Human Versus Machine Generated Text Authenticity Detection System

Akella Hima Bala Padmini[1*] and Dr. G. N. V. G. Sirisha[2]
[1] Computer Science and Engineering Department, Sagi Ramakrishnam Raju Engineering College, Bhimavaram, India.
himabala@srkrec.ac.in
[2] Computer Science and Engineering Department, Sagi Ramakrishnam Raju Engineering College, Bhimavaram, India.
sirishagadiraju@srkrec.ac.in
*Corresponding author: himabala@srkrec.ac.in

Abstract. With the emergence of AI models like GPT, Deep Fake technologies, differentiating AI and human written text is increasingly complex. These models can produce highly realistic and consistent content that is difficult to detect, creating a growing demand for effective AI versus Human Text Authenticity Detection Systems. These technologies can identify minor deviations in writing style, structure and language patterns through linguistic analysis and machine learning. The technology can be used in many ways in areas like education (e.g. detecting AI-written assignments), media (e.g. authorizing human signatures in news headlines), all the way to cybersecurity (e.g. spam filters of phishing messages that are generated by AI).This research focuses on classifying text as either AI-generated or written by a human. Where, the research dealt with five varying datasets provided by Hugging Face, Google, and Kaggle, and includes variety of text types, like literature, essays and news. DistilBERT achieved 96.32% accuracy on internal validation, surpassing the ANN model's 90.78%. However, when tested on unseen data, the ANN generalized better with 95.2% accuracy, while DistilBERT's performance dropped to 49.4%. This shows that in basic text classification simpler models when trained cautiously can adapt more readily than their complicated counterparts.

Keywords: Text Classification, ANN, DistilBERT, Model Generalization, Transformer Models, NLP Robustness, Tokenization Strategy.

1. Introduction

With the advancement of generative AI, distinguishing between human-authored and machine-generated text is now a prime concern. Modern sophisticated large language models (LLMs) such as GPT, BERT, and DistilBERT are capable of producing text that well imitates human tone and consistency, making it hard to trace AI origin. This legal writing, and cybersecurity, where content authenticity is critical. Eventhough many detection tools and classifiers have come into use, most lose their accuracy when applied outside controlled environments. These drawbacks are often caused by overfitting to specific datasets and over-reliance on deep, complex architectures that lack flexibility across diverse textual inputs and domains.

© The Author(s) 2025
S. Mantena (ed.), *Proceedings of the Conference on Social and Sustainable Innovation in Technology & Engineering (SASI-ITE 2025)*, Advances in Intelligent Systems Research 199,
https://doi.org/10.2991/978-94-6463-940-7_27

To solve these problems, this research finally compares two models among various models for AI-generated text detection: Artificial Neural Network (ANN) learned from word embeddings, and the transformer model DistilBERT. DistilBERT produced a high internal accuracy of 96.32%, but external performance fell to 49.4%. For the ANN model, with an internal accuracy of 90.78%, higher external accuracy was maintained at 95.2%, which indicates improved generalizability. These results point out the impact of model architecture, tokenization, and diversity during training. The paper recommends robust model choice in favor of adaptability under real-world settings over internal performance. The paper continues with literature review (Section 2), methodology (Section 3), results (Section 4), and conclusions with future directions (Section 5).

2. Literature Survey

Research on detecting Human written Text and AI-generated text has advanced rapidly, due to the results spanning over several methodological domains. These previous research works can be grouped into perplexity-based approaches, transformer-based models, and practical limitations and ethical concerns.

2.1 Perplexity-Based Methods

As perplexity-based techniques measure content predictability—AI-generated text is typically more consistent and structured than human writing—they continue to be an important tool for identifying AI-generated text. As demonstrated by DetectGPT (Mitchell et al., 2023), whose AUROC surpassed 0.97, research indicates that these techniques can attain high accuracy. Nevertheless, they have significant drawbacks, including poor robustness against paraphrasing or biased datasets (Walters, 2023), sensitivity to text length, and high computational cost (Mindner et al., 2023). Additionally, they lack explainability and fairness, and they struggle with domain variation and multilingual content (Jawahar et al., 2020; Wu et al., 2024). While perplexity is still useful for its interpretability and text predictability, these difficulties underscore the need for more flexible and scalable detection techniques.

2.2 Transformer-Based Methods

Transformer architectures significantly increased precision. On GPT-2 outputs, Ippolito et al. (2020) demonstrated that detectors performed better than humans (88% vs. 74%). Specifically, RoBERTa performed well within the domain (Jawahar et al., 2020; Guo et al., 2023), but it had trouble generalizing to altered or cross-domain texts. According to Prova & Nuzhat (2024), BERT (93%) outperformed the SVM and XGB baselines. In lab settings, GROVER and RoBERTa produced nearly flawless results (~98.5% F1) (Pu et al., 2023); however, Yan et al. (2023) and Li et al. (2023) observed that performance declined on complex multilingual or out-of-domain data. Multilingual detection was particularly noted by Wu et al. (2024) as an unresolved problem. Transformers therefore excel in the lab but have trouble generalizing.

2.3 Practical Limitations and Ethical Concerns

Numerous studies warn that reliable real-world implementation does not always correlate with high laboratory accuracy. A three-class detection framework (AI, Human, Uncertain) was proposed by Ji et al. (2023), which demonstrated that GPTZero could achieve 97% accuracy under controlled settings but that it would need regular updates in real-world scenarios. Turnitin's AI detection plugin only detected 54.8% of AI content, according to Perkins et al. (2023), highlighting the disconnect between

research and implementation. Additionally, Weber-Wulff et al. (2023) discovered that non-native authors had an accuracy rate of less than 80%, which raised questions about discrimination and fairness. While Pattyam (2021) and Santos (2023) emphasized the necessity to address semantic ambiguity, bias, and the ethical implications of detection technologies, Sadasivan et al. (2023) attacked popular techniques like GPTZero for having poor explainability. As a result, deployment is hampered by practical constraints such as bias, lack of explainability, and poor domain transfer.

2.4 Broader Perspectives

Related research show difficulties in identifying other AI-generated items in addition to text detection. A survey of 255 deepfake publications by Mubarak & Rami (2023) revealed inadequate domain transfer and scalability. The situation is further complicated by Elkhatat et al.'s (2023) observation of uneven performance between GPT-3.5 and GPT-4. According to Sardinha (2023), dialogue is where AI writing struggles the most, demonstrating that contextual awareness is still a difference. The necessity for equity, flexibility, and moral protections is becoming more widely acknowledged in the area.

3. Proposed Methodology

3.1 Dataset Collection and Integration

Table 1 presents five datasets containing a wide range of human-written and AI-generated texts, providing diverse data for training and testing machine learning models for AI versus human text classification.

Table 1: Datasets Description

S. No.	Dataset Name	Description	Source	Type of Text	Balanced (Human/AI)	Size
1	ai-text-detection-pile	Contains AI and human-written samples from The Pile; used for training classifiers.	Hugging Face	Literature, Academic Papers, Blogs, Stack Exchange, Github, PubMed, Books, News	Balanced Human: 13,000 AI: 13,000	26,000
2	AI-human-text	Human-written and AI-generated texts from various sources.	Hugging Face	Fiction, Non-fiction, News, Technical Writing	Imbalanced Human: 1,000 AI: 2,000	3,000
3	AI-and-Human-Generated-Text	AI vs. human text samples, mostly generic writing prompts.	Hugging Face	Essays, Dialogues, Storytelling, Expositions	Balanced Human: 7,500 AI: 7,500	15,000
4	train_essays.csv	Contains short essays labeled as human or AI-written.	Google	Academic, Reflective, Narrative, Argumentative	Imbalanced Human: 1,375 AI: 3	1,378
5	train_v2_drcat_02.xlsx	Large dataset of essays with labels and prompts.	Kaggle	Education, Technology, Environment, Science, Society, Politics, Health	Imbalanced Human: 27,371 AI: 17,497	44,868

3.2 Data preprocessing

Preprocessing for human versus AI text detection involves both common and model-specific steps. Common preprocessing includes converting text to lowercase, removing punctuation/special characters, tokenizing, removing and testing sets stopwords, and splitting data into training

Model-specific preprocessing goes as follows: BERT: Uses truncation=True, padding=True, max_length=128 for consistent input size. DistilBERT: Similar to BERT but with max_length=256 for handling longer sequences. LSTM + Word Embedding: Text is tokenized with max_words=10000, and sequences are padded to max_len=200 to maintain input shape for learning temporal patterns. ANN (Fit on Words): Uses the same max_words=10000 and max_len=200 setup as LSTM for consistent input dimensions in dense layers. Word2Vec + Classifier: Tokenized input is embedded and padded to max_len=200 for uniform input vectors. Naive Bayes: No padding or truncation; uses TF-IDF or count-based features with CountVectorizer or TfidfVectorizer, which support variable-length text in sparse matrix format.

Then, Data is divided into training and test sets in the ratio of 80:20.

3.3 Classifiers

Architecture-specific classifiers are used to accommodate every architecture. For instance, transformer-based models (DistilBERT and BERT) employ fixed-size input through truncation and padding, whereas LSTM and ANN models employ tokenization and sequence padding. Numerical representations of text are implemented through vectorization methods like CountVectorizer, TfidfVectorizer, and pre-trained word embeddings (Word2Vec) based on the model.

3.4 Implementation of Different Models

The approach targets the explicit implementation and settings utilized for every model:

•Naive Bayes: Employs CountVectorizer with a vocabulary size of 10,000 to transform text into word frequency vectors. It has no use for embeddings or padding. It is speedy but does not have much to offer in terms of capturing semantics.

• ANN (Fit on Words):Tokenizes with(max_words=10,000,max_len=200) and a 128-d embedding layer. Subsequently uses two dense layers: a hidden layer (64 ReLU units) and a sigmoid output layer. Trained for 10 epochs with batch_size=32, with a total of 1.3 million parameters.

• LSTM + Word Embedding: Inherits ANN's preprocessing but includes an LSTM layer with 64 units following the 128-dimensional embedding layer to allow for temporal pattern recognition. Trained on 5 epochs with default batch size (~32) and has 1.4 million parameters.

• Word2Vec + ANN: Utilizes static pretrained 300-dimensional Word2Vec embeddings. Input gets padded up to max_len=200 and vocabulary is limited to 10,000. Only the classifier head (128-unit dense layer) gets trained, yielding ~0.4 million parameters.

•BERT: Utilizes subword tokenization and contextual embeddings of dimension 768. Inputs get padded/truncated to 128 tokens. Model has 12 attention layers and 110 million parameters. Trained for 3 epochs with batch_size=16 and learning_rate=2e-5.

•DistilBERT: A lighter variant of BERT with 6 layers but identical 768 hidden size. Can handle longer sequences (max_len=256). Trained for 3 epochs with batch_size=16 and learning_rate=5e-5, having 66 million parameters

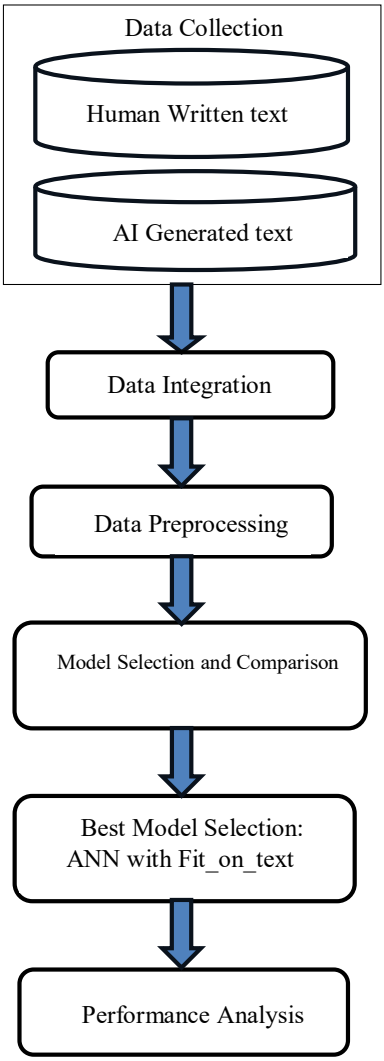

Fig. 1: Architecture of Human vs Machine Generated Text Authenticity Detection System

4. Results

During the first phase of research, three individual data sets namely—AI_Human_Generated_Text, ai_human_text, and ai_text_detection_pile — were used for assessing the performance of Multilayer Perceptron (MLP) model with TF-IDF vectorized features. All the data sets contained labeled instances as either AI or human generated. Preprocessing was done through TF-IDF vectorization (restricted to 15,000 features) and label encoding for binary classification. The MLP, comprising dense layers with ReLU activation and dropout regularization, performed well: 94.12% test accuracy on AI_Human_Generated_Text, 97.84% on ai_human_text, and 98.60% on ai_text_detection_pile. These results demonstrated the MLP's ability to differentiate between AI and human text but revealed a lack of generalization across diverse linguistic styles. In order to overcome this limitation, the three datasets were combined—each contributing 10,000 randomly chosen samples, creating a balanced aggregate dataset consisting of 30,000 text samples. This dataset was made more diverse and representative of varying writing styles. Seven models were run on this combined dataset: Naive Bayes with TF-IDF vectors, Naive Bayes with Mean Word Embeddings, ANN with Word2Vec word embeddings, ANN (FitOnText), LSTM, DistilBERT, and BERT.

Table 2: Performance Analysis of Different Models

S. No.	Model	Epochs	# Layers	Activation Functions	Training Accuracy	Testing Accuracy
1	Naive Bayes (TF-IDF Vector)	N/A	N/A	N/A	75.4%	77.37%
2	Naive Bayes (Mean Word Embedding)	N/A	N/A	N/A	72.6%	74.13%
3	ANN + Word2Vec	5	2	Sigmoid, ReLU	90.5%	88.65%
4	ANN with fit_on_text encoding	5	2	Sigmoid, ReLU	92.1%	90.78%
5	LSTM	5	3	Sigmoid, ReLU	91.8%	88.73%
6	BERT	3	12	N/A	86.0%	82.00%
7	DistilBERT	10	6	N/A	98.5%	96.32%

This assessment aimed to determine the most efficient model for generalized AI text detection across various types of content. The performance results are presented in Table 2.

LSTM and ANN with FitOnText employed Keras tokenization and padded sequence. LSTM extracts sequential dependencies, as in figure 2.

Document-level averaged embeddings were used in Word2Vec-based models, while BERT and DistilBERT employed contextual embeddings through transformers and fine-tuned classification heads, as in figure 3.

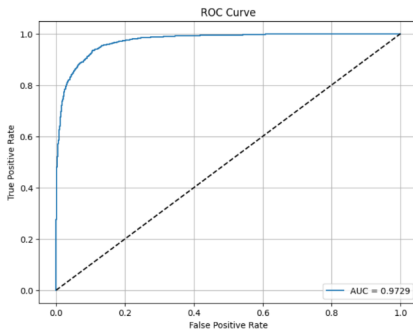

Fig 2: ROC Curve for ANN with FitOnText

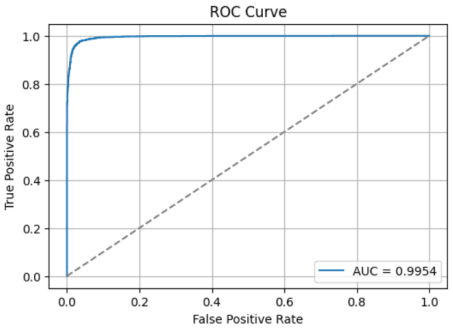

Fig 3: ROC Curve for DistilBERT

For further evaluation of model generalization capability, all the trained models were re-checked on a newly generated evaluation set of 15,000 unseen text samples, each drawn equally from the same three datasets 5,000 texts each from AI_Human_Generated_Text, AI-human-text and ai_text_detection_pile. This new test set was intentionally isolated from all training and early testing to mimic a realistic testing situation over various linguistic patterns.

Interestingly, although most models maintained robust performance, some had significant declines. FitOnText ANN persisted with 95.2% accuracy, and LSTM came next with 94.9%, reflecting their robustness and generalization ability. Word2Vec + ANN reached 92.3%, reflecting stable behavior too. On the contrary, DistilBERT, while having high train/test accuracy, fell to 49.3% on the new merged test set, indicating overfitting or being data-sensitive. Likewise, BERT scored 86.0%, having the same original testing score, but still less robust than more basic models. Naive Bayes (TF-IDF) was light and yet hit 79.5%, and its embedding version surprisingly hit 88.6%, perhaps being more robust against varying sentence structures.

In Figure 4, horizontal bar chart illustrates external accuracy scores across different models, with “Fit on Words” and “LSTM + Embedding” showing the highest general-

ization. DistilBERT performs significantly lower, indicating potential overfitting or domain mismatch. This dramatic drop in DistilBERT performance likely stems from sensitivity to domain shift, since distillBert is over-reliant on surface features, it may not deeply classify the patterns of text detection and possibly due to insufficient fine-tuning epochs. This aligns with prior literature noting transformer brittleness outside controlled environments.

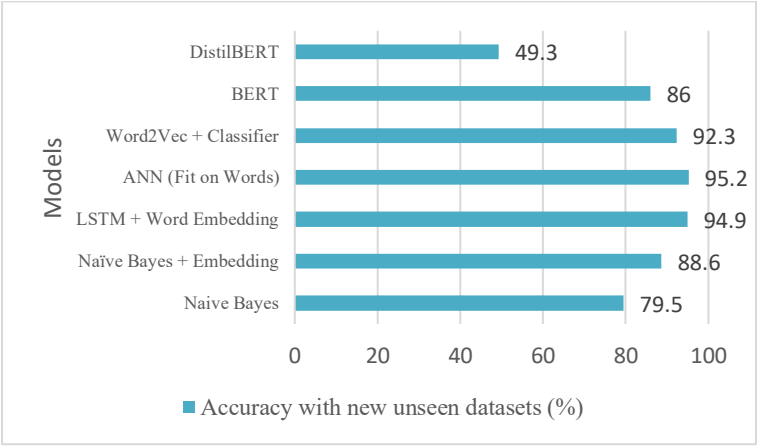

Figure 4: Model-wise test accuracy on new unseen datasets (%)

This reassessment highlights both internal performance and generalization ability. While transformer models are best at learning capacity, less complex neural networks such as FitOnText and Word2Vec provide more stable and robust results on unseen, mixed-domain data sets and are therefore better for deployability in a scalable way in AI-generated content detection systems.

5. Conclusion and Future Work

Transformer-based models such as BERT and DistilBERT demonstrated excellent internal accuracy as a result of deep contextual learning, but fared poorly in terms of generalizability—DistilBERT plummeted from 96.32% test accuracy to 49.3% on new data, which reflects overfitting. Lower-order models such as ANN (FitOnText) and LSTM, on the other hand, retained good performance on internal as well as external datasets, reflecting superior generalization.

This superior generalization is attributed to their use of uniform input representations, fewer parameters, and domain-agnostic training, which make them more versatile across different types of text. They also demand much less computation—DistilBERT required 38+ hours of training, whereas ANN required ~4 hours, making them cost-effective, scalable for deployment.

In practical scenarios, these simpler neural networks offer a preferable trade-off among accuracy, generalization, and computational resource usage.

5.1 Future Work:

Future research should focus on more sophisticated detection capabilities because plagiarism now takes on more complicated forms than just copy-paste verification. This entails expanding detection to multilingual language, where grammatical and semantic variances make identification challenging, and creating techniques for confirming the validity of code, since programming plagiarism frequently entails minute changes. Additionally, as many existing detectors are susceptible to minor perturbations or rewording, it is imperative to improve resilience against paraphrasing and adversarial attacks. Lastly, investigating hybrid strategies that use statistical methods, transformer-based models, and semantic similarity metrics may provide a more equitable trade-off between scalability, explainability, and accuracy in practical applications.

References

[1]. Sardinha, Tony Berber. "AI-generated vs human-authored texts: A multidimensional comparison." Applied Corpus Linguistics 4.1 (2024): 100083.

[2]. Ippolito, Daphne, et al. "Automatic detection of generated text is easiest when humans are fooled." arXiv preprint arXiv:1911.00650 (2019).

[3]. Jawahar, Ganesh, Muhammad Abdul-Mageed, and Laks VS Lakshmanan. "Automatic detection of machine generated text: A critical survey." arXiv preprint arXiv:2011.01314 (2020).

[4]. Wu, Junchao, et al. "A survey on llm-gernerated text detection: Necessity, methods, and future directions." arXiv preprint arXiv:2310.14724 (2023).

[5]. Prova, Nuzhat. "Detecting AI Generated Text Based on NLP and Machine Learning Approaches." arXiv preprint arXiv:2404.10032 (2024).

[6]. Sadasivan, Vinu Sankar, et al. "Can AI-generated text be reliably detected?." arXiv preprint arXiv:2303.11156 (2023).

[7]. Ji, Jiazhou, et al. "Detecting machine-generated texts: Not just" ai vs humans" and explainability is complicated." arXiv preprint arXiv:2406.18259 (2024).

[8]. Pattyam, Sandeep Pushyamitra. "AI-Enhanced Natural Language Processing: Techniques for Automated Text Analysis, Sentiment Detection, and Conversational Agents." Journal of Artificial Intelligence Research and Applications 1.1 (2021): 371-406.

[9]. Guo, Biyang, et al. "How close is chatgpt to human experts? comparison corpus, evaluation, and detection." arXiv preprint arXiv:2301.07597 (2023).

[10]. Mindner, Lorenz, Tim Schlippe, and Kristina Schaaff. "Classification of human-and ai-generated texts: Investigating features for chatgpt." International Conference on Artificial Intelligence in Education Technology. Singapore: Springer Nature Singapore, 2023.

[11]. Mitchell, Eric, et al. "Detectgpt: Zero-shot machine-generated text detection using probability curvature." International Conference on Machine Learning. PMLR, 2023.

[12]. Walters, William H. "The effectiveness of software designed to detect AI-generated writing: A comparison of 16 AI text detectors." Open Information Science 7.1 (2023): 20220158.

[13]. Weber-Wulff, Debora, et al. "Testing of detection tools for AI-generated text." International Journal for Educational Integrity 19.1 (2023): 26.

[14]. Elkhatat, Ahmed M., Khaled Elsaid, and Saeed Almeer. "Evaluating the efficacy of AI content detection tools in differentiating between human and AI-generated text." International Journal for Educational Integrity 19.1 (2023): 17.

[15]. Li, Yafu, et al. "Deepfake text detection in the wild." arXiv preprint arXiv:2305.13242 (2023).

[16]. Santos, Fátima C. Carrilho. "Artificial intelligence in automated detection of disinformation: a thematic analysis." Journalism and Media 4.2 (2023): 679-687.

[17]. Perkins, Mike, et al. "Game of tones: faculty detection of GPT-4 generated content in university assessments." arXiv preprint arXiv:2305.18081 (2023).

[18]. Mubarak, Rami, et al. "A survey on the detection and impacts of deepfakes in visual, audio, and textual formats." IEEE Access (2023).

[19]. Pu, Jiameng, et al. "Deepfake text detection: Limitations and opportunities." 2023 IEEE symposium on security and privacy (SP). IEEE, 2023.

[20]. Yan, Duanli, et al. "Detection of AI-generated essays in writing assessments." Psychological Test and Assessment Modeling 65.1 (2023): 125-144.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.