Fake It 'Til You Make It Human

Notebook for the ELOQUENT Lab at CLEF 2025

Aldan Creo[1,*], Maximiliano Hormazábal-Lagos[2], Héctor Cerezo-Costas[2] and Pedro Alonso-Doval[2]

[1]*Independent Author, Dublin, IE*
[2]*Fundación Centro Tecnolóxico de Telecomunicacións de Galicia, Vigo, ES*

Abstract

We explore four different approaches to evading AI-generated text detectors, each targeting distinct aspects of text humanization. Our methods range from surface-level character obfuscation through homoglyph substitution and typo injection, to cross-language translation that simulates non-native English patterns via back-translation from multiple source languages. We also investigate deeper humanization strategies including authentic character voice modeling through detailed persona creation, and cognitive simulation that models internal thought processes to influence writing decisions.

Our evaluation demonstrates substantial improvements over existing baselines, with all four approaches outperforming a GPT-3.5 baseline by factors of 10× to 17×. Our translation-based method achieved the strongest performance, securing second place in the ELOQUENT 2025 shared task with a mean evasion score of 0.64162. Interestingly, obfuscation-based methods consistently outperformed cognitive modeling approaches, revealing significant vulnerabilities in current AI detection systems and highlighting the need for more robust detection mechanisms.

Keywords

AI-Generated Text Detection, Machine-Generated Text Detection, AI-Generated Text Humanization, Large Language Models, Natural Language Processing

1. Introduction

With the advent and growing ubiquity of large language models (LLMs) in real-world applications, there has been increasing interest in developing reliable AI-generated text detectors. The misuse of LLMs in graded academic tasks, the generation of sensitive information, or the spread of fake news represents potential threats that are likely to become more prevalent as these models become increasingly indistinguishable from human performance.

Several benchmarks have been established to consolidate efforts in the detection of AI-generated text and to provide systematic frameworks for comparison and future improvement. In English, the Voight-Kampff benchmark in PAN 2024 [1], the AuTexTification Task at IBERLEF 2023 [2] and 2024 [3] — covering multiple languages of the Iberian Peninsula — and a SemEval-2024 task with similar objectives [4] are notable examples. In all these benchmarks, AI-generated text detection is framed as a binary classification task in which a detector must distinguish between human-written and machine-generated texts. The difficulty of these benchmarks is heavily influenced by the contexts used to generate the data and the sophistication of the LLMs employed to deceive the detectors. State-of-the-art LLMs are harder to detect as their outputs increasingly resemble human writing, exhibiting fewer artifacts such as repetitive phrasing or low lexical diversity that earlier models often produced and which could be caught by simple heuristics.

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain
*Corresponding author.
research@acmc.fyi (A. Creo); mhormazabal@gradiant.org (M. Hormazábal-Lagos); hcerezo@gradiant.org (H. Cerezo-Costas); palonso@gradiant.org (P. Alonso-Doval)
https://acmc.fyi (A. Creo); https://github.com/maxhormazabal (M. Hormazábal-Lagos); https://github.com/hmightypirate (H. Cerezo-Costas); https://github.com/PedroDoval (P. Alonso-Doval)
0000-0002-7401-5198 (A. Creo); 0009-0003-3687-1924 (M. Hormazábal-Lagos); 0000-0003-2813-2462 (H. Cerezo-Costas); 0009-0000-8255-3466 (P. Alonso-Doval)

The ELOQUENT Task at CLEF 2025 aims to advance research into strategies that can consistently evade current state-of-the-art AI-generated text detectors. Evasion and detection are two sides of the same coin: developing better evasion techniques will help foster the creation of more robust detectors, and vice versa.

We contribute four approaches to this task, which we describe in Section 3. They range from surface-level alterations, to non-native language pattern emulation through back-translation, to deeper "humanization" strategies that model character voices and cognitive processes. We describe our results in Section 4, where we show that our methods outperform a GPT-3.5 baseline by factors of 10 to 17 times higher, with `Lost in Translation` achieving the second place in the ELOQUENT 2025 shared task [5, 6] and discuss the different performances of our approaches in Section 5.

2. Background

The task of AI-generated text detection is traditionally considered a binary classification task. Hence the most practical approach is to train binary classifiers to perform the task, either with classical approaches such as logistic regression or SVMs [7] or using Transformer-based architectures such as the OpenAI detector [8].

Watermarking is another common technique used in AI-generated text detection. These methods typically involve modifying the heuristics of text generation, which requires access to the model's internal mechanisms. This allows the generated text to be detected and often attributed to a specific model instance. Watermarking approaches usually maintain a *red* and *green* list of words that can be used at each step of the generation process [9]. These lists may be bypassed in low-entropy contexts (e.g., code generation, legal documents) to avoid producing nonsensical output when no suitable words exist in the *green* list [10]. Watermarking can be particularly useful in scenarios where text generation is controlled or originates from widely used third-party models that implement well-known watermarking techniques.

A different approach involves strategies that analyze the statistics of the text. For example, GLTR [11] provides a human-in-the-loop assistant to aid in the detection of generated text using simple indicators extracted from a language model and generation heuristics (e.g., entropy of the generated text, rank of a word within the full vocabulary, probability of that word in context, etc.). With the help of a visualization tool, users can assess whether a text has a high likelihood of being generated.

DetectGPT [12] introduces a method based on the hypothesis that small perturbations to AI-generated text will result in lower token log probabilities compared to the original. An external model (e.g., T5 [13]) is used to perturb the text while preserving its original meaning.

Another approach that leverages statistical information from models is Binoculars [14]. Binoculars employs a pair of models that share the same token dictionary. It contrasts the outputs of both models to derive two metrics: the perplexity of the text and the cross-entropy between the models. The underlying intuition is that human-generated text typically exhibits higher entropy than AI-generated text. Cross-entropy serves as a gauge for how much entropy can be attributed to the context itself. When two similar models yield low cross-entropy, it generally indicates that the context has high intrinsic entropy. Binoculars is considered a strong baseline in AI-generated text detection. In Telescope [15], an additional fine-tuning step was introduced to enhance the detection capabilities of Binoculars, particularly for minority language models.

Paraphrasing the text using a different model has been tested for its effectiveness against various types of AI-generated text detectors. This method proves especially successful against watermarking algorithms, as it alters the word sequence generated by the original model [16]. In [17], the authors propose a method to generate a soft prompt — a learned prompt designed to maximize the likelihood of evading detection when generating outputs. This technique was successfully evaluated against DetectGPT and a fine-tuned OpenAI classifier developed for this task.

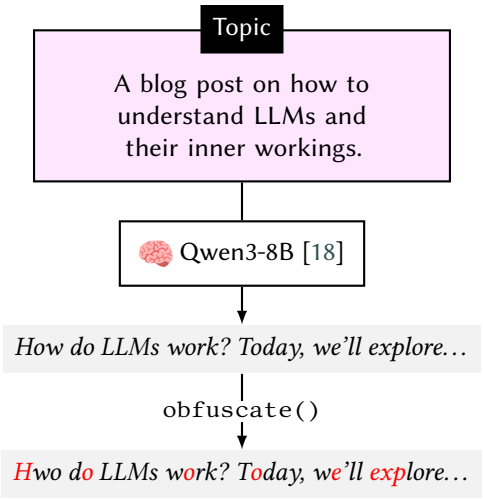

Figure 1: `Text Perturbation`. Character-level obfuscation through homoglyph substitution (red) and typo injection to mask AI-detectable traits.

3. Methods

We developed four different approaches to text humanization for the ELOQUENT 2025 Voight-Kampff shared task. Each method targets different aspects of what makes text appear human-written. Our methods range from simple post-processing attacks to more complex approaches that model human thinking processes.

All methods use Qwen3-8B [18] as the base language model but use different prompting strategies and post-processing techniques. The approaches go from character-level visual attacks (`Text Perturbation`) through language interference patterns (`Lost in Translation`) to cognitive modeling approaches (`Persona Immersion` and `ADHD Writing`). This varied set allows us to explore whether surface-level changes or deeper modeling of human writing processes works better for avoiding AI-generated text detection systems.[1]

3.1. `Text Perturbation`

The first of our approaches (Figure 1) is based on homoglyphs and typographical errors. This is our most obfuscative method, focusing on character-level visual attacks to avoid AI-generated text detection systems. This technique uses post-processing character substitution with visually similar Unicode characters from different scripts.

Our implementation uses a single-stage text generation process with Qwen3-8B at temperature 1.15, followed by systematic character replacement. We apply a 13% homoglyph attack rate using the SilverSpeak library [**?**], targeting different Unicode categories including Latin, Cyrillic, and Greek scripts. For additional hiding, we optionally add typos (swap, insert, delete, duplicate) at a 7% rate. The approach needs minimal prompt engineering, instead focusing on post-processing attacks that keep visual readability while potentially avoiding character-based detection methods.

For example, common substitutions include replacing Latin 'a' with Cyrillic 'а', 'e' with 'е', and 'o' with 'о'. While it might be possible to detect this approach using advanced systems that know about homoglyph attacks, it can work against detectors that rely mainly on surface-level text analysis without considering Unicode character variations.

[1]Prompts and examples shown in the figures have been simplified for illustration purposes. Complete implementation details are available in our supplementary materials.

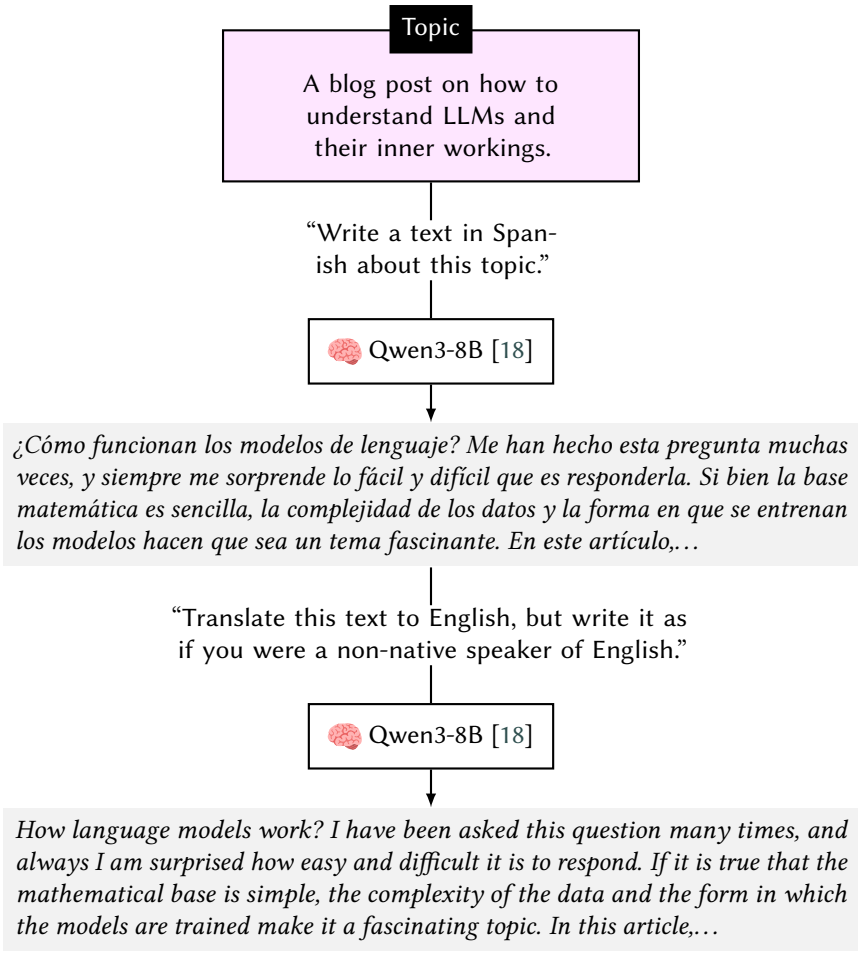

Figure 2: `Lost in Translation`. Two-stage cross-language generation that simulates non-native English patterns through back-translation, incorporating language-specific interference features.

3.2. `Lost in Translation`

Our translation-based approach (Figure 2) simulates non-native English writing patterns through a two-stage workflow that uses cross-language interference patterns. This method is based on the observation that non-native speakers produce text with specific language patterns that may not be well-represented in AI detection training data.

The first stage generates text directly in a randomly selected target language from our set of nine languages: French, Spanish, German, Italian, Portuguese, Dutch, Russian, Japanese, and Chinese. We use deterministic language selection based on item ID to ensure reproducibility. The second stage back-translates this text to English while specifically adding non-native speaker patterns typical of the source language.

Our prompts clearly instruct the model to include common language interference features such as false friends, literal translations, grammar interference, article and preposition errors, and tense confusion patterns. We use a temperature of 0.7 for initial generation and 0.8 for back-translation to introduce natural language variation, as the model's developers recommend a temperature of 0.7 for general usage. This approach is more efficient than a three-stage translation pipeline (English→target→English) while producing more authentic non-native patterns by generating content directly in the target language first, then translating with cultural and language context kept.

3.3. `Persona Immersion`

This approach (Figure 3) aims to create character-driven text through detailed character development. Rather than simply using style transfer techniques, this method seeks to perform deep character creation before writing, resulting in naturally human-like text variation.

Our two-stage process begins with detailed persona generation, where we create complete character profiles including background, age, personality quirks, life experiences, secret agendas, current situations, and most importantly, specific writing style characteristics. The persona generation stage uses temperature 1.5 to encourage creative character development, while the writing stage uses temperature 0.7 for consistent character voice maintenance.

The second stage involves writing from the persona's authentic perspective, with detailed prompting for character consistency and authenticity. Our prompts emphasize voice-specific instructions, encouraging the model to maintain the character's unique perspective, vocabulary choices, sentence structures, and thematic concerns throughout the text.

This approach creates natural human variation that goes beyond typical AI patterns because it models the complex relationship between a writer's personal characteristics and their writing style. By doing complete character development rather than surface-level style copying, the method produces believable, engaging content that reflects genuine human writing motivations and perspectives.

3.4. `ADHD Writing`

Our ADHD-inspired approach (Figure 4) models how internal thoughts influence writing style, word choices, and creative decisions, representing our most cognitively-motivated method. The core innovation lies in using random thought generation to introduce natural cognitive variability that makes writing more creative and human-like.

This three-stage process begins with persona generation that includes rich internal thought patterns and cognitive traits. The second stage generates text mixed with `[thought: ...]` tags that represent internal cognitive processes actively influencing writing decisions. Importantly, these thoughts don't just distract — they actively shape writing style, create topic associations, and drive creative leaps in the text. The final stage cleans the text by removing thought tags while keeping their influence on style and word choices.

Our implementation generates various types of thoughts including random tasks ("I need to feed the cat"), personal worries ("deadline stress"), sensory distractions ("that car outside is so loud"), memories, and writing ideas. Each thought immediately influences the next sentence through direct word associations or conceptual connections. For example, a thought about the writer having to feed their cat may lead to writing about "feeding your appetite for success". This creates a natural pattern where cognitive distractions drive creative word choices and topic transitions that mirror genuine human writing processes.

The key insight is that human writing is fundamentally influenced by concurrent internal thoughts and cognitive processes. By explicitly modeling these thought processes and their influence on writing decisions, we create text that shows the natural cognitive variability typical of human writing, making it more difficult for AI-generated text detection systems trained on coherent AI-generated content to identify.

4. Results

We present our results in this section, reporting on the five official metrics of the shared task we participated in (as defined by [1]):

- **Brier Score:** The complement of the traditional Brier score, equivalent to mean squared loss. Measures how well predicted probabilities match actual outcomes, with higher values indicating better calibration quality of probabilistic predictions.

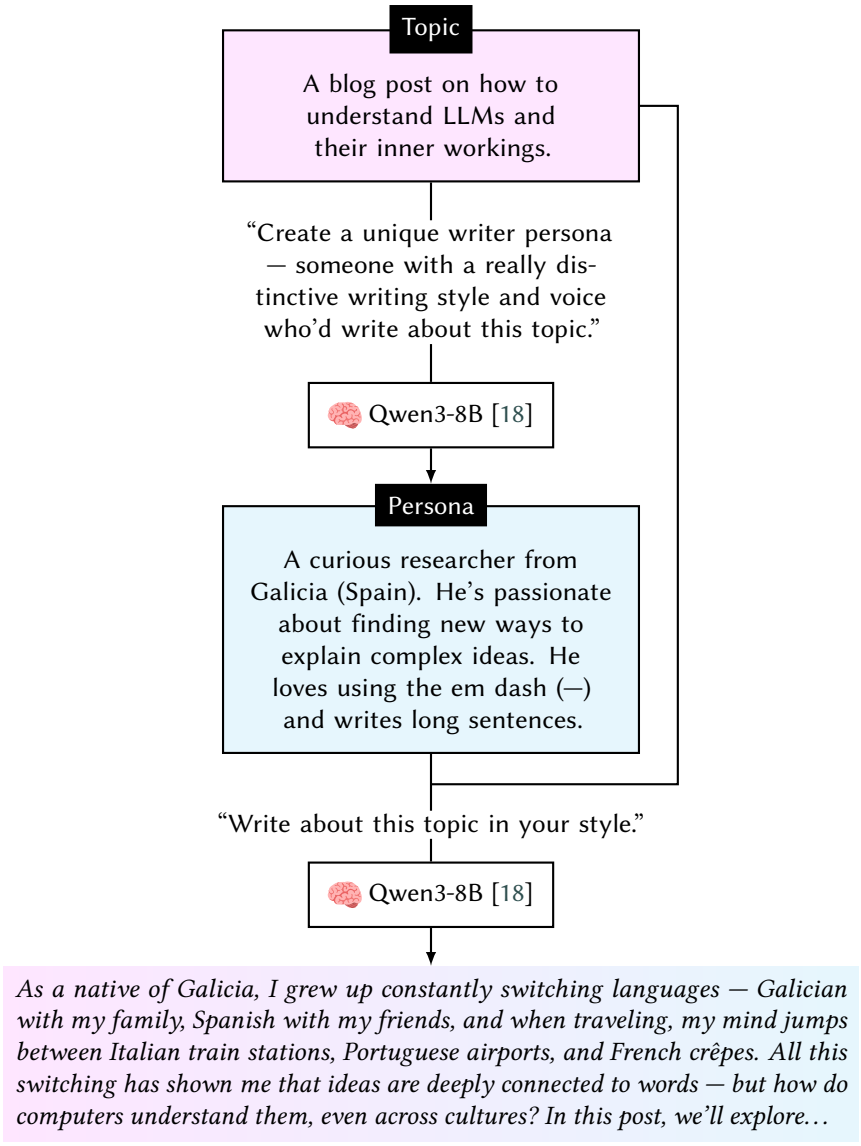

Figure 3: `Persona Immersion`. Character-driven text generation through detailed persona creation, producing a writing style that reflects human writing motivations and perspectives.

- **C@1:** A modified accuracy score that accounts for model confidence. Provides a single metric for classification performance that considers prediction confidence levels.
- **F1:** Harmonic mean of precision and recall for AI detection tasks. Balances both false positives and false negatives equally, providing a single metric for classification performance.
- **F0.5U:** A modified F-beta measure (precision-weighted) where beta=0.5 emphasizes precision over recall. This variant places greater weight on precision than standard F1 score.
- **Mean:** Average of normalized scores across all metrics, providing an overall performance summary that balances calibration, coverage, and classification accuracy.

Since the task objective is to evade AI detection, lower classification performance indicates better evasion success. We report the inverse of each metric to make higher values represent better evasion performance, which we believe is more intuitive for readers. Table 1 summarizes our results, showing the performance of our four approaches.

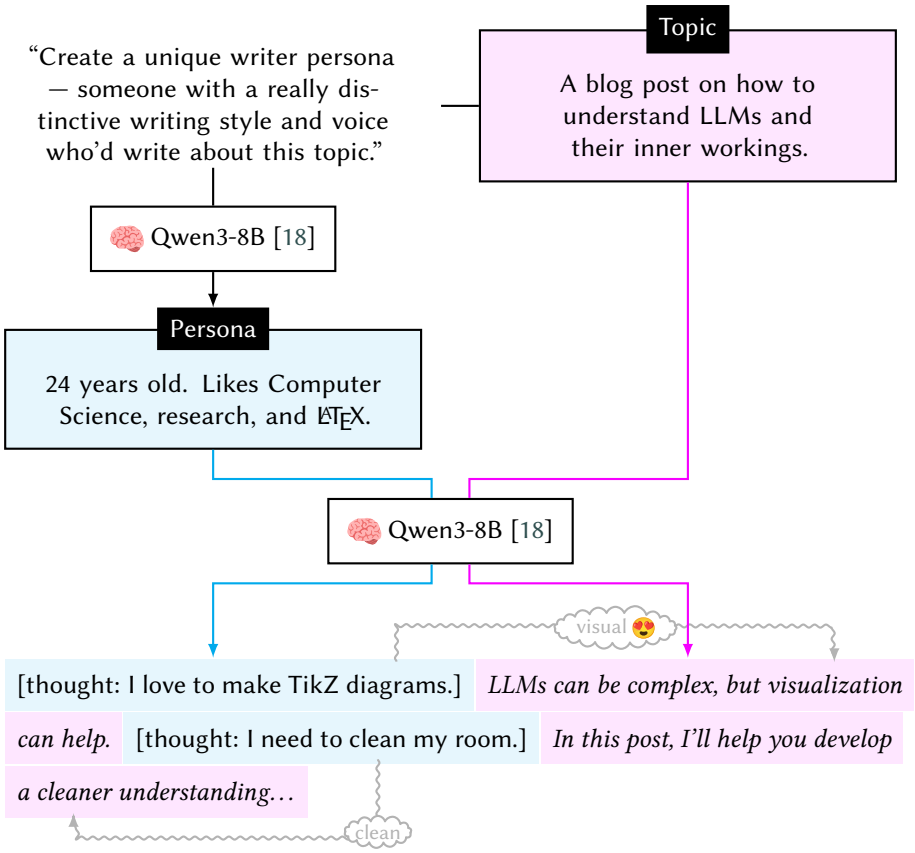

Figure 4: `ADHD Writing`. A persona generates internal thoughts that actively influence the writing process, creating natural variability in the output text.

Approach	Brier^{-1}	C@1^{-1}	F1^{-1}	F0.5U^{-1}	Mean^{-1}
`Lost in Translation`	0.565	0.627	0.559	0.457	0.642
`Text Perturbation`	0.402	0.451	0.405	0.346	0.521
`ADHD Writing`	0.316	0.358	0.274	0.177	0.425
`Persona Immersion`	0.219	0.269	0.208	0.140	0.367
`GPT-3.5` (baseline) [1]	0.056	0.032	0.042	0.034	0.037

Performance of our text humanization approaches (↑ higher is better).

All metrics are inverted so higher values indicate better evasion performance (successfully fooling AI detectors). `Lost in Translation` achieved the best overall performance, followed by `Text Perturbation`, `ADHD Writing`, and `Persona Immersion`. The GPT-3.5 baseline represents unmodified AI text for comparison.

5. Discussion

Our experimental evaluation reveals several important insights about the effectiveness of different humanization strategies for evading AI-generated text detection. The results shown in Table 1 demonstrate considerable variation in performance across our four text humanization approaches.

The `Lost in Translation` method achieved the best overall performance, with a margin of 0.121 mean score over the second-best `Text Perturbation` method, which is itself 0.096 better than the third-best `ADHD Writing` method. `Persona Immersion` performed the worst, with a mean score of 0.367, which is 0.274 lower than `Lost in Translation` and 0.058 lower than `ADHD Writing`.

Despite the performance variations across our methods, all four approaches demonstrate effectiveness at evading detection systems. The `GPT-3.5` baseline achieved only 0.037 mean score, indicating that

current detectors are highly effective at identifying AI-generated text. Our approaches significantly outperformed this baseline, with even our lowest-performing method (`Persona Immersion`) achieving nearly 10× better evasion (mean score 0.367). Notably, our best method (`Lost in Translation`) surpassed the previous year's winning team (mean score 0.056) by over 11×, a remarkable year-on-year improvement, and was this year's shared task second-best, only short of the best score (0.654) by 0.013.

The results highlight distinct performance patterns. We hypothesize that `Lost in Translation` and `Text Perturbation` outperform `ADHD Writing` and `Persona Immersion` because they operate through different mechanisms: the former methods either mask AI signatures using character alterations or introduce linguistic interference patterns that confuse detection, while the latter attempt to more authentically replicate human cognitive processes. Obfuscation methods create "artificial noise" that detectors struggle to parse, whereas cognitive modeling produces genuinely human-like text that paradoxically remains more detectable — possibly because current AI-generated text detection systems are specifically trained to distinguish between authentic human writing and AI attempts to mimic it.

Nonetheless, we see promise in further developing cognitive modeling approaches. Our preliminary `ADHD Writing` submission (not shown in Table 1) achieved scores of 0.845 / 0.814 / 0.859 / 0.904 / 0.685. After further refinement of our method, the results improved to the ones shown in Table 1, an improvement of 0.161 / 0.172 / 0.133 / 0.081 / 0.110. This is a very significant improvement, and we believe it shows great potential for further developing cognitive modeling approaches as they can produce text that is both more human-like to a human reader and more difficult for detectors to identify as AI-generated.

6. Conclusion

In this work, we presented four different approaches to text humanization for AI detection evasion, each targeting different levels of text manipulation. At the character level, `Text Perturbation` applies visual attacks by replacing letters with visually similar Unicode characters. At the linguistic level, `Lost in Translation` uses a two-stage process that generates text in other languages then back-translates to English with non-native patterns. At the cognitive level, we introduced two novel methods: `ADHD Writing` models internal thought processes that influence writing decisions through stochastic thought generation, while `Persona Immersion` creates detailed character profiles and writes from their authentic perspective to produce naturally human-like text variation.

We found that obfuscation-based methods (`Lost in Translation` and `Text Perturbation`) significantly outperformed cognitive modeling approaches (`ADHD Writing` and `Persona Immersion`). `Lost in Translation` achieved the best performance, making it the second-best submission in the ELOQUENT 2025 shared task. While cognitive modeling methods performed lower in the shared task metrics, they represent a fundamentally different approach that attempts to model genuine human cognitive processes rather than adding artificial noise to mask AI signatures.

Our results highlight important vulnerabilities in current AI-generated text detection systems, with all methods outperforming the baseline by factors of 10x to 17×. This work contributes to understanding how different humanization strategies — from surface-level character manipulation to deep cognitive modeling — can evade detection, which we hope can inform future research on building more robust AI-generated text detection systems.

Ethical Statement

This research is conducted purely for academic purposes to advance the understanding of AI-generated text detection systems and their vulnerabilities. Our primary goal is defensive: to identify current limitations in detection methods so that more robust systems can be developed, not to enable malicious use of text generation technologies.

We believe in responsible disclosure and make our methods and implementation publicly available to help the research community develop stronger detection mechanisms. By highlighting current blind

spots in AI detection systems, we aim to drive the development of detection systems that can account for advanced humanization techniques.

We do not encourage the use of these techniques for deceptive purposes such as academic dishonesty, misinformation campaigns, or any form of malicious content generation. The techniques presented here should be understood within the context of adversarial machine learning research, where understanding attack methods is essential for building robust defenses.

Our work follows established ethical guidelines for adversarial ML research, focusing on advancing the scientific understanding of AI detection capabilities and limitations. We hope that our findings will guide future research toward developing more resilient detection mechanisms that can better distinguish between human and AI-generated text, regardless of how advanced the humanization techniques are.

Supplementary Materials

For the purposes of reproducibility and to facilitate further research on the evasion and design of sturdier detectors, we make our implementation publicly available at https://github.com/ACMCMC/Voight-Kampff.

Disclosure of Interests

The authors have no competing interests to declare that are relevant to the content of this article.

Declaration on Generative AI

During the preparation of this work, the authors used Claude 4.0 Sonnet in order to: Formatting assistance, Grammar and spelling check, Improve writing style, Paraphrase and reword, Drafting content, Abstract drafting. After using these tools/services, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] J. Bevendorff, M. Wiegmann, J. Karlgren, L. Dürlich, E. Gogoulou, A. Talman, E. Stamatatos, M. Potthast, B. Stein, Overview of the "Voight-Kampff" Generative AI Authorship Verification Task at PAN and ELOQUENT 2024, in: G. Faggioli, N. Ferro, P. Galuščáková, A. G. S. Herrera (Eds.), Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2024, pp. 2486–2506. URL: http://ceur-ws.org/Vol-3740/paper-225.pdf.

[2] A. M. Sarvazyan, J. Á. González, M. Franco Salvador, F. Rangel, B. Chulvi, P. Rosso, Overview of AutexTification at IberLEF 2023: Detection and Attribution of Machine-Generated Text in Multiple Domains, in: Procesamiento del Lenguaje Natural, Jaén, Spain, 2023.

[3] A. M. Sarvazyan, J. Á. González, F. Rangel, P. Rosso, M. Franco-Salvador, Overview of IberAuTex-Tification at IberLEF 2024: Detection and Attribution of Machine-Generated Text on Languages of the Iberian Peninsula, Procesamiento del Lenguaje Natural 73 (2024).

[4] Y. Wang, J. Mansurov, P. Ivanov, J. Su, A. Shelmanov, A. Tsvigun, O. M. Afzal, T. Mahmoud, G. Puccetti, T. Arnold, et al., Semeval-2024 task 8: Multidomain, multimodel and multilingual machine-generated text detection, arXiv preprint arXiv:2404.14183 (2024).

[5] J. Karlgren, K. Artemova, O. Bojar, M. I. Engels, V. Mikhailov, P. Šindelář, E. Velldal, L. Øvrelid, Overview of ELOQUENT 2025: shared tasks for evaluating generative language model quality, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Sixteenth International Conference of the CLEF Association (CLEF 2025), Springer Lecture Notes in Computer Science, 2025.

[6] J. Bevendorff, Y. Wang, J. Karlgren, M. Wiegmann, A. Shelmanov, J. Mansurov, A. Tsivgun, I. Gurevych, P. Nakov, E. Stamatatos, M. Potthast, B. Stein, Overview of the "Voight-Kampff" Generative AI Authorship Verification Task at PAN and ELOQUENT 2025, in: 26th Working Notes of the Conference and Labs of the Evaluation Forum, CLEF 2025, CEUR-WS, 2025.
[7] D. Ippolito, D. Duckworth, C. Callison-Burch, D. Eck, Automatic detection of generated text is easiest when humans are fooled, arXiv preprint arXiv:1911.00650 (2019).
[8] I. Solaiman, M. Brundage, J. Clark, A. Askell, A. Herbert-Voss, J. Wu, A. Radford, G. Krueger, J. W. Kim, S. Kreps, et al., Release strategies and the social impacts of language models, arXiv preprint arXiv:1908.09203 (2019).
[9] J. Kirchenbauer, J. Geiping, Y. Wen, J. Katz, I. Miers, T. Goldstein, A watermark for large language models, in: International Conference on Machine Learning, PMLR, 2023, pp. 17061–17084.
[10] T. Lee, S. Hong, J. Ahn, I. Hong, H. Lee, S. Yun, J. Shin, G. Kim, Who wrote this code? watermarking for code generation, arXiv preprint arXiv:2305.15060 (2023).
[11] S. Gehrmann, H. Strobelt, A. M. Rush, Gltr: Statistical detection and visualization of generated text, arXiv preprint arXiv:1906.04043 (2019).
[12] E. Mitchell, Y. Lee, A. Khazatsky, C. D. Manning, C. Finn, Detectgpt: Zero-shot machine-generated text detection using probability curvature, in: International Conference on Machine Learning, PMLR, 2023, pp. 24950–24962.
[13] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, Journal of machine learning research 21 (2020) 1–67.
[14] A. Hans, A. Schwarzschild, V. Cherepanova, H. Kazemi, A. Saha, M. Goldblum, J. Geiping, T. Goldstein, Spotting llms with binoculars: Zero-shot detection of machine-generated text, arXiv preprint arXiv:2401.12070 (2024).
[15] H. Cerezo-Costas, P. Alonso-Doval, M. Hormazábal-Lagos, A. Creo, Telescope: Discovering multilingual llm generated texts with small specialized language models (2024).
[16] K. Krishna, Y. Song, M. Karpinska, J. Wieting, M. Iyyer, Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense, Advances in Neural Information Processing Systems 36 (2023) 27469–27500.
[17] T. Kumarage, P. Sheth, R. Moraffah, J. Garland, H. Liu, How reliable are ai-generated-text detectors? an assessment framework using evasive soft prompts, arXiv preprint arXiv:2310.05095 (2023).
[18] Q. Team, Qwen3 technical report, 2025. URL: https://arxiv.org/abs/2505.09388. arXiv:2505.09388.