Who is the Writer?Identifying the Generative Model by Writing Style

Anonymous ACL submission

Abstract

Texts generated by generative models closely resemble high-quality human-written texts, identifying human and model-generated texts presents a significant challenge.To address this, we present the Identify the Writer by Writing Style (IWWS) model, a novel approach designed to identify the writing styles of human and generative model. To establish a robust foundation for research in distinguishing texts generated by human and generative model, we also propose a comprehensive dataset, HumanGenTextify.Experimental results demonstrate the superiority of the IWWS model over existing methods. It not only achieves high accuracy in text source identification but also provides insights into the distinctive writing styles that characterize human and model-generated texts. Our work lays the groundwork for future explorations into automated text classification and opens new avenues for research into the authenticity.

1 Introduction

Since the release of ChatGPT, the gap between human capabilities and large language models (LLMs) has gradually narrowed(Tang et al., 2023). LLMs can achieve human-level performance in many fields(Jansen et al., 2022), and the open-source community is witnessing a surge in open-source models like LLaMA (Touvron et al., 2023),Bloom (Workshop et al., 2022) and ChatGLM(Du et al., 2021) . These models are capable of generating coherent, fluent, and meaningful texts, significantly improving the quality of generated text. It is becoming increasingly difficult to distinguish their output from human writing, both grammatically and semantically, posing considerable challenges to the social information ecosystem(Ghosal et al., 2023).

Research(Ueoka et al., 2021) indicates that false information generated by state-of-the-art LLMs is more credible than that created by humans, highlighting the challenge humans face in distinguishing between human and model-generated texts(Spitale et al., 2023). The need for practical identification of model-generated texts has garnered widespread attention. One approach involves watermarking generated texts.However, this technique requires modifications to the text generation process that could lower content quality. (Kirchenbauer et al., 2023).On the other hand, techniques like GPT-zero, DetectGPT(Mitchell et al., 2023), and classifiers from OpenAI(OpenAI et al., 2023) require access to deployed models, leading to significant costs and resource consumption. Moreover, the undisclosed internal mechanisms of many LLMs reduce their interpretability, presenting a challenge for users in understanding the decision-making process and addressing potential biases and errors(Fröhling and Zubiaga, 2021).

Thus, this paper explores the feasibility of automatically identifying whether fragments are written by humans or generated by large language models using a small model. To achieve this goal, we constructed a comprehensive dataset, HumanGenTextify, aimed at preserving the core information and context of the data, bridging the text generation differences between humans and large models. We also proposed a multi-dimensional feature fusion framework that considers the grammatical features, semantic coherence, and writing style differences of the text to distinguish between human-written and large language model-generated texts. Furthermore, by introducing a new loss function based on contrastive learning, our framework can extract high-quality feature representations from complex text data, providing support for the automatic identification task.Our main contributions include:

- We compute the perplexity(PPL) for each token across various text sources by , integrating these scores into the embeddings to enhance text source differentiation;

- Proposing a loss function by constructing a similarity matrix and contrastive learning which significantly enhances identify performance based on their writing styles;
- By creating the HumanGenTextify dataset to establish a robust foundation for research in distinguishing texts generated by human and generative model.

2 IWWS Model

To identify whether a text is created by a human or a generative model, we have proposed the method of Identifying the Writer by Writing Style (IWWS). The overview is depicted in Figure 1.

2.1 Centroids for Writing Styles

Our IWWS model introduces a novel approach to identify the writing style of each generation source, whether human or model-generated, by calculating centroids. A centroid represents the average of all embedding vectors belonging to the same generation source, effectively capturing the core characteristics of that group's writing style. This method allows analysis of writing styles by creating a mathematical representation of what distinguishes one group's writing from another's.

2.2 Similarity Matrix and Centroids Analysis

By assessing the distances between each text embedding and the style centroids of various sources, our model is designed to keep each text embedding close to its source's style centroid. The similarity matrix $s_{ji,k}$ aims for higher similarity values within the same source and lower values across different sources. it defined as the cosine similarity between each embedding vector e_{ji} and all centroids c_k ($1 \leq j, k \leq 2$ and $1 \leq i \leq M$), constructing a similarity matrix that defines the relationships between each e_{ji} and all centroids c_k.

$$S_{ji,k} = w \cdot \cos(e_{ji}, c_k) + b \tag{1}$$

where w and b are learnable parameters. We constrain the weight ($w > 0$) because we desire a greater cosine similarity to correspond to a higher degree of similarity. Figure 1 illustrates the entire process, showcasing features from different text sources, embedding vectors, and similarity scores, each represented by different colors. This approach optimizes the model's ability to accurately classify texts by ensuring embedding vectors are nearer to the correct centroid while distancing them from others, thereby optimizing classification boundaries.

This methodological framework underpins the model's capacity to discern and quantify the nuanced differences in writing styles across a diverse range of texts, highlighting its potential for applications in identifying the origins of text whether generated by humans or models.

We employ the softmax function and cross-entropy loss to refine this process, optimizing the model to ensure that each text sample is accurately classified according to the generation source that best matches its writing style. This reflects the writing style of either humans or generative models(Crothers et al., 2023).

Softmax: We set a softmax on $\mathbf{S}_{ji,k}$, where $k = 1, 2$ to make the output equal to 1 if $k = j$, otherwise the output is 0. Hence, the loss on each embedding vector $\mathbf{e}_{ji}$ can be defined as:

$$L(e_{ji}) = -S_{ji,j} + \log \sum_{k=1}^{N} \exp(S_{ji,k}) \tag{2}$$

This means that each embedding vector is pushed closer to its style centroid and pulled away from the centroids of other styles.

Cross-Entropy: Learning of embedding vectors is optimized through the cross-entropy loss. For each embedding vector, the model predicts its similarity scores with all centroids, which are then transformed into a probability distribution using the softmax function. The cross-entropy loss function calculates the difference between this predicted probability distribution and the actual one-hot encoded labels, quantifying the error. During training, by minimizing the cross-entropy loss, the model learns to adjust parameters to ensure embedding vectors are closer to the correct centroid while distancing from others, optimizing classification boundaries.

$$L(p, q) = -\sum_{i} p(i) \log q(i) \tag{3}$$

where, $p(i)$ represents the true distribution of the target categories (0 for human, 1 for model-generated labels), and $q(i)$ represents the probability distribution predicted by the model. For each sample, the difference between the true labels and the predicted probability distribution is computed. The model adjusts its parameters to minimize this loss, thereby improving the accuracy of predictions for the correct category.

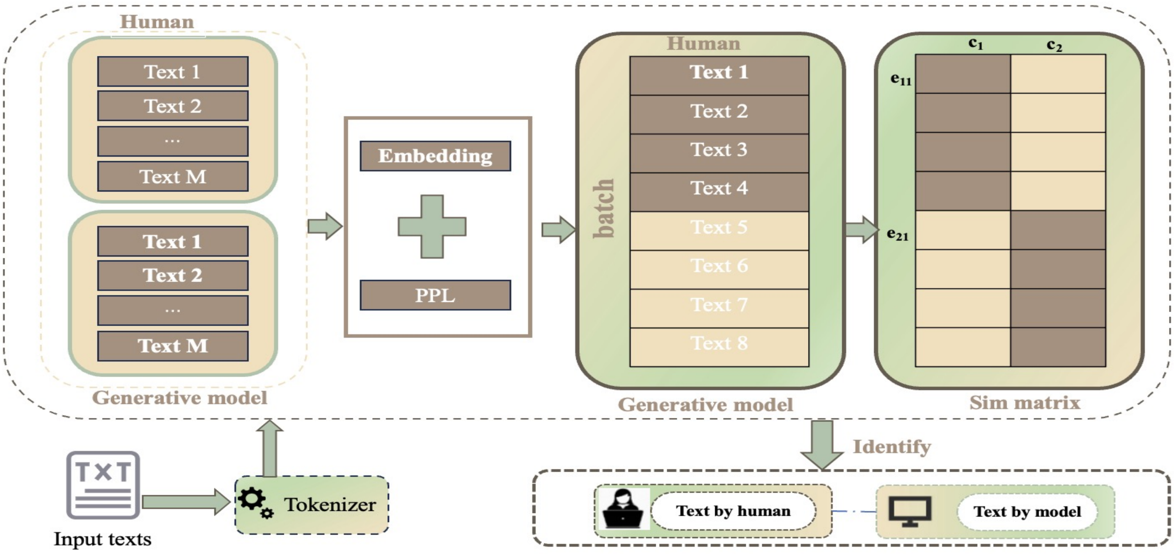

Figure 1: Method overview. Different colors indicate texts/embeddings from different sources..

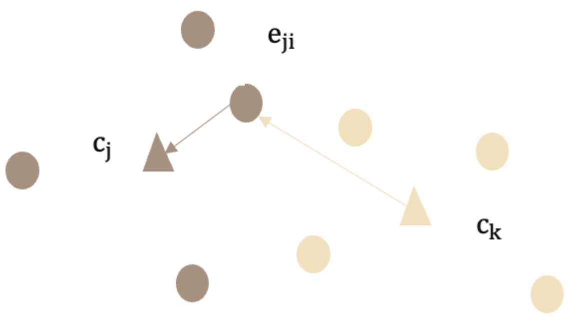

Figure 2: The loss function.It aims to pull the embedding closer to the centroid representative of the text's origin and push it away from the centroids of other text sources.

2.3 Embedding Enhancement

We enhance text embeddings by integrating semantic,syntactic features extracted using a pretrained BERT model,and Perplexity (PPL) scores to enrich the embeddings. This method involves initially processing text data to capture its inherent semantic and syntactic nuances via BERT. Accordingly,to further refine embeddings, we incorporate PPL scores,aim to leverage the model's uncertainty in text generation as an additional feature,enhancing our model's ability to differentiate between human and model-generated texts.

2.4 Training Method

Our training approach processes multiple texts simultaneously in batches that include two sources of text (human or model-generated), with an average of M texts per source. Initially, semantic and syntactic features of text fragments are extracted using a pretrained BERT model(Pizarro, 2019).

These features are then combined with the PPL of the text to construct an enhanced embedding vector that includes PPL information. Feature vector x_{ji} (where $1 \leq j \leq 2$ and $1 \leq i \leq M$) represents features extracted from texts of source j. These features are inputted into the network for further processing.

3 Experiment

3.1 Datesets

In our experiments, we utilized the English data provided in Task 1 of the AuTexTification dataset[1].

Additionally, we created own dataset, HumanGenTextify, by integrating human-written texts from the AuTexTification dataset with texts generated by three large language models (Bloom-7b, ChatGLM-6b, LLaMA2-7b). We developed a dataset for identifying human and generative model texts, emphasizing preserving and enhancing the core information and context of the original texts while introducing new expressions to increase diversity and authenticity. Our innovative approach involves rewriting existing texts with large language models rather than merely extracting the first few tokens, addressing the limitations of methods that only use the first five tokens as prompts in capturing the full scope of articles, supporting

[1] https://sites.google.com/view/autextification/data

Table 1: The Dataset of model-generated dectection task

Datasets	Train	Test	Mean_len	Max_len
AuTexTification	33846	21833	305.4	588
HumanGenTextify	35224	21283	288.3	633

Table 2: Performance metrics for text identify methods

Method	AuTexTification			HumanGenTextify		
	Precision (%)	Recall (%)	F1 (%)	Precision (%)	Recall (%)	F1 (%)
FT-RoBERTa	77.09	78.13	76.09	80.02	69.15	77.16
TALN-UPF	80.03	74.16	68.16	79.74	75.12	70.50
CIC-IPN-CsCog	64.77	69.50	74.14	70.02	72.23	68.10
IWWS	79.5	78.04	**78.13**	**80.29**	**76.03**	**79.26**

Table 3: The ERR (%) of text Dectection

Cross-Entropy	+Similarity Matrix
8.3	7.18

model generalization, and simulating real text generation processes. This dataset aims to reflect real-world text generation scenarios, providing a solid foundation for distinguishing between human and machine-generated texts and offering valuable resources for exploring the behaviors of human and machine text generation. We found that with nucleus sampling (Holtzman et al., 2019), using a top-p of 0.9 and a temperature of 0.7, the models generated texts of higher quality.

3.2 Metrics

We define our task as a binary classifier, where it is commonly believed that examining the ROC curve and the Area Under the Curve (AUC) as a performance metric is considered comprehensive. However, it is argued in literature(Wu et al., 2023)that these metrics alone are insufficient when measuring the identify accuracy of LLMs. To address this problem, we have adopted the Equal Error Rate (EER) as our primary metric. A lower EER value indicates better effectiveness in minimizing both false acceptances and false rejections simultaneously.

3.3 Results

Table 2 summarizes the performance metrics using different identify methods like FT-RoBERTa, TALN-UPF,CIC-IPN-CsCog(Sarvazyan et al., 2023) and our writing style.

On AuTexTification dataset, our IWWS reached a precision of 79.5%, a recall of 78.04%, and an F1 score of 78.13%. The result highlights the outstanding performance both precision and recall, particularly when compared to other methods such as Fine-tuned RoBERTa and CIC-IPN-CsCog, where our approach showed significant improvement across all metrics.

On our HumanGenTextify dataset, the IWWS method achieved a precision of 80.29%, a recall of 76.03%, and an F1 score of 79.26%. Compared to FT-RoBERTa and TALN-UPF, our method had higher precision and F1 scores on this dataset, underscoring the effectiveness of our approach in identifying human and machine-generated texts.

Table 3 provides a comparative evaluation of EER performance. The initial column reports results utilizing cross-entropy exclusively, while the subsequent column details EER outcomes derived from our IWWS model. We can see our approach yields an EER of 7.18%, an improvement over the conventional method's EER of 8.3%, marking a reduction of 1.17%. This demonstrates that our method, by integrating multidimensional text features with an optimized loss function, more effectively reduces classification errors.

4 Conclusion

In this paper,we have introduced the IWWS method, an innovative approach combining perplexity-based embeddings with writing style analysis, to distinguish between human and model-generated texts. Compared to existing models, IWWS demonstrates superior performance, notably enhancing text source identification accuracy. Additionally, we propose a new dataset,HumanGenTextify, offers a rich resource for further exploration.

References

Evan Crothers, Nathalie Japkowicz, and Herna L Viktor. 2023. Machine-generated text: A comprehensive survey of threat models and detection methods. *IEEE Access*.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2021. Glm: General language model pretraining with autoregressive blank infilling. *arXiv preprint arXiv:2103.10360*.

Leon Fröhling and Arkaitz Zubiaga. 2021. Feature-based detection of automated language models: tackling gpt-2, gpt-3 and grover. *PeerJ Computer Science*, 7:e443.

Soumya Suvra Ghosal, Souradip Chakraborty, Jonas Geiping, Furong Huang, Dinesh Manocha, and Amrit Singh Bedi. 2023. Towards possibilities & impossibilities of ai-generated text detection: A survey. *arXiv preprint arXiv:2310.15264*.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Tim Jansen, Yangling Tong, Victoria Zevallos, and Pedro Ortiz Suarez. 2022. Perplexed by quality: A perplexity-based method for adult and harmful content detection in multilingual heterogeneous web data. *arXiv preprint arXiv:2212.10440*.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models. *arXiv preprint arXiv:2301.10226*.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. *arXiv preprint arXiv:2301.11305*.

OpenAI, :, Josh Achiam, et al. 2023. Gpt-4 technical report.

Juan Pizarro. 2019. Using n-grams to detect bots on twitter. In *CLEF (Working Notes)*.

Areg Mikael Sarvazyan, José Ángel González, Marc Franco-Salvador, Francisco Rangel, Berta Chulvi, and Paolo Rosso. 2023. Overview of autextification at iberlef 2023: Detection and attribution of machine-generated text in multiple domains. *arXiv preprint arXiv:2309.11285*.

Giovanni Spitale, Nikola Biller-Andorno, and Federico Germani. 2023. Ai model gpt-3 (dis) informs us better than humans. *arXiv preprint arXiv:2301.11924*.

Ruixiang Tang, Yu-Neng Chuang, and Xia Hu. 2023. The science of detecting llm-generated texts. *arXiv preprint arXiv:2303.07205*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Honai Ueoka, Yugo Murawaki, and Sadao Kurohashi. 2021. Frustratingly easy edit-based linguistic steganography with a masked language model. *arXiv preprint arXiv:2104.09833*.

BigScience Workshop, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Kangxi Wu, Liang Pang, Huawei Shen, Xueqi Cheng, and Tat-Seng Chua. 2023. Llmdet: A third party large language models generated text detection tool. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2113–2133.

Limitations

The limitation of this paper is not succeeded in more refined levels of detection, such as the ability to track and identify texts generated by specific models. Future work could focus on enhancing the precision of detection techniques, thereby enabling more detailed analysis and recognition of texts from various sources and types.

Ethics Statement

All work in this paper adheres to the ACL Code of Ethics.