst this connectivity, the proliferation of fake news looms as a grave threat, wielding the power to manipulate public sentiment, sway elections, and fracture communities. In response, our pioneering initiative bridges technology and journalism, leveraging Convolutional Neural Networks (CNNs) to construct a robust system for fake news detection. Recognizing the limitations of human cognition in unraveling deceptive textual cues, our approach harnesses CNN's prowess in deciphering nuanced patterns. Beyond technological strides, our mission aims to empower societies with the discernment tools necessary to safeguard truth in the face of misinformation. This report stands as a testament to our commitment: fortifying digital integrity while upholding the values of truth and democratic discourse

II. RELATED WORK

A. Overview

There have been quite a several initiatives taken to achieve fake news detection:

-Hiramath and Deshpande Explored various deep learning methods like Logistic Regression, Naïve Bayes, Support Vector Machine, Random Forest, and Deep Neural Networks for fake news detection. Highlighted the advantages of deep learning in handling extensive data, learning complex patterns, and improving accuracy over time.

-Li, Liu, and Zhou Introduced Discriminative-FANG, leveraging graph neural networks for fake news detection. Acknowledged limitations such as heavy reliance on labeled data for training and concerns about feature quality and computational complexity.

-Rama Krishna and Adimoolam Utilized Decision Tree and Support Vector Machine algorithms for fake news detection. Noted the interpretability advantage of Decision Trees but faced challenges due to dataset size and the system's limitations in detecting sophisticated fake news.

-Kong, Tan, Gan, and Samsudin Employed NLP techniques and TensorFlow-Keras deep learning models for fake news detection. Recognized the need for quality and quantity of training data, highlighted the effectiveness of N-gram vectors, and emphasized human verification in automated detection.

-Jose, Kumar, and Chandran (2021): Deployed Logistic Regression, Decision Trees, Random Forest, and Bi-directional LSTM for fake news detection. Focused on OSM networks, achieving high accuracy rates, especially with Bi-directional LSTM, while acknowledging the need for improvement in detecting various types of fake news.

B. outcome

Each study contributes distinct methodologies, highlighting the strengths and limitations of different approaches in combating fake news. They underscore the ongoing efforts to develop robust detection systems, emphasizing the need for diverse techniques and continual improvements to address the evolving challenges posed by misinformation.

III. ABBREVIATIONS AND ACRONYMS

Abbreviations used in this research paper are CNN: Convolutional Neural Networks, SVM: Support Vector Machines, LR: Logistic Regression, DT: Decision Trees, PSO: Particle Swarm Optimization, DACA: Deferred Action for Childhood Arrivals, ROC: Receiver Operating Characteristic, Conv Convolution, PySwarms: implements Particle Swarm Optimization, NumPy: Numerical Python, Keras: open-source neural network library written in Python., x_seq:- x sequence(input or features for training neural networks)., X_padded: variable representing padded sequences., avg: Average, val: Validation

IV. PROPOSED MODEL

3.1. Proposed Solution for Fake News Detection

CNNs and Decision Trees: In the current digital era, where information spreads at unprecedented speeds, the proliferation of fake news has become a formidable challenge with far-reaching consequences. Misleading information can sway public opinion, influence political events, and undermine the very foundations of democratic societies. It is imperative to develop robust and effective methods to detect and combat fake news, preserving the integrity of information dissemination. To address this pressing issue, our proposed solution harnesses the power of Convolutional Neural Networks (CNNs) and Decision Tree classifiers, combining deep learning with traditional machine learning techniques to create a versatile and accurate fake news detection system. parenthesis (like this). (A parenthetical sentence is punctuated within the parentheses.)

3.2 Data Collection and Preprocessing:

Our journey begins with meticulous data collection, where we gather a comprehensive dataset consisting of news articles from various sources. This dataset includes both verified true news articles and those identifiedas fake. Thorough data preprocessing follows, involving the cleaning and formatting of text data. This processincludes the removal of HTML tags, punctuation, and stop words, as well as tokenization and lowercasing. The aim is to ensure that the data is in a consistent and usable format for subsequent analysis.

3.3 Data Labeling:

With the dataset in hand, we proceed to label each news article as either "real" (representing trustworthy information) or "fake" (indicating misleading or false content). This meticulous labeling process forms the foundation for training and testing our fake news detection model.

3.4 Data Splitting:

To evaluate the model's performance accurately, we divide the labeled dataset into two distinct subsets. Typically, 80% of the data is allocated for training, while the remaining 20% is reserved for rigorous testing and validation.

3.5 Text Tokenization and Padding:

Text tokenization is a crucial step where we deconstruct the news articles into individual words or tokens. Additionally, we apply padding to ensure that all tokenized sequences have a consistent length, typically by adding zeros to shorter sequences. This step prepares the text data for input into the CNN model.

3.6 Pre-trained Word Embeddings (GloVe):

Specify the directory where the GloVe embeddings file is located (GLOVE_DIR).

3.7 CNN Model for Feature Extraction:

The heart of our proposed solution lies in the design and implementation of a Convolutional Neural Network(CNN) customized for feature extraction from text data. This CNN architecture comprises several layers, including an embedding layer to process tokenized text, convolutional layers to detect PROPOSED SOLUTION 10 | P a g e intricate patterns, max-pooling layers for effective feature selection, and dense layers for precise classification. The CNN is trained using the training dataset to learn and extract meaningful features from the text, a critical aspect of fake news detection. He text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use the scroll-down window on the left of the MS Word Formatting toolbar.

3.8 Hyperparameter Optimization:

To further enhance the performance of our CNN model, we employ optimization techniques like Particle Swarm Optimization (PSO) to fine-tune hyperparameters. PSO explores a range of settings within the CNN, such as the number of filters and kernel sizes, to maximize the model's accuracy.

3.9 Feature Extraction:

Once the CNN model is trained, it is utilized to extract pertinent features from both the training and testing data. These extracted features encapsulate essential patterns and characteristics within the news articles, aiding in accurate classification.

3.10 Decision Tree Classifier:

To augment the model's classification capabilities, we incorporate a Decision Tree classifier. This classifier is trained on the features extracted by CNN, leveraging the knowledge acquired during feature extraction to improve its overall performance in classifying news articles as either true or fake.

3.11 Model Evaluation:

Our proposed solution undergoes rigorous evaluation using standard machine learning metrics. These include accuracy, which measures the model's overall correctness, a confusion matrix that provides valuable insights into true positive, true negative, false positive, and false negative predictions, a comprehensive classification report offering detailed performance statistics, and an ROC curve illustrating the model's ability to effectively distinguish between true and fake news articles.

3.12 Visualization and Interpretation:

We visualize the results through an ROC curve, offering insights into the classifier's trade-offs between true positives and false positives at different thresholds. Additionally, we provide interpretation and analysis of the results, considering real-world implications and addressing any potential biases or limitations in the model.

In conclusion, our comprehensive proposed solution combines the strengths of Convolutional Neural Networks and Decision Tree classifiers to address the critical challenge of fake news detection. By integrating advanced deep learning techniques with traditional machine learning, we aspire to contribute a robust and practical tool for combating the growing problem of fake news in our information-driven society.

V. METHODOLOGY

4.1 Approach

The approach of "Fake News Detection using CNN with PSO Optimization and Decision Tree Classifier" is a multi-step process aimed at utilizing Convolutional Neural Networks (CNNs) optimized by Particle Swarm Optimization (PSO) and employing a Decision Tree classifier for identifying fake news. Here's a breakdown of the approach and the algorithms used:

4.1.1 Convolutional Neural Networks (CNNs):

Convolutional Neural Networks (CNNs), renowned for image analysis, adapt to textual data using key layers: the Embedding Layer, converting words to dense vectors; the Conv1D Layer, capturing local patterns through convolutional operations; and the GlobalMaxPooling1D Layer, condensing extracted features to emphasize crucial elements. This adaptable architecture allows CNNs to excel in understanding intricate linguistic patterns, making them invaluable tools in text analysis. Through sequential processing, these layers empower CNNs to discern semantic relationships, capture local nuances, and distill essential textual features, contributing significantly to tasks like fake news detection within the realm of natural language processing.

4.1.2 Particle Swarm Optimization (PSO):

Particle Swarm Optimization (PSO) draws inspiration from collective behavior observed in natural systems like flocking birds or swarming insects. This metaheuristic optimization technique serves as a powerful tool in refining the hyperparameters of Convolutional Neural Network (CNN) models. Employed iteratively, PSO operates by refining a population of potential candidate solutions. These solutions, represented as particles in the optimization space, continuously update their positions and velocities based on their own best-known positions and the collective best-known positions of the entire swarm. Through this collaborative adjustment mechanism, PSO effectively navigates the hyperparameter space, seeking and converging towards the optimal configurations that enhance the CNN model's performance for tasks such as text analysis or fake news detection..

4.1.3 Decision Tree Classifier

The Decision Tree Classifier, a supervised learning algorithm, constructs a hierarchical, tree-like structure to facilitate classification tasks. It operates by evaluating various features within a dataset, making decisions at each node based on these features' values. This sequential evaluation ultimately leads to the creation of branches and leaves that represent distinct decision paths. Notably, Decision Trees offer interpretability by showcasing the decision-making process in a comprehensible tree format, allowing users to comprehend how classifications are reached. Additionally, this algorithm excels in capturing intricate and non-linear patterns present in the data, enabling it to effectively discern complex relationships between features, thereby making it a valuable tool in classification tasks across diverse domains.

4.2 System Architecture

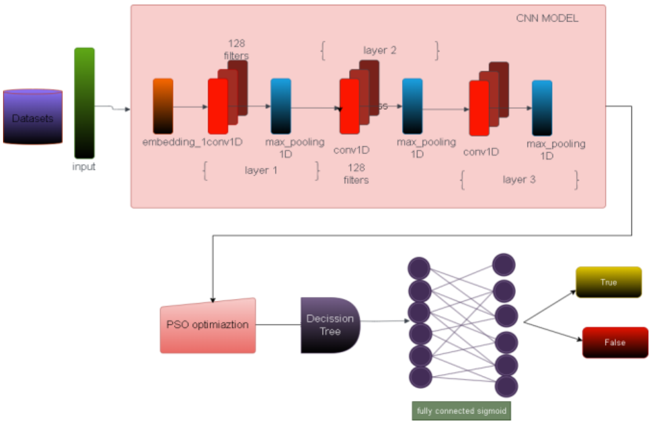
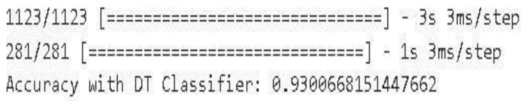


Figure 4.2 Flowchart of the model

VI. IMPLEMENTATION AND RESULTS

5.1 Accuracy

The reported accuracy of 93.01% using a Decision Tree Classifier indicates the proportion of correct predictions made by the model on unseen data, showcasing its strong predictive performance



```
1123/1123 [==============================] - 3s 3ms/step
281/281 [==============================] - 1s 3ms/step
Accuracy with DT Classifier: 0.9300668151447662
```

Figure 5.1 Accuracy of model

5.2 Confusion Matrix

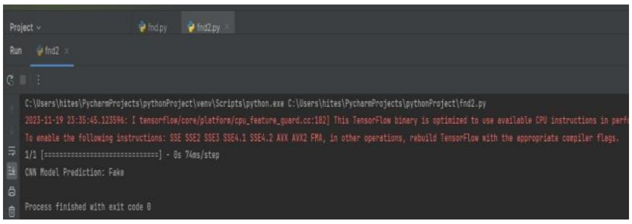
A confusion matrix is a table used in machine learning to describe the performance of a classification model. It's a grid that summarizes the predictions of a model against the actual classification of the data.

	Predicted Positive	Predicted Negative
Actual Positive	**TP (3991)**	**FN (339)**
Actual Negative	**FP (289)**	**TN (4361)**

Figure 5.2 Confusion matrix

5.3 Testing Results

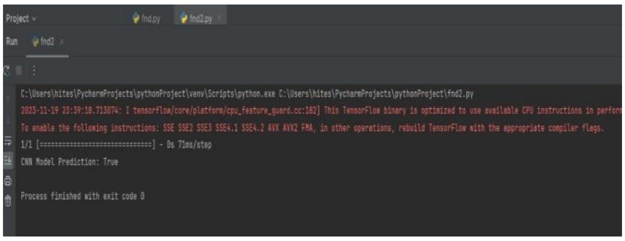
Interpretation of the prediction is essential for understanding the model's performance The label "Fake" indicates that the model believes the provided text exhibits characteristics associated with fake news.The label "True" suggests that the model considers the text to be more aligned with genuine, non-fake news.



```
C:\Users\hites\PycharmProjects\pythonProject\venv\Scripts\python.exe C:\Users\hites\PycharmProjects\pythonProject\fnd2.py
2023-11-19 23:35:45.123596: I tensorflow/core/platform/cpu_feature_guard.cc:182] This TensorFlow binary is optimized to use available CPU instructions in perf
To enable the following instructions: SSE SSE2 SSE3 SSE4.1 SSE4.2 AVX AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
1/1 [==============================] - 0s 74ms/step
CNN Model Prediction: Fake

Process finished with exit code 0
```

Figure 5.3.1 Fake prediction results



```
C:\Users\hites\PycharmProjects\pythonProject\venv\Scripts\python.exe C:\Users\hites\PycharmProjects\pythonProject\fnd2.py
2023-11-19 23:39:18.713874: I tensorflow/core/platform/cpu_feature_guard.cc:182] This TensorFlow binary is optimized to use available CPU instructions in perform
To enable the following instructions: SSE SSE2 SSE3 SSE4.1 SSE4.2 AVX AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
1/1 [==============================] - 0s 71ms/step
CNN Model Prediction: True

Process finished with exit code 0
```

Figure 5.3.2 True prediction results

5.4 ROC Curve

The Receiver Operating Characteristic (ROC) curve is a graphical representation used to evaluate the performance of a classification model. It shows the diagnostic ability of a binary classifier system as its discrimination threshold is varied.

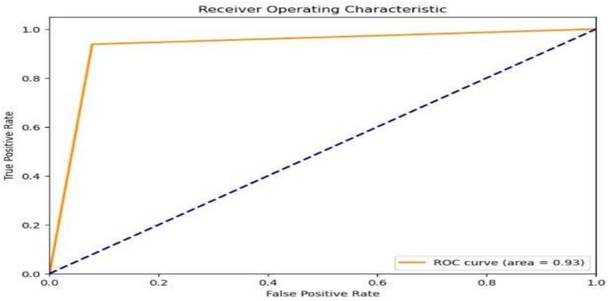


Figure 5.4 ROC curve

VII. CONCLUSION

This project merges technology and journalism, utilizing Convolutional Neural Networks (CNNs) and optimization techniques to combat fake news. It acknowledges the complexity of textual patterns and CNNs' adeptness in identifying intricate details. Employing Particle Swarm Optimization fine-tunes CNNs, ensuring robustness. The project integrates a pre-trained CNN model with a Decision Tree Classifier, enhancing news evaluation. Testing demonstrates the CNN model's potential in content-based detection. Beyond technical aspects, the project emphasizes empowering communities to discern truth. It marks a pivotal moment in merging journalistic integrity and innovation, aiming to fortify digital infrastructure and uphold democratic principles. This holistic approach signifies a milestone in combating false information and preserving information integrity in the digital era.

REFERENCES

[1] H. Cao, J. Deng, G. Dong and Duan, "A Discriminative Graph Neural Network for Fake News Detection," 2021 2nd International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE), Zhuhai, China, 2021, pp. 224-228, doi : 10.1109/ICBASE53849.2021.00049.

[2] X. Jose, S. D. M. Kumar and P. Chandran, "Characterization, Classification and Detection of Fake News in Online Social Media Networks," 2021 IEEE Mysore Sub Section International Conference (MysuruCon), Hassan, India, 2021, pp. 759-765, doi: 10.1109/MysuruCon52639.2021.9641517.

[3] N. L. S. R. Krishna and M. Adimoolam, "Fake News Detection system using Decision Tree algorithm and compare textual property with Support Vector Machine algorithm," 2022 International Conference on Business Analytics for Technology and Security (ICBATS), Dubai, United Arab Emirates, 2022, pp. 1-6, doi: 10.1109/ICBATS54253.2022.9758999.

[4] Z. Tian and S. Baskiyar , "Fake News Detection using Machine Learning with Feature Selection," 2021 6th International Conference on Computing, Communication and Security (ICCCS), Las Vegas, NV, USA, 2021, pp. 1-6, doi: 10.1109/ICCCS51487.2021.9776346.

[5] A. Matheven and B. V. D. Kumar, "Fake News Detection Using Deep Learning and Natural Language Processing," 2022 9th International Conference on Soft Computing & Machine Intelligence (ISCMI), Toronto, ON, Canada, 2022, pp. 11-14, doi: 10.1109/ISCMI56532.2022.10068440.

[6] C. K. Hiramath and G. C. Deshpande, "Fake News Detection Using Deep Learning Techniques," 2019 1st International Conference on Advances in Information Technology (ICAIT), Chikmagalur, India, 2019, pp. 411-415, doi: 10.1109/ICAIT47043.2019.8987258.

[7] S. H. Kong, L. M. Tan, K. H. Gan and N. H. Samsudin, "Fake News Detection using Deep Learning," 2020 IEEE 10th Symposium on Computer Applications & Industrial Electronics (ISCAIE), Malaysia, 2020, pp. 102-107, doi: 10.1109/ISCAIE47305.2020.9108841.

[8] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. ACM SIGKDD explorations newsletter, 19(1):22–36, 2017.

[9] Natali Ruchansky, Sungyong Seo, and Yan Liu. Csi: A hybrid deep model for fake news detection. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, Singapore, pages 797–806, 2017

[10] Naeem, S.B., Bhatti, R. & Khan, A. (2020, July 12). An exploration of how fake news is taking over social media and putting public health at risk. Wiley Public Health Emergency Collection. https://doi.org/10.1111%2Fhir.12320

[11] J. Zhang, B. Dong, and P. S. Yu. Fake detector: Effective fake news detection with deep diffusive neural network. In 2020 IEEE 36th International Conference on Data Engineering (ICDE), pages 1826–1829, 2020.

[12] F. A. Ozbay and B. Alatas , "Fake news detection within online social media using supervised artificial intelligence algorithms," Physica A: Statistical Mechanics and its Applications 540, no. 12317417,pp.1-17,2020.