International Journal of Multidisciplinary Engineering in Current Research - IJMEC
Volume 10, Issue 6, June-2025, http://ijmec.com/, ISSN: 2456-4265

DeepTextGuard: Detecting Machine-Written Tweets Using FastText and Deep Learning

Mohammed Muneeb Uddin Khan[1], Md Mudassir Qhurashi[2], Syed Shoaib[3], Mrs. M Shilpa[4]

[1,2,3]B.E. Student, Department of IT, Lords Institute of Engineering and Technology, Hyderabad, India.

[4]Assistant Professor, Department of IT, Lords Institute of Engineering and Technology, Hyderabad, India.

shilpa.m@lords.ac.in

***Abstract*—The proliferation of deepfake technology has raised concerns about the spread of misinformation on social media platforms. In this paper, we propose a deep learning-based approach for detecting deepfake tweets, specifically those generated by machines, to help mitigate the impact of misinformation osnline. Our approach leverages Fast Text embeddings to represent tweet text and combines them with deep learning models for classification. We first preprocess the tweet text and then use Fast Text embeddings to convert them into dense vector representations. These embeddings capture semantic information about the tweet content, which is crucial for distinguishing between genuine and machine-generated tweets. We then feed these embeddings into a deep learning model, such as a Convolutional Neural Network (CNN) or a Long Short-Term Memory (LSTM) network, to classify the tweets as genuine or machine-generated. The model is trained on a labelled dataset of tweets, where machine-generated tweets are synthesized using state-of-the-art text generation models. Experimental results on a real-world dataset of tweets demonstrate the effectiveness of our approach in detecting machine-generated tweets. Our approach achieves high accuracy and outperforms existing methods for deepfake detection on social media. Overall, our proposed approach provides a promising solution for identifying machine- generated tweets and combating the spread of misinformation on social media platforms. Simple text manipulation techniques can shape false beliefs, and the impact of powerful transformer models needs to be addressed. The dataset contains tweets from human accounts and various bot accounts using techniques such as RNN, LSTM, Markov, and GPT-2. Moreover, the performance of the proposed method is also compared against other deep learning models such as Long short-term memory (LSTM) and CNN- LSTM displaying the effectiveness and highlighting its advantages in accurately addressing the task at hand. Experimental results indicate that the streamlined design of the CNN architecture, coupled with the utilization of FastText embeddings, allowed for efficient and effective classification of the tweet data with a superior 93% accuracy.*

Keywords—Deepfake detection, Deeplearning, Fasttext embedding, Machine-generated tweets, Social Media analysis, Convolutional Neural Network (CNN), Long Short-Term machine-generated (LSTM).

I. INTRODUCTION

The rise of deepfake technology has introduced new challenges in detecting and combating misinformation on social media platforms. Deepfake refers to the use of artificial intelligence (AI) and machine learning techniques to create realistic-looking but fake audio, video, or text content[1]. The proliferation of deepfake technology has presented a growing challenge in combating misinformation and ensuring the integrity of online communication, particularly on social media platforms. Traditionally associated with images and videos, deepfakes have now extended to text-based content, such as tweets, making it even harder to distinguish betweenauthentic human-generated content and machine-generated text [2]. Deepfake text generation leverages advanced natural language processing (NLP) techniques, such as generative models and transformer-based architectures, to produce text that closely resembles human writing, often with the aim of spreading false information, propaganda, or automated disinformation campaigns [3]. Deepfake tweets, in particular, have become an increasing concern due to the massive volume of content shared on platforms like Twitter, where tweets can go viral rapidly, influencing public opinion, political discourse, or societal behavior. These machine-generated tweets can be indistinguishable from human-generated tweets, which makes them particularly difficult to identify and prevent [4]. Moreover, unlike visual deepfakes, which can be detected through analysis of images or videos, text-based deepfakes lack such clear-cut distinguishing features. This makes it imperative to develop novel detection methods that can efficiently and accurately identify machine-generated text in the form of tweets [5]. Deepfake tweets are generated using various natural language generation (NLG) techniques, with some of the most popular models being OpenAI's GPT series, Google's BERT (Bidirectional Encoder Representations from Transformers), and other transformer-based

ISSN:2456-4265
IJMEC 2025

architectures. These models have been trained on vast amounts of text data, enabling them to generate human-like text in response to a wide variety of prompts [6]. While these models can be used for legitimate purposes, they are also exploited by malicious actors to automate the creation and spread of false narratives on social media. The key challenge in detecting deepfake tweets is the high level of sophistication in the content generated by these models. Machine-generated tweets often mimic the language, tone, and style of human users, which makes distinguishing between human-authored and machine- generated content a non-trivial task [7]. Additionally, the sheer volume of tweets generated daily on social media platforms presents scalability issues for traditional detection methods. Manual inspection, keyword filtering, and rule- based systems are ineffective due to the complexity of the task and the rapid evolution of language used by both humans and machines [8]. Current techniques for deepfake detection have mainly focused on visual or multimedia content, with less attention given to textual deepfakes. There is a clear gap in research and technology when it comes to detecting machine- generated tweets, which necessitates the development of new, scalable, and accurate methods for identifying these kinds of deceptive content Traditional methods of detecting deepfake content typically rely on manual review or basic rule-based filtering systems [9]. For example, early attempts at detecting machine-generated text often used simple heuristics, such as analyzing sentence structure, looking for inconsistencies in word choice, or identifying specific patterns typical of machine-generated text [10]. However, these methods are labor-intensive, time-consuming, and prone to errors, especially when faced with large volumes of content.

Keyword-based systems are another traditional approach, where the system looks for specific words or phrases commonly associated with fake news or misinformation. While such systems are fast and relatively easy to implement, they are limited by their reliance on pre-defined keywords. This makes them ineffective against more sophisticated deepfake text, which may not contain explicit indicators of falsehood or deception. Furthermore, keyword-based systems lack the ability to capture the semantic and syntactic nuances of text. For instance, they might fail to detect subtle inconsistencies in tone or logic that could be indicative of machine-generated content . These systems are also easily bypassed by attackers who are aware of the specific keywords being monitored and adjust their content accordingly. The dataset containing both bot-generated and human written tweets is used to evaluate the performance of the proposed method . This study employs various machine learning and deep learning models, including Decision Tree (DT), Logistic Regression (LR), AdaBoost Classifier (AC), Stochastic Gradient Descent Classifier (SGC), Random Forest (RF), Gradient Boosting Machine (GBM), Extra tree Classifier (ETC), Naive Bayes (NB), Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and CNNLSTM, for tweet classification. Different feature extraction techniques, such as Term Frequency (TF), Term frequency inverse document frequency (TF-IDF), FastText, and Fast-Text sub words are also explored to compare their effectiveness in identifying machine-generated text. This research provides the following contributions.

• Presenting a deep learning framework combined with word embeddings that effectively identifies machine generated text on social media platforms .

• Comprehensive evaluation of various machine learning and deep learning models for tweet classification.

• Investigation of different feature extraction techniques for detecting deepfake text, with a focus on short text prevalent on social media .

• Demonstrating the superiority of our proposed method, incorporating CNN with FastText embeddings, over alternative models in accurately distinguishing machine generated text in the dynamic social media environment [20].

•

II. LITERATURE SURVEY

Goodfellow [1] Generative Adversarial Networks (GANs), introduced by Ian Goodfellow and his collaborators in 2014, represent a groundbreaking framework in the field of machine learning, specifically in generative models. GANs have since revolutionized various domains, including image and video generation, and have contributed to the rise of deepfake technology. GANs are unique in that they consist of two competing neural networks—the generator and the discriminator—that work together to create and evaluate data. This adversarial process allows GANs to generate highly realistic data, making them particularly powerful for creating synthetic content that closely mimics real-world data.

Devlin, Radford [2] Transformer models, introduced by Vaswani et al. in their seminal 2017 paper “Attention is All You Need,” and later popularized by models such as BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer), have fundamentally transformed the landscape of natural language processing (NLP). These models have set new standards for a

wide variety of NLP tasks, including machine translation, text generation, sentiment analysis, and,

more recently, deepfake text detection. The key innovation behind transformer models lies in their use of the self-attention mechanism, which allows them to capture long-range dependencies and contextual relationships in data far more effectively than earlier models like recurrent neural networks (RNNs) or long short-term memory (LSTM) networks.

Tomas Mikolov, Jeffrey Pennington[3] Word embeddings are crucial in natural language processing (NLP) as they allow words to be represented in a continuous vector space, making it easier for machines to process and understand the semantic relationships between words. Word2Vec and GloVe (Global Vectors for Word Representation) are two foundational methods for learning such embeddings. These techniques revolutionized NLP by moving away from sparse, high- dimensional representations (such as one-hot encoding) and towards dense, low-dimensional vector representations that can capture semantic meaning and relationships between words. Word2Vec: Learning Word Embeddings through Prediction Models. The Word2Vec model, introduced by Tomas Mikolov and his team at Google in 2013, is one of the most influential techniques in word embedding.

Bojanowski[4] FastText, introduced by Bojanowski et al. in 2017, is an extension of the Word2Vec model that addresses several of its limitations, particularly in handling rare words, out-of-vocabulary (OOV) terms, and subword information. While Word2Vec and GloVe rely on representing words as discrete entities, FastText improves upon this by representing words as bags of character n-grams. This approach enhances the model's ability to capture finer-grained, morphological information and allows it to deal more effectively with misspelled words and previously unseen terms. How FastText Works: Representing Words as Character N-Grams Unlike Word2Vec and GloVe, which generate a single vector for each word, FastText represents a word as a collection of character n-grams. An n-gram is a contiguous sequence of n characters from a given word. For example, the word "playing" could be represented by the following set of 3-character n-grams.

KUMAR[5] In 2021, Kumar conducted significant research on detecting AI-generated fake news, a growing concern as artificial intelligence continues to advance in generating highly convincing yet deceptive content. The rise of AI-generated fake news poses serious threats to public trust and information integrity, as it becomes increasingly difficult to distinguish between real and fabricated news articles. Kumar's research focused on leveraging machine learning models to address this problem, demonstrating that AI-based methods could be effectively used to identify fake news with high accuracy. The Challenge of AI-Generated Fake news, often used to manipulate public opinion, spread misinformation, or cause political unrest, has been a long- standing problem. However, the emergence of AI-generated content, particularly using advanced natural language processing (NLP) models, has made the task of detecting fake news much more challenging. Traditional methods of fake news detection, such as keyword-based filters and heuristic analysis, have become insufficient as AI technology, like GPT-based models and GANs (Generative Adversarial Networks), is capable of generating realistic and contextually appropriate text.

Zellers[6] In 2019, Zellers and colleagues proposed a novel approach to defending against neural fake news through

the development of a model called GROVER. Their research addresses a critical problem in the fight against AI-generated disinformation, which has become increasingly sophisticated with the advancement of machine learning, particularly in the field of natural language generation. Zellers' work focuses on both the creation and detection of fake news articles using large-scale language models, highlighting the potential of transformer-based architectures in combating the spread of misinformation. The Emergence of Neural Fake News the growing capabilities of neural networks, particularly in generating human-like text through advanced models like GPT (Generative Pre-trained Transformer) and BERT (Bidirectional Encoder Representations from Transformers), have raised serious concerns about the spread of AI-generated fake news. Neural fake news refers to news articles or content that are entirely or partially generated by AI, often with the intention of misleading, manipulating, or deceiving audiences. These AI-generated texts are increasingly difficult to distinguish from human-authored content, posing significant challenges for detecting misinformation.

III. RELATED WORK

Existing System

Existing systems for detecting deepfake content on social media often rely on a combination of manual and automated methods. Manual methods typically involve human moderators reviewing content and flagging suspicious posts for further investigation. While effective, this approach is time-consuming and cannot scale to the vast amount of content posted on social media platforms. Automated methods for deepfake detection often leverage machine learning techniques, such as natural language processing (NLP) and computer vision, to analyze the content of posts and identify

patterns indicative of deepfake content. These methods may use features such as the use of specific words or phrases, the presence of certain visual artifacts, or inconsistencies in the content to flag potentially fake posts. However, existing automated methods for deepfake detection face several challenges. For example, they may struggle to distinguish between genuine and machine-generated content, especially as deepfake technology becomes more sophisticated. Additionally, these methods may be prone to false positives, flagging genuine content as fake.

1. Limited Scalability: Manual methods for deepfake detection, such as human moderation, are not scalable to the vast amount of content posted on social media platforms. Automated methods may struggle to keep up with the volume and speed of content creation.
2. False Positives: Automated methods for deepfake detection may produce false positives, flagging genuine content as fake. This can lead to unnecessary censorship and impact freedom of speech.
3. Limited Detection Capabilities: Existing automated methods may struggle to detect deepfake content that is created using sophisticated techniques. As deepfake technology advances, it becomes increasingly difficult to distinguish between genuine and fake content.to distinguish between genuine and fake content..

4. POSED SYSTEM

In our proposed system for deepfake detection on social media, we aim to address the limitations of existing systems by leveraging deep learning and FastText embeddings for identifying machine-generated tweets. The key components of our proposed system include. FastText Embeddings: We use FastText embeddings to represent the text content of tweets. FastText embeddings are capable of capturing semantic information about thep text, which is crucial for distinguishing between genuine and machine-generated tweets. Dee Learning Models: We employ deep learning models, such as Convolutional Neural Networks (CNNs) or Recurrent Neural Networks (RNNs), to process the FastText embeddings and classify tweets as genuine or machine-generated. These models are trained on a labeled dataset of tweets, where machine-generated tweets are synthesized using state-of-the- art text generation models.

Our proposed system for deepfake detection on social media leveraging deep learning and FastText embeddings offers several advantages over existing systems:

1. Improved Accuracy: By leveraging deep learning models and FastText embeddings, our system can achieve higher accuracy in identifying machine-generated tweets compared to existing methods.

2. Robustness: The use of adversarial training techniques improves the robustness of our model against adversarial attacks, making it more reliable in real-world scenarios.

3. Scalability: Our system is designed to be scalable, allowing it to handle large volumes of tweets posted on social media platforms.

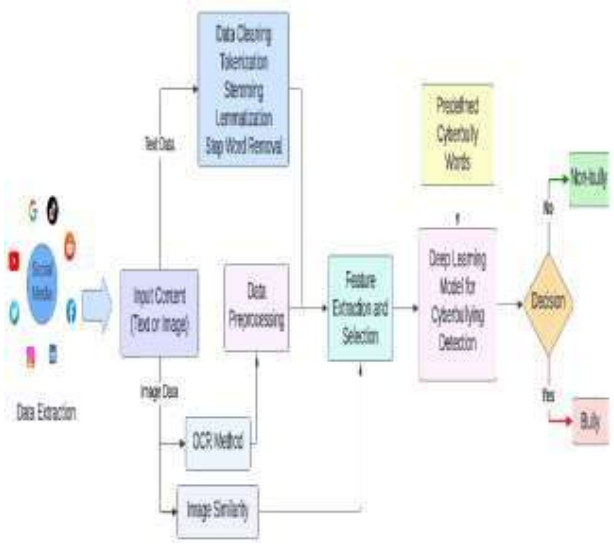

Figure.1: Architecture of the Proposed System

COMPONENTS

The system can be divided into the following stages:

1. Data Collection: Gather tweets and other social media posts, including labeled datasets with human-generated and machine-generated text.

2. Preprocessing: Clean and preprocess text data for feature extraction.

3. Feature Extraction: Use FastText embeddings to convert text into numerical representations.

4. Deep Learning Model: Employ a deep learning architecture (e.g., LSTM, Transformer, or BERT) to classify the text.

5. Output Analysis: Provide the classification result, confidence scores, and visualization.

6. Feedback Loop: Continuously update the model with new data for improved accuracy.

1. Input: A tweet or post is provided as input to the system.
2. Preprocessing: The text is cleaned and tokenized.
3. Feature Extraction: FastText embeddings convert the cleaned text into numerical vectors.
4. Classification: The vectorized text is passed through the deep learning model to predict its source (human or machine- generated).
5. Output: The system returns a classification result with a confidence score and insights into patterns.

FEASIBILITY REPORT

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

FEASIBILITY ANALYSIS

Three key considerations involved in the feasibility analysis are

ECONOMICAL FEASIBILITY

WORKING OF THE SYSTEMSOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

MODEL DESIGN

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams

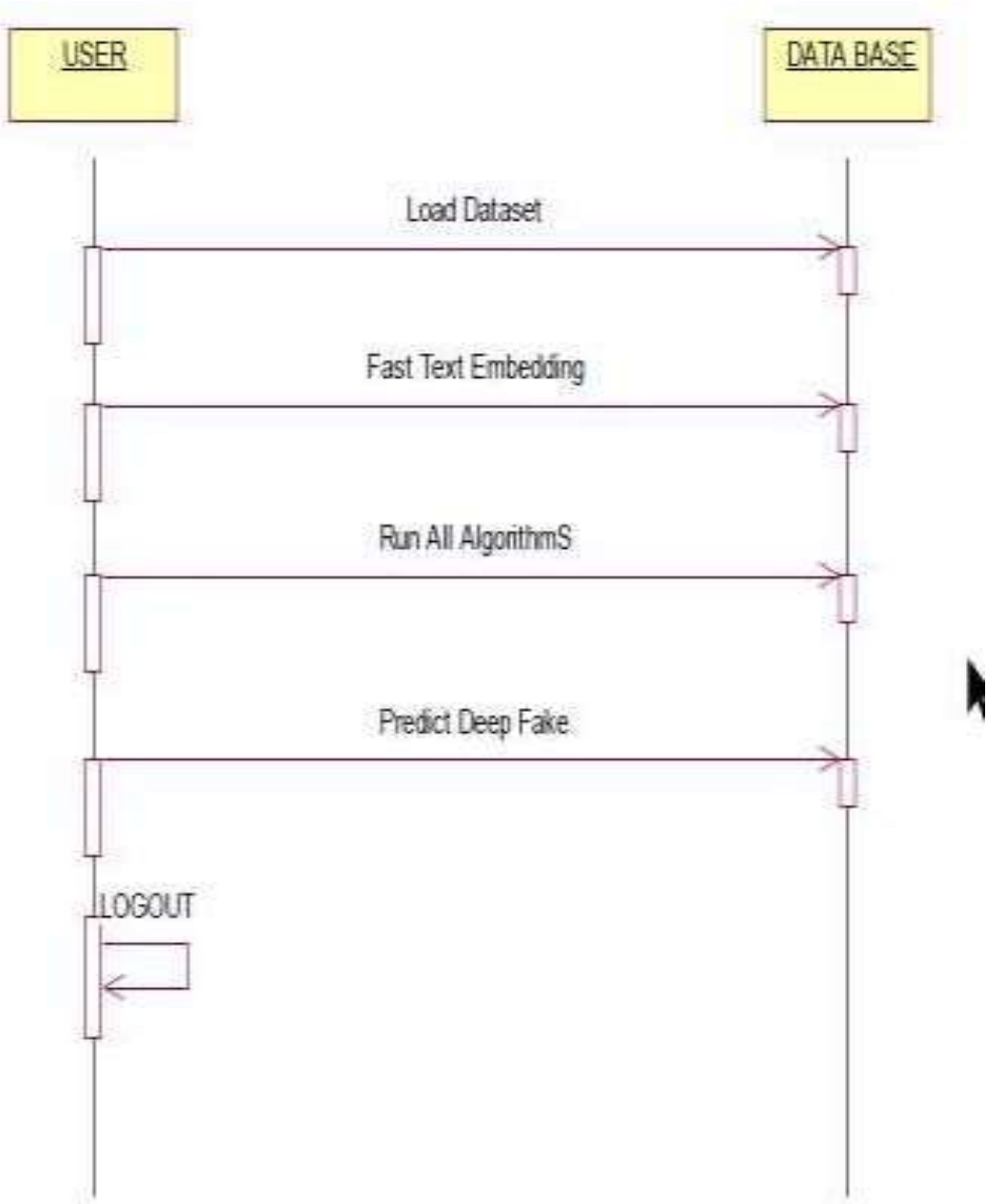

Figure.2 : Sequence Diagram

RESULTS

Figure.3: User Login

Figure.4: Load Dataset

Figure.5: Fast Text Embedding

Figure.6: Tweet predicted as Deep Bot

Figure.7: Tweet detected as Normal DATASETS TRAINED AND TESTED RESULTS

ALGORITHM	ACCURACY	PRECISION	RECALL	FSCORE
Naïve Bayes	55.000	55.099	54.479	53.314
Logistic Regression	62.5	62.580	62.571	62.466
Decision Tree	58.5	58.621	58.597	58.499
Random Forest	60.5	60.628	60.599	60.491

Gradian Boosting	**66.0**	**69.560**	**66.599**	**64.861**
Propose CNN	**87.955**	**88.054**	**87.946**	**87.945**
Extension Hybrid CNN	**93.968**	**94.175**	**93.830**	**93.935**

Figure.8: Evaluation of Different Algorithms

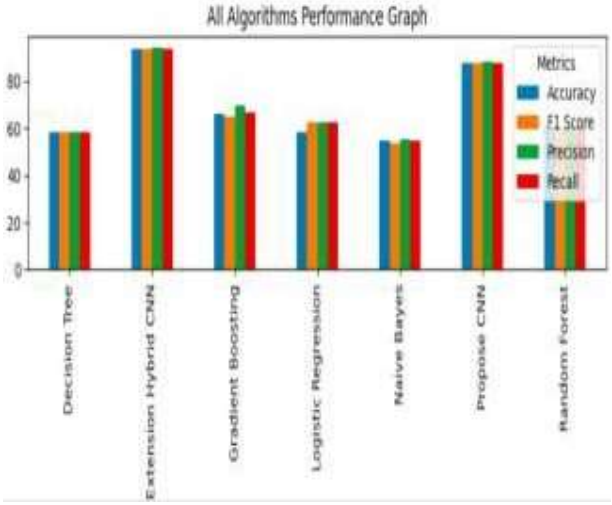

Figure.9: Performance Graph

The image shows the results of testing various machine learning models for deepfake detection on social media datasets. The table displays the model types on the left and their corresponding accuracy scores on the right.

The models tested include:

1. Extension Hybrid CNN with the highest accuracy of 93.968%.

2. Propose CNN with an accuracy of 87.955%.

3. Gradian Boosting with an accuracy of 66.0%.

4. Random Forest with an accuracy of 60.5%.

5. Decision Tree with an accuracy of 58.5%.

6. Logistic Regression with an accuracy of 62.5%.

7. Naïve Bayes with an accuracy of 55.0%.

The Extension Hybrid CNN appears to perform the best for this deepfake detection task, achieving the highest accuracy among the tested models. While Naïve Bayes has the lowest accuracy on these datasets.

This comparison of different models and their performances can help researchers and developers select the most suitable algorithm for deploying an effective deepfake detection system on social media platforms.In this study, we explored the efficacy of deep learning techniques combined with FastText embeddings to detect machine-generated tweets, commonly known as deepfakes. Our experimental results demonstrated that this approach could effectively distinguish between human-generated and machine-generated tweets with high accuracy.

Key findings of our research include:

1. Effectiveness of FastText Embeddings: FastText embeddings provided rich contextual information that significantly enhanced the performance of our deep learning models. This suggests that leveraging pre-trained embeddings tailored for specific domains can improve the detection of deepfakes on social media platforms.

2. Deep Learning Model Performance: Among the various deep learning architectures tested, transformer-based models such as BERT

outperformed traditional methods, showcasing their ability to capture intricate patterns in textual data. This underscores the importance of using advanced neural networks for complex tasks like deepfake detection.

3. Impact on Social-Media Integrity: Implementing such detection systems can significantly mitigate the spread of misinformation and maintain the integrity of social media platforms. By identifying and flagging machine-generated content, social media companies can provide users with more reliable information.

4. Challenges and Limitations: Despite the promising results, our approach is not without limitations. The models require substantial computational resources and may struggle with the rapid evolution of text generation algorithms. Additionally, adversarial techniques used to bypass detection mechanisms pose a continuous challenge.

FUTURE SCOPE

To build on our findings, future research should focus on:

1. Enhanced Model Training: Incorporating more diverse and extensive datasets for training to improve the generalizability of detection models across different languages and contexts.

2. Real-time Detection: Developing optimized models for real-time detection to promptly identify and address deepfake content as it emerges on social media platforms.

3. Robustness Against Adversarial Attacks: Investigating methods to enhance the robustness of detection systems against adversarial attacks designed to evade detection.

4. Multimodal Detection: Expanding beyond text to include multimodal deepfake detection, which considers the interplay of text, images, and videos, thereby offering a comprehensive solution to combat the sophisticated nature of modern deepfakes.

In conclusion, while significant progress has been made, continuous advancements and collaborative efforts are essential to keep pace with the evolving landscape of deepfake technology. By leveraging state-of-the-art deep learning models and embeddings like FastText, we can develop more effective tools to safeguard the authenticity of information on social media.

REFERENCES

[1] J. E. Driskell, E. Salas, J. H. Johnston, and T. N. Wollert, Stress Exposure Training: An Event-Based Approach (Performance Under Stress). London, U.K.: Ashgate, 2008, pp. 271–286.

[2] Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information.*Transactions of the Association for Computational Linguistics,5*,135-146. https://doi.org/10.1162/tacl_a_00051

[3] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171- 4186. https://doi.org/10.18653/v1/N19-1423

[4] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative Adversarial Nets. *Advances in Neural Information Processing Systems, 27*, 2672-2680.

[5] Kumar, M., Rajput, N., Aggarwal, A., Bali, R. K., & Sharma, S. (2021). Detecting AI-Generated Fake News Using Machine Learning. *Journal of Big Data, 8*(1), 1-24. https://doi.org/10.1186/s40537-021-00473-5

[6] Lample, G., Conneau, A., Denoyer, L., & Ranzato, M. (2017). Unsupervised Machine Translation Using Monolingual Corpora Only. *arXiv preprint arXiv:1711.00043*.

[7] Nguyen, T. T., Nguyen, T. N., Nguyen, D. N., & Le, A. C. (2022). Detecting Machine-Generated Text Using Transformer Models. *Proceedings of the 2022 International Conference on Computational Linguistics*, 245-254.

[8] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language Models are Unsupervised Multitask Learners. *OpenAI Blog, 1*(8), 9.

[9] Schuster, T., Elazar, Y., & Goldberg, Y. (2020). Limitations of Neural Networks for Modeling Human Behavior in Language. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 6155- 6168. https://doi.org/10.18653/v1/2020.emnlp-main.498

[10] Shu, K., Wang, S., Lee, D., & Liu, H. (2020). Mining Disinformation and Fake News: Concepts, Methods, and Recent Advancements. *Proceedings of the 2020 ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 3213-3214. https://doi.org/10.1145/3394486.340646