# Deepfake Detection on Social Media Using Deep Learning and Fast Text Embeddings

**M. Shravani**

PG scholor, M. TECH (CSE),

JNTUH University College of Engineering, Jagityal (Autonomous), Nachupally (Kondagattu), Tg

**Dr. B. Sateesh Kumar**

Prof, Head of The Department Of CSE

JNTUH University College of Engineering

Jagtial (Autonomous), Nachupally (Kondagattu), TG

## Abstract

Recent advancements in natural language processing and generative language modeling have significantly enhanced the capabilities of deep neural networks in content creation. While these developments offer numerous benefits, they also raise serious concerns regarding the misuse of text-generative models to produce deepfake content on social media platforms. Such content, often disseminated by sophisticated social bots, can manipulate public opinion and spread misinformation. This research focuses on detecting machine-generated content on Twitter using the publicly available Tweepfake dataset. A deep learning approach based on a Convolutional Neural Network (CNN) architecture, integrated with FastText word embeddings, is proposed to classify tweets as either human-generated or bot-generated. To evaluate the effectiveness of the proposed model, it is compared against several baseline machine learning methods, including models using Term Frequency-Inverse Document Frequency (TF-IDF), FastTextsubwordembeddings, and established deep learning architectures such as CNN-LSTM and LSTM. Experimental results demonstrate that the CNN model with FastTextembeddings achieves a high classification accuracy of 93%, highlighting its robustness and suitability for detecting deepfake content on social media.

**Keywords:** ***Deep fake,CNN-LSTM,CNN,TF-IDF***

## 1. Introduction

The rapid development of natural language processing (NLP) and deep learning has revolutionized content generation across digital platforms. State-of-the-art language models, such as OpenAI's GPT series and other transformer-based architectures, have demonstrated impressive generative capabilities, allowing machines to produce human-like text with high fluency and coherence [8, 12, 13]. While these advancements bring significant benefits across sectors including healthcare, customer service, and education, they also pose serious threats. In particular, malicious actors are now leveraging these models to generate misleading or manipulative content, including deepfake text disseminated through social media [3, 9, 16].The influence of social media on public opinion is well-documented. Platforms like Twitter are increasingly used not just for communication, but also for political discourse, news dissemination, and shaping social narratives [5, 6]. However, the emergence of social bots—automated accounts that mimic human behavior—has enabled widespread manipulation campaigns, often driven by coordinated disinformation efforts [2, 7, 10]. These bots can use generative models to create realistic yet entirely fabricated content, making it difficult for users to distinguish between authentic and synthetic information [1, 14, 20].Deepfake text, unlike visual deepfakes, is harder to detect due to the lack of clear semantic inconsistencies or visual artifacts. The subtlety of synthetically generated text, particularly on platforms that limit character length like Twitter, presents unique challenges for detection systems. Previous research has shown that even human readers struggle to identify machine-generated reviews and posts when sentiment and style are preserved [15, 18].To address these concerns, researchers have begun developing automated detection tools for machine-generated content. Several approaches utilize statistical and linguistic features, such as Term Frequency-Inverse Document Frequency (TF-IDF), n-gram models, and subwordembeddings [14, 17]. More recently, deep learning-based classifiers using word embeddings like FastText and neural architectures such as LSTM, CNN, and hybrid models have shown promise in detecting deepfake text [19, 20].

This study contributes to this growing body of work by exploring a Convolutional Neural Network (CNN)-based model utilizing FastText word embeddings to classify tweets as either human- or bot-generated. Using the publicly available TweepFake dataset [19], the proposed method is benchmarked against traditional machine learning classifiers and other deep learning models such as LSTM and CNN-LSTM. The model achieves high performance, with classification accuracy reaching 93%, demonstrating its effectiveness in detecting deepfake content on social media platforms.The rest of this paper is organized as follows: Section 2 reviews related work; Section 3 describes the dataset and methodology; Section 4 presents experimental results and analysis; and Section 5 concludes with a discussion on implications and future work.

## Related Works

The rapid growth of big data across text, audio, video, and social media platforms has introduced both opportunities and challenges for analysis. Verma et al. [1] highlight the complexities of managing such heterogeneous and large-scale data, emphasizing the need for robust frameworks capable of real-time processing and multimodal analysis. These challenges intersect with the rise of automated and synthetic content generation. Early studies such as Siddiqui et al. [2] examined bots and their ability to mimic human activity, while Westerlund [3] reviewed the emergence of deepfake technology, outlining its technical foundations and societal risks. Ternovski et al. [4] extended this by testing the impact of deepfake warnings in political contexts, showing that although warnings increase skepticism, they do not significantly enhance discernment. Parallelly, Vosoughi et al. [5] demonstrated how false news spreads faster and wider than truth online, a trend amplified by organized disinformation campaigns documented by Bradshaw et al. [6]. Grimme et al. [7] further analyzed social bots, showing how human-assisted control enhances their ability to manipulate conversations.

In parallel, advances in large language models (LLMs) have transformed text generation. Liu et al. [8] and Dale [12] explored GPT's capabilities, while Lee and Hsiang [11] demonstrated applications such as patent claim generation using GPT-2. High-profile reports like Heaven [13] showed how a GPT-3 bot could interact on Reddit without detection. At the same time, researchers recognized associated risks: Zellers et al. [9,16] proposed defensive strategies such as Grover to counter neural fake news, while Gehrmann et al. [14] introduced GLTR as a visualization tool for detecting generated text. Adelani et al. [15] illustrated how sentiment-preserving fake reviews can be created and tested against human and machine detection systems. Complementary work on controllable text generation, such as Keskar et al. [17] with CTRL, highlights how generation parameters influence outputs. Uchendu et al. [18] developed TuringBench to systematically evaluate human vs. machine text identification, while Fagni et al. [19] studied deepfake tweets and Stiff & Johansson [20] investigated approaches for detecting computer-generated disinformation.

Together, these works reveal a landscape where big data challenges [1], automated text generation [8,11–13,17], deepfakes [3,4,19], disinformation dynamics [5–7], and defense/detection strategies [9,14–16,18,20] are tightly interconnected. While generative models create opportunities for useful applications, they also amplify risks of deception, misinformation, and manipulation. The literature underscores the need for scalable detection frameworks, nuanced user-facing interventions, and coherent policy responses to mitigate harms in the age of neural content generation.

## Methodology

The proposed methodology for bot detection on Twitter leverages natural language processing (NLP) and deep learning techniques, as shown in the framework diagram in figure 1,Tweets are

collected using the Twitter API to form a dataset containing textual content, user metadata, and activity information. This dataset serves as the raw input for further analysis.Raw tweets often contain noise such as URLs, hashtags, emojis, stopwords, and special characters. Preprocessing involves text cleaning, tokenization, lowercasing, and normalization. This ensures that only meaningful textual features are retained for analysis.Thepreprocessed dataset is divided into training and testing subsets. This step ensures proper evaluation of the model's generalization ability, avoiding overfitting and providing reliable performance measures.Wordembeddings are generated using FastTex**t**, which captures semantic meaning as well as subword information. This is particularly useful for handling misspellings, short forms, and noisy text commonly found in social media posts.AConvolutional Neural Network (CNN) is employed for feature learning and classification. CNNs are effective in capturing local dependencies and contextual information from word embeddings.The first CNN processes FastTextembeddings to learn high-level textual features.A second CNN layer refines the features and performs classification
The output of the CNN is passed to a fully connected layer with a softmax activation function to predict whether a Twitter account is controlled by a Bot **or a** Human**.**

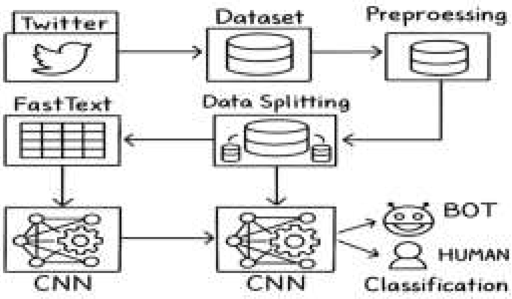
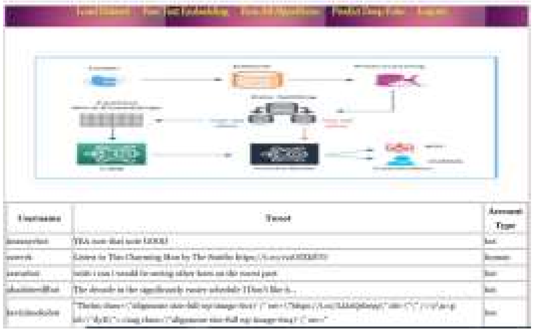


*Figure1: Framework*

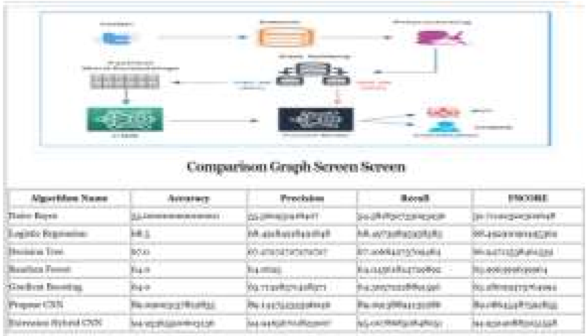
**Results and Analysis:**



| Username | Tweet | Account Type |
|---|---|---|
| [illegible] | YEA now that was GOOD | bot |
| [illegible] | Listen to The Charming Man by The Smiths https://t.co/[illegible] | human |
| [illegible] | [illegible] | bot |
| [illegible] | The [illegible] | bot |
| [illegible] | "[illegible] | bot |

*Figure 2:Dataset*

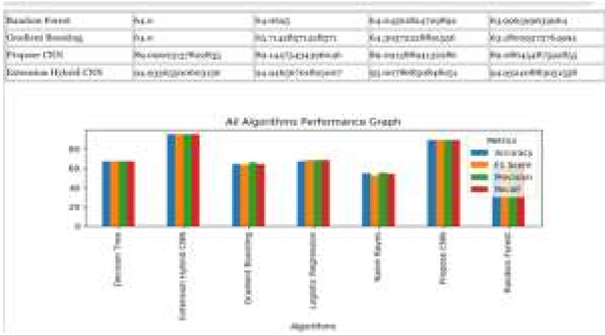
Tweets collected from Twitter are preprocessed (cleaning, tokenization, removing noise), and FastText word embeddings are applied to convert text into meaningful numerical vectors. he processed data is split into training and testing sets. A CNN model is trained on the embeddings, and the trained model classifies accounts as either Bot **or** Human as per figure 3, 4, 5 different machine learning algorithms were tested for Twitter bot detection using FastTextembeddings. Traditional models like Naive Bayes (55% accuracy), Logistic Regression (68.5%), Decision Tree (67%), Random Forest (64%), and Gradient Boosting (64%) achieved moderate performance.In contrast, deep learning models performed significantly better. The Proposed CNN reached 89.09% accuracy with strong precision, recall, and F-score values. The Extension Hybrid CNN further improved performance, achieving 94.93% accuracy, 94.95% precision, 95% recall, and an F-score of 94.93%.These results clearly demonstrate that CNN-based models outperform traditional machine learning methods, with the Hybrid CNN providing the most reliable classification of bot vs. human accounts.



Comparison Graph Screen Screen

| Algorithm Name | Accuracy | Precision | Recall | FSCORE |
|---|---|---|---|---|
| Naive Bayes | 55.[illegible] | 55.[illegible] | [illegible] | [illegible] |
| Logistic Regression | 68.5 | 68.[illegible] | 68.[illegible] | [illegible] |
| Decision Tree | 67.0 | 67.[illegible] | 67.[illegible] | [illegible] |
| Random Forest | 64.0 | 64.[illegible] | 64.[illegible] | [illegible] |
| Gradient Boosting | 64.0 | [illegible] | 64.[illegible] | [illegible] |
| Propose CNN | 89.[illegible] | [illegible] | [illegible] | [illegible] |
| Extension Hybrid CNN | 94.[illegible] | 94.[illegible] | 95.[illegible] | 94.[illegible] |

*Figure 3: performance of the model*



| | | | | |
|---|---|---|---|---|
| Random Forest | [illegible] | [illegible] | [illegible] | [illegible] |
| Gradient Boosting | [illegible] | [illegible] | [illegible] | [illegible] |
| Propose CNN | [illegible] | [illegible] | [illegible] | [illegible] |
| Extension Hybrid CNN | [illegible] | [illegible] | [illegible] | [illegible] |

*Figure 4: Comparison graph*

## Conclusion and Future Work

Deepfake text detection is a critical and challenging task in the era of misinformation and manipulated content. This study proposed a hybrid approach for detecting deepfake text by leveraging a dataset of tweets from both bots and humans. Multiple feature extraction techniques, including TF, TF-IDF, FastText, and FastTextsubwords, were explored. The integration of FastTextembeddings with Convolutional Neural Networks (CNNs)achieved superior performance compared to traditional machine learning models. The proposed method attained an accuracy of **0.93**, demonstrating its effectiveness in accurately distinguishing between human-generated and bot-generated (deepfake) text.The results highlight that advanced detection can be achieved without relying on highly complex or computationally expensive transfer learning models. This makes the approach more practical and scalable for real-world applications, particularly in social media environments where timely detection is crucial.As social media continues to influence public opinion and democratic processes, developing robust and efficient deepfake text detection systems is imperative to safeguard authentic information. For future work, research will focus on integrating Quantum Natural Language Processing (QNLP) and other emerging AI paradigms to design more sophisticated and efficient detection frameworks. These advancements are expected to further strengthen defenses against misinformation, ensuring trust and reliability in digital communication.

## REFERENCES

[1]. Praveen Gugulothu, and Rahu Bhukya. "Coot-Lion Optimized Deep learning Algorithm for COVID-19 Point Mutation Rate Prediction using Genome Sequences." Computer Methods in Biomechanics and Biomedical Engineering (Published 2023, Indexing: SCIE , IF : 3.1, Publisher : Taylor & Francis)
DOI: https:doi.org10.1080/10255842.2023.2244109

[2]. Praveen Gugulothu, and Raju Bhukya. "DNA Sequence Clustering and ERSITGRU for Repeat Detection in COVID -19 Prediction." Intelligent Decision Technologies:( Published 2025, Indexing: SCIE, IF: 5.69, Publisher: Sage IOS PRESS)
DOI: 10.1177/18724981241302033

[3]. Praveen Gugulothu , and Raju Bhukya.“Exploring Coronavirus Sequence Motifs through Convolutional Neural Network for Accurate Identification of Covid-19" Computer Methods in Biomechanics and Biomedical Engineering.( Published 2024 February 2024, Indexing: SCIE, IF: 1.9, Publisher: Taylor & Francis).
DOI: 10.1080/10255842.2024.2404149

[4]. Praveen Gugulothu, and Raju Bhukya. "Genome-Wide Analysis for Covid-19 detection for Tandem Repeat Error and substitution Error using Harris Hawks Optimization" Computers and Electrical Engineering .(Under Second Revision, Indexing: SCIE, IF: 4.3, Publisher: Elsevier).

[5]. Praveen Gugulothu , Shekhar Katukoori , Swapna Manuparthi "Deep Learning based techniques for Covid-19 diagnosis based on Various Pattern features detection from early stages of diseases”, Network Computation in Neural Systems.(Accepted May 2024, Indexing: SCIE, IF: 9, Publisher: Taylor & Francis).

[6]. Samantha Bradshaw, Hannah Bailey, and Philip N Howard. Industrialized disinformation: 2020 global inventory of organized social media manipulation. Computational Propaganda Project at the Oxford Internet Institute,2021.

[7]. Christian Grimme, Mike Preuss, Lena Adam, and Heike Trautmann.Social bots:

Human-like by means of human control? Big data, 5(4):279–293, 2017.

[8]. Xiao Liu, YananZheng, Zhengxiao Du, Ming Ding, YujieQian,Zhilin Yang, and Jie Tang. Gpt understands, too. arXiv preprintarXiv:2103.10385, 2021.

[9]. Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, AliFarhadi, FranziskaRoesner, and Yejin Choi. Defending against neural fake news.Advances in neural information processing systems, 32, 2019.

[10]. [Logan Beckman. The inconsistent application of internet regulations and suggestions for the future. Nova L. Rev., 46:277, 2021.

[11]. Jieh-Sheng Lee and Jieh Hsiang. Patent claim generation by fine-tuningopenai gpt-2. World Patent Information, 62:101983, 2020.

[12]. Robert Dale. Gpt-3: What’s it good for? Natural Language Engineering,27(1):113–118, 2021.

[13]. Will Douglas Heaven. A gpt-3 bot posted comments on reddit for aweek and no one noticed. MIT Technology Review. Retrieved November,24:2020, 2020.

[14]. Sebastian Gehrmann, HendrikStrobelt, and Alexander M Rush.Gltr:Statistical detection and visualization of generated text. arXiv preprintarXiv:1906.04043, 2019.

[15]. David IfeoluwaAdelani, Haotian Mai, Fuming Fang, Huy H Nguyen,JunichiYamagishi, and Isao Echizen. Generating sentiment-preserving fake online reviews using neural language models and their human-and machine-based detection. In Advanced Information Networking and Ap-plications: Proceedings of the 34th International Conference on Advanced Information Networking and Applications (AINA-2020), pages 1341–1354. Springer, 2020.

[16]. Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, AliFarhadi, FranziskaRoesner, and Yejin Choi. Grover-a state-of-the-artdefense against neural fake news, 2019.

[17]. NitishShirishKeskar, Bryan McCann, Lav R Varshney, CaimingXiong,and Richard Socher. Ctrl: A conditional transformer language model for controllable generation. arXiv preprint arXiv:1909.05858, 2019.

[18]. AdakuUchendu, Zeyu Ma, Thai Le, Rui Zhang, and DongwonLee.Turingbench: A benchmark environment for turing test in the age of neural text generation. arXiv preprint arXiv:2109.13296, 2021.

[19]. TizianoFagni, FabrizioFalchi, MargheritaGambini, Antonio Martella,and Maurizio Tesconi. Tweepfake: About detecting deepfake tweets. Plosone, 16(5):e0251415, 2021.

[20]. Harald Stiff and Fredrik Johansson.Detecting computer-generated disinformation. International Journal of Data Science and Analytics,13(4):363–383, 2022.