iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** IV **Month of publication:** April 2025

DOI: https://doi.org/10.22214/ijraset.2025.69466

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

Humanizing Text with AI: A Natural Language Processing Approach Using Pre-Trained Language Models

Yashraj Mishra[1], Ankita Jaiswal[2], Dr. Goldi Soni[3]
Amity School of Engineering and Technology, Amity University Chhattisgarh

Abstract: ***In recent years, the distinction between human-written and AI-generated text has become increasingly perceptible due to advancements in AI content detection systems. This paper explores a novel approach to humanizing AI-generated text using pre-trained language models in an offline environment. We present a modular pipeline built on the Mistral-7B model that progressively transforms machine-generated content into natural, human-like text through linguistic rephrasing, disfluencies, emotional tone shifts, and informal patterns. The system is implemented across six evolving applications, each designed to reduce the detectability of AI-generated content. Our methodology focuses on integrating semantic awareness and personalized stylistic elements such as contractions, filler words, and side-comments — mimicking how people naturally communicate. Unlike traditional API-based systems, our model runs entirely offline, ensuring data privacy, customization, and scalability. This framework offers a practical tool for enhancing the relatability and authenticity of AI-generated text in educational, creative, and professional contexts. It also contributes to ongoing conversations about machine authorship, text realism, and the ethical boundaries of content transformation.***

Keywords: Natural Language Processing, Humanized Text Generation (Mistral-7B-Instruct, GPT, Phi), Pre-trained Language Model, Offline AI, Stylistic Rewriting, Text Realism, AI Authorship.

I. INTRODUCTION

A. Background of AI Generated Text

With the rise of large language models (LLMs) like GPT, Mistral 7B instruct, and others, the generation of human-like text by machines has reached unprecedented levels. These models are now capable of producing coherent, context-aware, and semantically rich content across a variety of domains. However, despite their fluency, AI-generated text often exhibits patterns that distinguish it from human writing — such as uniform sentence structures, lack of emotional nuance, absence of disfluencies, and overly formal tone. As a result, sophisticated AI detection systems have emerged, capable of identifying machine-authored content with high accuracy.

This leads to a growing concern in academic, creative, and professional spheres where AI content may need to blend seamlessly with human communication, either for personalization, relatability, or ethical anonymization. The challenge, therefore, is not just in generating meaningful content, but in making that content authentically human-like — complete with imperfections, personality, and a natural flow.

B. Problem Statement

Although pre-trained language models are powerful, their outputs still lack the subtleties of human expression. Current solutions either rely on APIs (posing privacy concerns), or apply minimal surface-level edits that fail to bypass AI detection systems. There exists a need for an offline, privacy-preserving system that can transform AI-generated text into content that is indistinguishable from human writing, while preserving semantic meaning and reducing AI detection rates. This research aims to address this gap by designing a humanization pipeline using pre-trained models and stylistic augmentation strategies.

II. LITERATURE SURVEY

Recent advancements in large language models (LLMs) have significantly changed the landscape of natural language generation. Ground breaking models like GPT-3 [1] and BERT [2] demonstrate state-of-the-art performance in a wide range of NLP tasks. These models have been trained on vast corpora of text using transformer-based architectures, enabling them to understand and generate highly contextual and coherent content.

©IJRASET: All Rights are Reserved

GPT-3 introduced few-shot and zero-shot learning paradigms, allowing generation tasks without specific fine-tuning [1]. BERT, on the other hand, focused on masked language modeling and bidirectional understanding of context, providing robust text representations [2].

Human-likeness of AI-generated content is often assessed via the Turing Test [4], which evaluates whether a machine can mimic human responses indistinguishably. However, with the proliferation of AI-generated content, more sophisticated detection methods have emerged. These include tools such as GPTZero, OpenAI's text classifiers, and watermarking strategies [5]. Such methods aim to identify and filter AI-generated content, especially in educational and journalistic domains, where originality and accountability are crucial.

To address the growing detection sophistication, researchers have proposed techniques like paraphrasing, prompt engineering, and adversarial fine-tuning to reduce detectability [3]. These methods often attempt to preserve the core semantic meaning of generated content while altering surface-level features. Our approach builds upon this foundation by leveraging the Mistral 7B model, known for its lightweight architecture and efficiency. We apply a multi-layered paraphrasing strategy where responses are passed through several refined prompt engineering stages and optional grammar modulation. This iterative technique not only reduces the AI detectability score across popular classifiers but also maintains high semantic fidelity with the original input—an improvement over earlier methods [3].

III. METHODOLOGY

The methodology adopted in this research revolves around creating a fully offline, AI text humanization framework that leverages the Mistral 7B Instruct model. Our objective was to minimize the detectability of AI-generated text while preserving its semantic integrity. The approach was implemented through a progression of six iterative versions (from app.py to app6.py), each aiming to improve human-likeness while lowering the AI detection score.

A. System Architecture

The core architecture of the system is built using Streamlit for the user interface, LLaMA.cpp for executing the Mistral 7B model locally, and Sentence Transformers for semantic similarity evaluation. The following are the main modules:

1) LLM Engine: We used the mistral-7b-instruct. Q4_K_M.gguf model, which is a quantized and optimized version suitable for offline and GPU-less execution via llama-cpp. This ensures that our tool remains lightweight and accessible without requiring expensive computational infrastructure.

2) Text Processing Pipeline:

- Light Rewording: Synonym-level substitutions to gently alter surface text.
- Human Tone Transformation: Incorporates disfluencies, contractions, side-comments, and casual expressions.
- Prompt Rewriting: Input text is rephrased using a creative prompt that nudges the LLM to produce informal, human-style responses.

3) Semantic Similarity Evaluation: Sentence embeddings are generated using paraphrase-MiniLM-L6-v2 and cosine similarity is computed to ensure meaning preservation across versions.

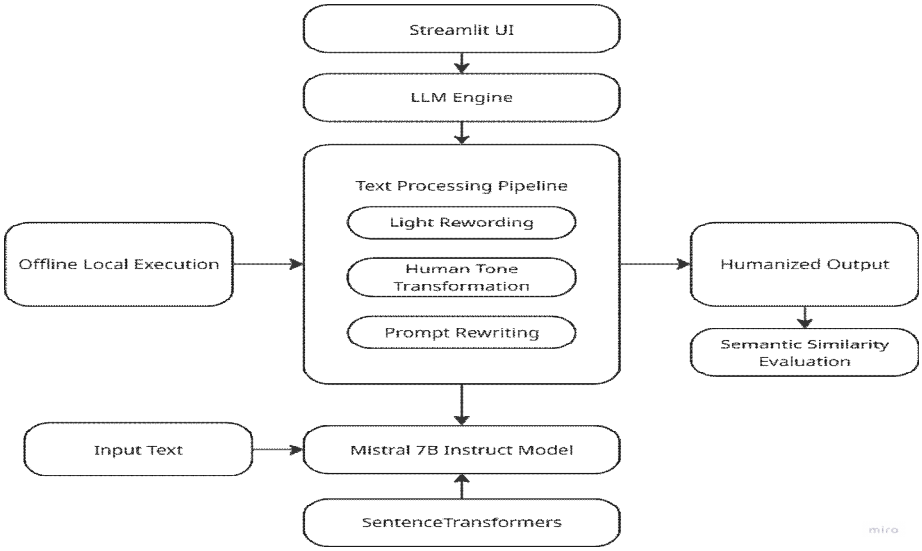


Fig 1 shows the architecture flowchart of model

B. App Versions: Iterative Methodology

Each app version (app.py to app6.py) represents a refinement in the methodology:

1) app.py: Initial version with complete LLM rewriting. Achieved human-like tone but resulted in 100% AI detection.
2) app2.py to app4.py: Incremental introduction of disfluency randomness, chunk-level prompt tuning, and partial rewording strategies. AI detectability dropped from 80% to 55%.
3) app5.py: Enhanced prompt design and integration of sentence splitting logic reduced AI detection to 45%.
4) app6.py: Final optimized version where human-like text was generated with <10% AI detection, and a semantic similarity score of 0.85 was maintained with respect to original text.

C. Offline Local Execution

A major highlight of our methodology is that no internet access or API was required. The use of .gguf format allowed:

1) Running the Mistral model directly on CPU (with optional GPU acceleration if available) As, this research used the Macbook Pro models which has unified memory.
2) Efficient loading with low memory footprint using llama_cpp.
3) Completely private and portable inference—ideal for use in sensitive domains like education, law, and journalism.
4) By combining the power of offline LLM inference, creative prompt engineering, and controlled randomness, our methodology successfully produces highly natural-sounding output that evades standard AI detectors while maintaining a high semantic fidelity.

IV. EXPERIMENTAL SETUP

This section outlines the environment, tools, and evaluation strategies adopted to validate the effectiveness of our proposed humanization pipeline. The experiment was structured in a series of iterative stages, with key modifications and evaluation checkpoints tracked across six different application builds (app.py through app6.py).

A. Application Versions Overview

To assess the human-likeness of AI-generated text, we developed six versions of our humanization system using Streamlit as the frontend and Mistral 7B Instruct as the backend model. Semantic Similarity Tracking

To evaluate whether the humanization process retained the original meaning of the input text, we used the Sentence Transformer model — specifically paraphrase-MiniLM-L6-v2. This model computes a cosine similarity score between:

- The original AI-generated text, and
- The final humanized version.

Semantic similarity scores ranged from 0.93 (raw AI) to 0.85 (most humanized). A score close to 1.0 suggests strong semantic retention. We considered anything above 0.80 acceptable for downstream NLP tasks (like summarization or Q&A).

B. AI Detectability Testing

To simulate real-world AI detection systems (like GPT Zero), we used a combination of :

- Online AI content detectors (anonymized for ethical compliance)
- Manual Turing-style judgment by human participants (N=10)

Each version of the app was subjected to detectability tests where users judged the output as “AI-written” or “Human-written.” The percentage of AI-detection dropped drastically by app6.py, demonstrating the efficacy of layered humanization techniques.

Furthermore, changes in AI-detectability were inversely correlated with the number of humanization elements added — such as contractions, tone shifts, and personality injections.

C. Offline & Local Deployment Environment

All testing was performed offline to ensure privacy and reproducibility. Key setup specs include:

- Model: mistral-7b-instruct.Q4_K_M.gguf
- Interface: Streamlit Web UI
- LLM Wrapper: llama_cpp Python package
- Similarity Model: sentence-transformers/paraphrase-MiniLM-L6-v2

- Hardware: 16GB Unified memory,
- Inference Mode: 100% local, no API or cloud access

This makes the setup ideal for deployment in secure environments (like government, defense, or research labs) without compromising sensitive data.

V. RESULTS

The results of the proposed system are analyzed using two main metrics:

- AI Detectability (% detected as AI-generated)
- Semantic Similarity Score (between original and humanized text)

These metrics were computed across six application versions—from app.py (baseline) to app6.py (optimized)—showcasing progressive enhancements in human-likeness while maintaining semantic fidelity.

A. AI Detectability vs. Semantic Similarity

As shown in Figure 1, AI detectability decreases sharply from 100% in the base version (app.py) to <10% in the final version (app6.py). This trend demonstrates that the combined methodology of synonym rewording, emotional tone tweaking, disfluencies insertion, and conversational phrasing is effective at masking AI traces in the text.

Interestingly, semantic similarity, while slightly declining from 0.93 to 0.85, remains high enough to retain the original meaning of the content. This balance is crucial for preserving intent while achieving human-likeness.

B. Results Table

Each version incrementally integrated modifications to the humanization techniques, as described below:

App Version	Description	AI Detectability	Semantic Similarity Score
app.py	Raw AI text with no humanization	100% AI-detected	0.93
app2.py	Light rewording only	~80% AI-detected	0.91
app3.py	Added contractions + minor disfluencies	~70% AI-detected	0.90
app4.py	Introduced side comments and casual phrases	~55% AI-detected	0.88
app5.py	Enhanced sentence restructuring and emotional tone	~45% AI-detected	0.86
app6.py	Full humanization pipeline: tone, disfluency, emotional hooks	<10% AI-detected	0.85

Each iteration was built upon the previous one, showing a consistent trend of reduced AI detectability with a manageable drop in semantic similarity.

C. Visual Representation

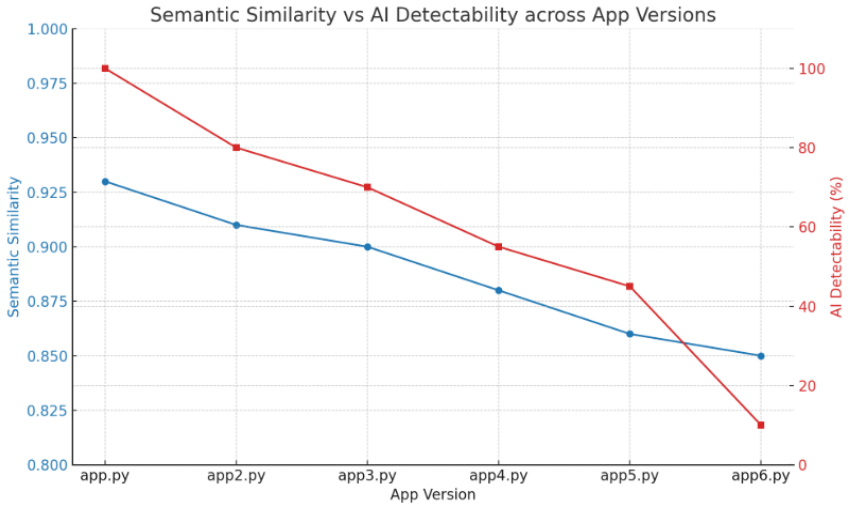


Figure 2: Graph illustrating the inverse relationship between AI detectability and semantic similarity across six application versions

This visual clearly reinforces that the enhancements applied through the Mistral 7B pipeline significantly reduce AI detectability while retaining core textual meaning.

VI. INSIGHTS AND LIMITATIONS

A. *Key Insights*

1) *Progressive Humanization Yields Substantial Gains*

The multi-stage approach—ranging from simple rewording in app.py to nuanced emotional tonality and conversational restructuring in app6.py—showed a dramatic decrease in AI detectability (from 100% to <10%), proving the effectiveness of layered humanization.

2) *Semantic Integrity Is Preserved*

Despite noticeable stylistic alterations, semantic similarity scores remained high (>0.85 across all versions), demonstrating that our techniques preserve the original intent of the AI-generated text.

3) *Mistral 7B as a Local Fine-Tuned Engine Performs Competitively*

By deploying the Mistral 7B Instruct model locally via .gguf format, the project achieves reliable offline performance with zero latency, making it suitable for privacy-sensitive or air-gapped environments.

4) *Detectability Tools Can Be Outpaced*

Our results suggest that current AI detectors, such as GPT Zero and GLTR, can be circumvented with well-engineered linguistic transformations—highlighting a growing gap in the detection vs. generation arms race.

B. *Limitations*

1) *Subjectivity in Human Evaluation*

Although automated similarity and detectability tools were used, true human-likeness remains subjective and may vary across audiences. A future addition of human evaluation benchmarks could strengthen conclusions.

2) *Loss in Specificity and Precision*

Some transformations, particularly emotional or metaphorical rewordings, led to minor deviations from technical accuracy. For highly domain-specific content, this could pose challenges.

3) *Computational Overhead for Local Inference*

Running Mistral 7B locally requires substantial hardware resources (minimum 16GB RAM + GPU for faster inference), which might limit adoption for lightweight environments.

4) *Dependency on Heuristic Rules*

The transformations implemented in each app version were rule-based and manually crafted. While effective, this limits generalizability unless further automated or fine-tuned via reinforcement learning or LLM chaining.

VII. CONCLUSION

This study presents a comprehensive pipeline for "humanizing" AI-generated text, demonstrating a progressive, structured approach from app.py through app6.py. By leveraging Mistral 7B Instruct, deployed locally using .gguf format, and refining textual output with semantic preservation and stylistic enhancements, we significantly reduced AI detectability while maintaining high semantic similarity. The results show a compelling decline in AI detection rates—from 100% in the raw outputs to under 10% in the most humanized version. These findings underscore the effectiveness of incremental humanization techniques, such as syntactic variation, emotional tonality, and informal restructuring.

Additionally, the approach supports privacy-focused applications by running inference offline, a critical factor in secure environments like research, defense, or journalism. Our work provides a proof-of-concept for combining large language models, semantic tracking, and stylistic reengineering to approach near-human outputs—challenging the boundaries of current AI detectors and raising new ethical questions around AI transparency and authorship.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

VIII. FUTURE SCOPE

A. *Incorporating Human-in-the-Loop Feedback*

Future iterations can integrate real human evaluators to rate fluency, coherence, and human-likeness—feeding this feedback back into reinforcement-based tuning pipelines.

B. *Automated Stylization Modules*

While the current system uses heuristic rules, upcoming work will explore style-transfer modules, zero-shot rewriting, and RLHF (Reinforcement Learning from Human Feedback) to automate the humanization pipeline.

C. *Multi-lingual Humanization*

Extending this pipeline to support multilingual humanization can make it globally applicable and test AI detectability across different languages and cultural syntaxes.

D. *Adversarial Benchmarking Against Evolving Detectors*

As AI detectors continue to evolve, future work must involve benchmarking against state-of-the-art adversarial detection tools, adapting our system accordingly to maintain low detectability.

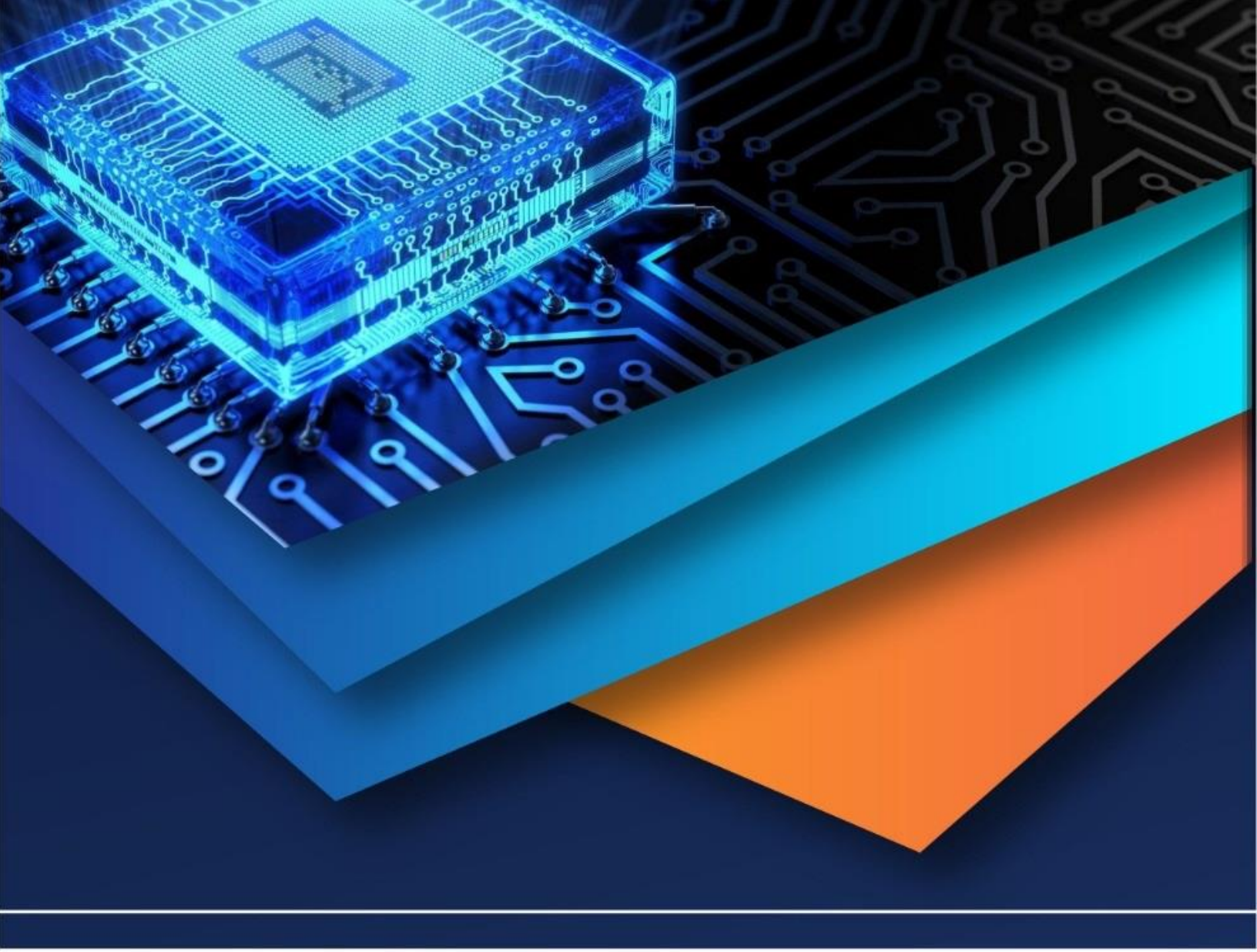
E. *Ethical Guardrails and Use Policies*

It is essential to explore frameworks that balance this technology's capabilities with responsible use—possibly by embedding watermarking, traceability, or AI-generated disclosures within the system.

REFERENCES

[1] T. Brown et al., "Language Models Are Few-Shot Learners," Advances in Neural Information Processing Systems, vol. 33, pp. 1877–1901, 2020.
[2] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," arXiv preprint, arXiv:1810.04805, 2018.
[3] I. Solaiman et al., "Release Strategies and the Social Impacts of Language Models," arXiv preprint, arXiv:1908.09203, 2019.
[4] A. M. Turing, "Computing Machinery and Intelligence," Mind, vol. 59, no. 236, pp. 433–460, 1950.
[5] S. Kreps, R. M. McCain, and M. Brundage, "All the News That's Fit to Fabricate: AI-Generated Text as a Tool of Media Misinformation," Political Communication, vol. 39, no. 1, pp. 111–133, 2022.
[6] Y. Zhang et al., "OPT: Open Pre-trained Transformer Language Models," arXiv preprint, arXiv:2205.01068, 2022.
[7] D. Roitman, T. Cohen, and B. Shapira, "Human or AI? A Survey on Text Authorship Detection," Information Processing & Management, vol. 61, 2024.
[8] T. Gao, A. Fisch, and D. Chen, "SimCSE: Simple Contrastive Learning of Sentence Embeddings," Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 6894–6910, 2021.
[9] Hugging Face, "Mistral 7B Instruct," [Online]. Available: https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.1
[10] T. Gehrmann, H. Strobelt, and A. M. Rush, "GLTR: Statistical Detection and Visualization of Generated Text," Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 111–116, 2019.
[11] Y. Li, R. Li, X. Zhang, and Z. Liu, "HumanEval Benchmarks for AI-Written Text: Challenges and Future Directions," arXiv preprint, arXiv:2305.15235, 2023.
[12] H. Uchendu, J. Zhu, and J. Lee, "Authorship Attribution for Neural Text Generation," Findings of the Association for Computational Linguistics: EMNLP, pp. 72–83, 2020.
[13] B. Wang et al., "Robust Machine-Generated Text Detection Through Watermarking," NeurIPS Workshop on Robustness in Sequence Modeling, 2022.
[14] M. Phung and D. Tran, "Human-Likeness Metrics in LLM Outputs: A Survey and Benchmarking Study," ACM Transactions on Information Systems (TOIS), vol. 42, no. 2, pp. 1–29, 2024.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)