Expert Systems With Applications 296 (2026) 129001

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Stylometry recognizes human and LLM-generated texts in short samples

Karol Przystalski [a,b,*], Jan K. Argasiński [b,c], Iwona Grabska-Gradzińska [b], Jeremi K. Ochab [b,d]

[a] Exadel Na Zjeździe 11, 30-527, Kraków, Poland
[b] Faculty of Physics, Astronomy and Applied Computer Science, Jagiellonian University, Łjasiewicza 11, 30-348, Kraków, Poland
[c] Sano Centre for Computational Medicine, Czarnowiejska 36/C5, 30-054, Kraków, Poland
[d] Mark Kac Centre for Complex Systems Research, Jagiellonian University, Łjasiewicza 11, 30-348, Kraków, Poland

ARTICLE INFO

Keywords:
Stylometry
Large language models
Machine-generated text detection
AI detection
Benchmark dataset

ABSTRACT

The paper explores stylometry as a method to distinguish between texts created by Large Language Models (LLMs) and humans, addressing issues of model attribution, intellectual property, and ethical AI use. Stylometry has been used extensively to characterise the style and attribute authorship of texts. By applying it to LLM-generated texts, we identify their emergent writing patterns. The paper involves creating a benchmark dataset based on Wikipedia, with (a) human-written term summaries, (b) texts generated purely by LLMs (GPT-3.5/4, LLaMa 2/3, Orca, and Falcon), (c) processed through multiple text summarisation methods (T5, BART, Gensim, and Sumy), and (d) rephrasing methods (Dipper, T5). The 10-sentence long texts were classified by tree-based models (decision trees and LightGBM) using human-designed (StyloMetrix) and n-gram-based (our own pipeline) stylometric features that encode lexical, grammatical, syntactic, and punctuation patterns. The cross-validated results reached a performance of up to 0.87 Matthews correlation coefficient in the multiclass scenario with 7 classes, and accuracy between 0.79 and 1. in binary classification, with the particular example of Wikipedia and GPT-4 reaching up to 0.98 accuracy on a balanced dataset. Shapley Additive Explanations pinpointed features characteristic of the encyclopaedic text type, individual overused words, as well as a greater grammatical standardisation of LLMs with respect to human-written texts. These results show – crucially, in the context of the increasingly sophisticated LLMs – that it is possible to distinguish machine- from human-generated texts at least for a well-defined text type

1. Introduction

In the dynamicaly expanding field of natural language processing, Large Language Models (LLMs), introduced by transformative models like GPT, have revolutionized approaches to analyze language by enabling machines to mimic human-like text generation. As the use of pretrained AI models becomes increasingly common, growing concerns around issues like ownership, attribution, intellectual property rights, and responsible usage highlight theurgent need for advanced methods to ensure ethical deployment and proper crediting of AI-generated work – alongside the development of reliable model detection tools.

The problem of stylometry and authorship attribution is a crucial aspect in this context. Stylometry, meaning the quantitative study of linguistic style patterns, is a valuable tool for effective text differentiation. Byexamining subtle differences in writing style, one can discover unique markers that distinguish one author from another. Stylometric features provide a detailed understanding of the characteristics of particular LLMs, offering a granular approach towards model identification. This not only facilitates differentiation but also enhances our comprehension of the linguistic idiosyncrasies ingrained in these models. The challenge lies in accurately attributing text to the correct author or model, especially as language models grow more sophisticated and their outputs increasingly indistinguishable from human writing. This paper explores the utilization of machine learning techniques in identifying stylistic markers and patterns that are characteristic for specific language models, augmenting our ability to differentiate them with greater accuracy. By focusing on properties such as the word choice and syntactic patterns, our aim is to uncover the linguistic fingerprints that distinguish one model results from another.

The exploration of stylometry in model detection and differentiation reaches far beyond technical considerations towards ethical implications. Understanding the characteristic stylometric properties of language models productions contributes to securing the responsible AI practices, promoting transparency and accountability. LLM safety and

* Corresponding author.
E-mail addresses: karol.przystalski@uj.edu.pl (K. Przystalski), jan.argasinski@uj.edu.pl (J.K. Argasiński), iwona.grabska@uj.edu.pl (I. Grabska-Gradzińska), jeremi.ochab@uj.edu.pl (J.K. Ochab).

https://doi.org/10.1016/j.eswa.2025.129001
Received 31 August 2024; Received in revised form 1 July 2025; Accepted 10 July 2025
Available online 14 July 2025

ethics are the most important concerns in this regard. Ensuring that language models are used ethically should account for such issues as bias, misinformation, and the potential for generating harmful content. By promoting stylometry, this paper aims to provide a distinctive perspective ,thereby contributing to a more comprehensive understanding of language model deployment in diverse applications. This approach not only improves our ability to safeguard intellectual property but also cultivates a culture of responsibility and trust in the AI community.

The research presented in this paper provides an innovative approach to distinguish between models. As we navigate the complex interplay of technology, ethics, and stylometry, our goal is to contribute to the responsible advancement of natural language processing technologies.

The main contributions of this paper are as follows:

1. **Application of stylometry to differentiate texts**: The paper applies stylometry to distinguish between texts generated by LLMs and human-authored texts. Stylometry, traditionally used for authorship attribution and literary style analysis, is shown to be effective in identifying writing patterns specific to LLMs.
2. **Creation of a diverse dataset**: The study constructs a dataset based on (a) human-written Wikipedia texts, (b) their summaries processed through various text summarization methods (T5, BART, Gensim, and Sumy), and (c) summaries generated by LLMs (GPT-3.5, GPT-4, LLaMa 2/3, Orca, and Falcon) prompted a given term only. This dataset allows for a comprehensive analysis of different text generation methods.
3. **High classification performance**: The study demonstrates that tree-based classifiers (decision trees and LightGBM) can achieve high performance in classifying texts, reaching up to 0.87 Matthews correlation coefficient in multiclass scenarios (with 7 classes) and up to 1.00 accuracy in binary classification (e.g., distinguishing Wikipedia from GPT-4-generated texts at 0.98 accuracy).
4. **Insights into LLM and human text characteristics**: The paper provides detailed insights into specific features that differentiate LLM-generated texts from human-authored texts. It highlights that LLM-generated texts tend to have more grammatical standardization and may overuse certain words or punctuation marks compared to human-written texts.
5. **Implications for ethical AI use**: The paper emphasizes the need for robust methods to track and identify AI-generated outputs to ensure ethical AI use, addressing concerns around model attribution, intellectual property, and responsible deployment of AI technologies.
6. **Potential for stylometry in future AI applications**: The research suggests that stylometry could continue to be a valuable tool for distinguishing machine-generated texts from human-authored ones, especially as LLMs become more sophisticated, highlighting its potential role in future AI applications and governance.

This manuscript is structured into six sections including Sections 1–6. In the Section 1 the rationale for the presented research is provided. In the Section 2 we present important background for our work. The design of our own experiments is detailed in Section 3. Section 4 of the classification are visualised in the next section. Finally the Sections 5 and 6 section include general remarks, known limitations and possible future directions for the research along with the inventory of crucial findings.

2. Related works

Stylometry, the study of linguistic style, has long been a important tool in authorship attribution, and its relevance has grown significantly with the advent of Large Language Models. As these models produce increasingly human-like text, the ability to distinguish between human-authored and machine-generated texts (MGT) becomes essential, not just for academic and forensic purposes, but also for ensuring the safety and ethical use of LLMs.

In this section we present works relevant to the theme of stylometry itself and related to MGT, particularly by LLMs; we mention research about stylometric modeling; and finally showcase papers that tackle the theme of safety and ethics regarding emerging generative linguistic tools.

2.1. Stylometry and author attribution

Neal et al. (2017) in *Surveying stylometry techniques and applications* provide an extensive overview of stylometry research, focusing on authorship attribution, verification, profiling, stylochronometry, and adversarial stylometry. The survey is in depth, covering different subtasks, datasets, experimental methods, and contemporary approaches. It includes detailed performance analysis taking into account 1000 authors using 14 different algorithms. The paper exposes key challenges such as scaling authorship analysis to account for a large number of authors with minimal text samples available. It also presents ongoing research challenges and showcases different software tools that support stylometric analysis - both open-source and commercial options.

A survey of modern authorship attribution methods (Stamatatos, 2009) gives an detailed presentaion of the various computational methods utilized in the field of authorship attribution. It traces the evolution of these methods from their inception in the 19th century, highlighted by the seminal study of Mosteller (1968), to the contemporary techniques that leverage statistical and computational approaches. This survey discusses the main characteristics, strengths, and weaknesses of modern authorship attribution methods.

2.2. Stylometric modeling

Paper titled *TDRLM: Stylometric learning for authorship verification by topic-debiasing* (Hu et al., 2023) proposes a “Topic-Debiasing Representation Learning Model” (TDRLM) to enhance stylometric authorship verification. The TDRLM utilizes a topic-debiasing attention mechanism with position-specific topic scores to mitigate the influence of topical bias in tokenized texts. Experimental results demonstrate that the TDRLM outperforms current state-of-the-art stylometric learning models and advanced language models, achieving the highest Area Under Curve (AUC) scores of 92.47 % for the Twitter-Foursquare dataset and 93.11 % for the ICWSM Twitter dataset. The study highlights that topic-related words can negatively impact machine learning algorithms for authorship verification, prompting the development of the TDRLM model to improve verification accuracy.

The evolution of current methods is well exemplified by a series of papers by Bhattacharjee et al. (2023), Kumarage et al. (2023), Kumarage and Liu (2023). Kumarage et al. (2023) and Kumarage and Liu (2023) used a fusion architecture of fine-tuned RoBERTa augmented with a combination of stylometric features – lexical, syntactic, and structural such as lexical richness, readability, punctuation counts, word / sentence / paragraph counts etc. Interestingly, the authors used as a baseline XGBoost with either stylometric or bag-of-word features, which allowed them to use SHAP explanation in the same vein as we do in the present paper. The fusion was proved beneficial especially for short texts (Twitter timelines) and limited training data, but out-of-distribution problems (cross-domain or unseen LLMs) remained challenging. To improve these issues, Bhattacharjee et al. (2023) turned away from stylometric classfiers to self-supervised contrastive learning and unsupervised domain adaptation techniques at the cost of losing the explainability.

2.3. Authorship-stylometry and LLMs

Large language models: A Survey by Zhao et al. (2023) provides a comprehensive overview of LLMs, their development, capabilities, and applications. The authors review notable LLMs, such as GPT, LLaMa, and

PaLM, discussing their design, strengths, and limitations. The paper explores various methods used for constructing and enhancing LLMs, examines key datasets utilized for training and evaluation, and assesses these models' performance across standard benchmarks. It highlights LLMs' significant advancements in natural language tasks, largely attributable to their training on massive datasets, reflecting the importance of data scale in model performance.

Argamon (2018) contributes with *Computational forensic authorship analysis: Promises and pitfalls* – a comprehensive examination of the techniques involved in computational authorship analysis, focusing on their application within legal and forensic contexts. Authors highlight how these methods have advanced to the point of being reliable enough for real-world legal applications, underscoring their evolution and growing acceptance in rigorous environments. Paper discusses various computational methods, detailing their underlying assumptions, necessary analytic controls, and the crucial reliability testing they must undergo to ensure their effectiveness. Moreover, the paper addresses the potential pitfalls of these techniques, offering guidance to practitioners on how to achieve results that are not only trustworthy but also comprehensible.

Learning stylometric representations for authorship analysis (Ding et al., 2017) explores a neural network approach to learn stylometric representations that capture various linguistic features such as topical, lexical, syntactical, and character-level characteristics. This methodology aims to improve the tasks of authorship characterization, identification, and verification by mimicking the human sentence composition process and incorporating these diverse linguistic categories into a distributed representation of words. The effectiveness of this approach is demonstrated through extensive evaluations across multiple datasets, including Twitter, blogs, reviews, novels, and essays, where the proposed models notably outperform traditional stylometric and other baseline methods. This research highlights the potential of neural networks in extracting and utilizing complex stylistic features for detailed authorship analysis in diverse textual domains.

With the question *Can large language models identify authorship?* Huang et al. (2024a) explores the capabilities of LLMs in performing authorship verification and attribution tasks without requiring domain-specific fine-tuning. The authors demonstrate that LLMs can effectively conduct zero-shot, end-to-end authorship verification and accurately attribute authorship among multiple candidates. Furthermore, the study sift how these models can offer explainability in their analysis, focusing particularly on the role of linguistic features.

Learning interpretable style embeddings via prompting LLMs (Patel et al., 2023) presents an innovative approach for deriving interpretable style embeddings, called LISA embeddings, from LLMs using prompting techniques. The authors address the challenge of uninterpretable style vectors commonly produced by current neural methods in style representation learning, which are problematic for tasks that require high interpretability like authorship attribution. To overcome this, they employ prompting to generate a synthetic dataset of stylometric annotations. This dataset facilitates the training of LISA embeddings, which are designed to be interpretable and useful for analyzing author styles in texts. Additionally, the authors contributed by releasing both the synthetic stylometry dataset and the LISA style models, enabling further exploration and development in the field of stylometry and style analysis.

A model-independent redundancy measure for human versus ChatGPT authorship discrimination using a Bayesian probabilistic approach (Bozza et al., 2023) introduces a novel method to distinguish between human-authored texts and those generated by AI models like ChatGPT. This approach utilizes a model-independent redundancy measure that effectively captures syntactical differences between human and machine-generated texts. The researchers employed a Bayesian probabilistic framework, specifically using the Bayes factor, to provide a robust and consistent classification criterion. This method proves particularly effective even with short text samples, demonstrating its potential utility in forensic and other analytical settings where distinguishing between human and AI authorship is crucial. The study highlights the applicability of this technique across various languages and text genres, indicating its broad potential for addressing the challenges posed by the increasing sophistication of MGT in academic and professional contexts.

Authors of *Who wrote it and why? Prompting large language models for authorship verification* (Hung et al., 2023) offer a new technique named `PromptAV`. This method utilizes Large Language Models (LLMs) to perform authorship verification effectively and with improved interpretability. Authors claim that the `PromptAV`, demonstrates improved performance compared to existing state-of-the-art baselines, particularly in scenarios with limited training data. It enhances interpretability by providing intuitive explanations, making it a promising tool for applications in forensic analysis, plagiarism detection, and identifying deceptive content in texts. This approach is meant to address the current limitations of traditional stylometric and deep learning methods, which typically require extensive data and lack explainability (e.g., Bhattacharjee et al., 2023).

The paper *T5 meets Tybalt: Author attribution in early modern english drama using large language models* (Hicke & Mimno, 2023) explores the application of LLMs for authorship identification in Early Modern English drama. The study finds that LLMs, specifically a fine-tuned T5-large model, can accurately predict the author of short passages and outperform traditional baselines like logistic regression, SVM with a linear kernel, and cosine delta. However, the presence of certain authors in the model's pretraining data introduces biases, leading to occasional confident misattributions of texts. This highlights both the promising potential and the concerning limitations of using LLMs for stylometric analysis in literary studies.

Finally, the paper titled *Detecting ChatGPT: A survey of the state of detecting ChatGPT-generated text* (Dhaini et al., 2023) provides an overview of current approaches for identifying text generated by ChatGPT. It highlights the challenges of distinguishing between human-written and machine-generated content, especially given the high fluency and human-like quality of ChatGPT outputs. The survey reviews various datasets specifically created for this detection task, examines different methodologies employed, and discusses qualitative analyses that help identify characteristics unique to ChatGPT-generated text. It also explores the broader implications for domains such as education, law, and science, emphasizing the need for effective MGT detection methods to maintain content integrity.

2.4. LLM detection benchmarks

A non-exhaustive list of datasets designed for MGT detection can be found in Wu et al. (2025). Most of these benchmark datasets are English-only (TuringBench Uchendu et al., 2021, CHEAT Yu et al., 2025, OpenLLMText Chen et al., 2023, GROVER Zellers et al., 2019, TweepFake Fagni et al., 2021, ArguGPT Liu et al., 2023b, MAGE Li et al., 2024a, PAN's Voight-Kampff Generative AI Detection task Bevendorff et al., 2024). Others include Spanish (AuTexTification Sarvazyan et al., 2023a), Chinese (HC3, HC3 Plus Guo et al., 2023; Su et al., 2024) or rarely they are multilingual (MULTITuDE Macko et al., 2023, M4 Wang et al., 2024b). The more recent ones are also multi-domain (Bevendorff et al., 2024; Li et al., 2024a,a; Macko et al., 2023; Sarvazyan et al., 2023a; Wang et al., 2024b), and use diverse LLM generators (see especially TuringBench Uchendu et al., 2021, MAGE Li et al., 2024a), which is particularly challenging to collect for multiple languages. Such multi-generator and multi-domain benchmarks allow one to test generalizability of the classifiers to unseen domains and unseen LLMs, a realistic scenario considering the rate of development of both closed and open-source LLMs. An even more comprehensive overview in terms of domains and languages is given in Macko et al. (2023). It is worth stressing that collecting and generating a well-controlled benchmark with multiple domains, generators, and languages is a considerable endeavour, as is well-known in corpus linguistics (Lüdeling & Kytö, 2008).

When evaluating MGT detectors, caution is required with regard to training on data external to these benchmarks, as some of the data were collected from other primary sources.

2.5. LLM detection methods

We refer to the *Overview of the "Voight-Kampff" Generative AI Authorship Verification Task at PAN and ELOQUENT 2024* by Bevendorff et al. (2024) as a recent benchmark of available methods. The submissions included mainly (i) perplexity-based systems, (ii) term-based systems (iii) and ensembles of both. The first ones rely mainly on the perplexities of a set of known LLMs (which is a limitation on its own) used as features for a classifier. The second class uses fine-tuned classifiers (often neural ones such as modified versions of BERT or their ensembles) with word embeddings or linguistic (stylometric) features; some other involve fine-tuned generative LLMs (regarded as unreliable by Wu et al., 2025). Several proposals, e.g., Guo et al. (2024a,b), Miralles et al. (2024), Yadagiri et al. (2024), made advantage of various sets stylometric features, while others found it useful to augment data or simply expand the training dataset. A noteworthy example is Lorenz et al. (2024), whose SVM classifier based on tf-idf features (sic!) by was ranked third beating all neural baselines and most of the neural-based competitors, mostly thanks to its robustness to the many obfuscation strategies designed by the Task's authors. *We regard our boosted trees classifier to follow a similar simplistic, inexpensive but effective design, often overlooked in recent research.*

Among many other methods reviewed by Wu et al. (2025) or Crothers et al. (2023) that were not represented in the PAN's task are the logits-based statistics. This family encompasses primarily zero-shot methods which, however, require access to a surrogate (often weaker) language model or, ideally, to the source LLM – hence they are dubbed white-box methods – to obtain its raw outputs that in turn allow to determine the likelihood of a text being generated by the LLM. The black-box statistical methods, on the other hand, do not require such access. Instead, given an original text they machine-regenerate it and subsequently compare these versions to obtain a similarity score.

Lastly, a whole strain of research has been MGT watermarking, wherein an imperceptible signature is embedded in the generated texts either by insertion of modified training samples (to defend against unauthorised LLM fine-tuning), manipulating the logits output distribution or token sampling, or character- and word-level replacements at the post-processing stage (especially useful when using black-box models). Lu et al. (2024), Sadasivan et al. (2025) report successful attack strategies on some of such watermarking schemes by iterative paraphrasing or probing a watermarked LLM to infer the signatures. *In the present paper, we assume that the adversary does not use a watermarked LLM.*

2.6. Robustness of LLM detection

A number of issues can degrade the performance of MGT detection. Wu et al. (2025) divided them into out-of-distribution challenges and attacks.

The first type includes detection across domains (i.e., usually different text types or genres involving different vocabulary, style, topics, and overall distribution of textual features), across languages (but also including texts written by non-native speakers), and across LLMs (i.e., generally detection of LLMs not available during the detector's training). In the latter case, there are some reports (Sarvazyan et al., 2023b) that supervised MGT detectors tend to generalise well across LLM scales but less so across their model families. For neural detectors, therefore, incorporating MGT from various sources is recommended, since an additional fine-tuning – even on small samples – can effectively alleviate this issue. *In our study, no data external to the dataset described in Section 3.1.1 is used for training, and we do not tackle the issue of cross-language detection* (reported as challenging by Bevendorff et al., 2024). *The cross-domain detection is tested on an existing benchmark by Sarvazyan et al. (2023a).*

The potential attacks include: paraphrase (where LLM output is subsequently paraphrased by another model in order to change the textual feature distribution of the original MGT; Sadasivan et al., 2025, see, e.g.,), adversarial (involving textual perturbations on the level of characters like various misspelling strategies, Stiff & Johansson, 2022, see , syntax, Bhat & Parthasarathy, 2020, see , or lexis, Crothers et al., 2022, see), prompt (using complex and varied prompts for MGT, see Guo et al., 2023; Liu et al., 2024b) attacks and models trained specifically to confound existing detectors. These attacks affect MGT detectors differently, depending on whether they are watermarking-based (specifically targeted by paraphrase and adversarial attacks), zero-shot or fine-tuned supervised detectors. The latter can effectively defend against some of these attacks by continually expanding training datasets, e.g., with adversarial examples. Notably, Bevendorff et al. (2024) report that in the joint PAN and ELOQUENT detection-obfuscation task, none of the obfuscation submissions managed to beat in terms of their difficulty simple methods such as Unicode obsfuscations or shortening text length. *In this study, we perform a one-step paraphrase attack (repeated paraphrasing is possible, as in Sadasivan et al., 2025), but we do not include any attacks in the training data.*

We do not cover the issue of mixed texts (human-edited MGT or LLM-edited human-written texts or texts whose separate parts come from either human or machine).

2.7. LLMs safety and ethics

The application of stylometry to LLMs is particularly important given the potential risks associated with their misuse, such as the generation of misleading information, deepfake text, or malicious content, as described below. We note, however, that mere detection that a text has been machine-generated – which is the objective of the present paper – does not imply that it is untrustworthy or malicious. Schuster et al. (2020) reported that, even though human language tends to stylistically change when deceiving, stylometry fails to detect malicious use of (now perhaps obsolete) LLMs, and that such issues involve a whole ecosystem of fraud (including among others fact-checking, users' feedback, and content propagation through social networks).

A survey of safety and trustworthiness of large language models through the lens of verification and validation (Huang et al., 2024b) provides a detailed examination of the safety and trustworthiness concerns associated with LLMs. It categorizes the known vulnerabilities of LLMs into three main types: inherent issues, external attacks, and unintended bugs. The study extends traditional verification and validation (V&V) techniques, commonly used in software and deep learning model development, to enhance the safety and reliability of LLMs throughout their lifecycle. Specifically, the survey discusses four complementary V&V techniques: falsification and evaluation, verification, runtime monitoring, and the implementation of regulations and ethical guidelines. These approaches are aimed at ensuring that LLMs align with safety and trustworthiness requirements, addressing both existing challenges and potential risks.

Another survey – *On large language model (LLM) Security and Privacy: The Good, the Bad, and the Ugly* (Yao et al., 2024) offers a detailed exploration of the security and privacy dimensions associated with LLMs. It assesses how LLMs can both enhance and threaten cybersecurity in various applications. The authors categorize their findings into beneficial uses ("The Good"), such as improving code security and data privacy, offensive applications ("The Bad"), like their use in user-level attacks due to their sophisticated reasoning capabilities, and inherent vulnerabilities ("The Ugly") that could be exploited maliciously. The survey emphasises the dual nature of LLMs in cybersecurity, showcasing their potential to advance security measures while also posing significant risks if not carefully managed and regulated. Furthermore, it identifies areas needing further research, such as model and parameter extraction attacks and the development of safe instruction tuning, underlining the complexity and evolving nature of LLM applications in security contexts.

Adversarial stylometry: Circumventing authorship recognition to preserve privacy and anonymity (Brennan et al., 2012) introduces the field of adversarial stylometry. This research area focuses on strategies like obfuscation and imitation to effectively counter authorship recognition methods, which are crucial for maintaining privacy and anonymity in written communication. The study demonstrates that manual techniques, where individuals intentionally alter their writing style, are particularly effective at evading detection, often reducing the accuracy of stylometric tools to the level of random guesses. Even individuals with no prior knowledge of stylometry or limited time investment can successfully employ these strategies. Additionally, the paper discusses the efficacy of various obfuscation techniques and highlights the limited effectiveness of automated methods such as machine translation.

ChatGPT and a new academic reality: Artificial Intelligence-written research papers and the ethics of the large language models in scholarly publishing (Lund et al., 2023) addresses the transformative effects of ChatGPT and similar large language models on academic and scholarly environments. Paper highlights several key concerns, including the potential for inherent biases in training data and algorithms that could compromise scientific integrity. Additionally, the it raises critical ethical issues, such as the ownership of content produced by these models and the proper use of third-party content, which are essential for maintaining transparency and fairness in academic publishing. The discussion extends to the responsibilities of researchers and publishers in ensuring that these technologies are utilized in a manner that upholds the ethical standards of scholarly work.

Last, but not least – *ChatGPT and the rise of large language models: the new AI-driven infodemic threat in public health* by De Angelis et al. (2023) examines the dual-edged impact of LLMs on public health. It acknowledges the potential of LLMs to aid scientific research through their ability to process and generate large amounts of data quickly. However, it critically highlights the risk of an "AI-driven infodemic", where the rapid and widespread dissemination of misinformation could be facilitated by these same technologies. The paper calls for urgent policy actions to mitigate these risks, emphasizing the need for a balanced approach in harnessing the benefits of LLMs while safeguarding against their potential to undermine public health and the integrity of scientific research. This includes the establishment of regulatory frameworks and the proactive monitoring of the use of LLMs to prevent the spread of false information.

3. Metholodogy

The process of the proposed solution is divided into several steps. The data acquisition and cleaning is explained in the first part of the chapter Section 3.1.1. The data was next extended by the summaries generated with various text summarisation methods Section 3.1.2. In the next step, we added additional short terms descriptions generated using different language models Section 3.1.3. Finally, based on the stylometric features, we differentiate between the texts generated by the models and the humans Sections 3.3, 3.4.

3.1. Dataset

3.1.1. Human texts: Wikipedia summaries

The dataset is based on Wikipedia terms using two different Python libraries: `datasets` from HuggingFace[1] (Lhoest et al., 2021) and `Wikipedia-API`. In the first method we used the dataset from 2022 named `20220301.simple`. Similarly to how Bevendorff et al. (2024) collected their data, our choice was dictated by the date of GPT-3.5 release, so that we avoid contaminating the human-authored and edited texts with MGT in the view of its increased presence in Wikipedia (Brooks et al., 2024). We obtained 1500 terms using the first method and 1048 terms using the second. The final dataset used in this paper consists of 2439 terms. The number is a result of the preprocessing part and the removal of all examples that did not meet one of the following requirements:

- the term text consists of at least 1100 alphanumerical characters, including punctuation marks,
- consists of at least 10 sentences,
- the first 10 sentences do not include references (bibliography).

Each term description that did not fulfil the above requirements was removed from the dataset. Before the above validation, non-latin letters were removed, and characters like duplicated whitespaces were removed, including brackets, semicolons, and dots. For classification purposes, the texts were shortened to a maximum of 18 sentences (this is the maximum number of sentences generated by GPT models, despite the 10-sentence limit). Additionally, we removed several outlying texts (which had high sentence counts, mainly due to improper spaCy segmentation of lists or enumerations), resulting in 2424 terms.

Table 1
Basic summariser dataset statistics: total number of tokens (including punctuation), fraction of punctuation tokens in the total token count, mean number of tokens in a sentence, number of sentences, and the maximal number of sentences. The numbers are averages and standard deviations across all documents.

	Gensim	Sumy	T5	BART
#tokens	71 ± 36	249 ± 62	220 ± 110	61 ± 12
fraction of punctuation [%]	13.4 ± 4.9	13.8 ± 4.0	28 ± 11	13.1 ± 4.3
#tokens / sentence	28.8 ± 9.8	24.4 ± 6.0	14.0 ± 13.0	19.7 ± 5.5
#sentences	2.5 ± 1.0	10.4 ± 2.2	21.0 ± 15.0	3.19 ± 0.7
max. #sentences	9	41	138	7

3.1.2. Text summarizers

We used four text summarisation methods for comparison: (1) A very popular Python method in the gensim library (ˇ Reh˚ u ˇ rek & Sojka, 2010). It is already outdated, as there are more complex methods based on transformers that reportedly give better results. (2–3) Transformer-based T5 (Raffel et al., 2020) and BART summarizers (Lewis et al., 2019). (4) The last summarization method is called sumy and is implemented in the `sumy`[2] library.

Every summarisation method is provided with the Wikipedia terms descriptions, but each has different parameters to be set. We tried to set such parameters to obtain a summary of about 10 sentences for each term. The gensim summarizer does have a 'number of sentences' parameter, but we did not set it to an exact number. It produced a sufficient number of sentences and in case it exceeded the limit, we just dropped the excess sentences. For the T5 and BART summarizes we got the best results with setting the maximum number of characters to 1000. The length penalty parameter and number of beans were left to the standard values of 2.0 and 4, respectively. Sumy has a parameter that allows one to set the exact number of sentences, which we set to 10.

Table 1 shows basic statistics of the dataset. In particular, T5 tends to produce a highly varying summary length, both in terms of tokens and sentences, and amount of punctuation. The reason is its failure, resulting in repetition of the same letters or words and a number of full stops. The other summarisers do not produce such artefacts, with BART generating a low number of short sentences, Gensim a low number of relatively longer sentences, and Sumy a larger number of sentences.

3.1.3. LLM-generated descriptions

Large language models were used to generate term descriptions from scratch, i.e., they were provided only with terms they were prompted to describe, but not with any part of the Wikipedia articles. We chose

[1] https://huggingface.co

[2] https://pypi.org/project/sumy

Table 2
Basic LLM dataset statistics: total number of tokens (including punctuation), fraction of punctuation tokens in the total token count, mean number of tokens in a sentence, number of sentences, and the maximal number of sentences. The numbers are averages and standard deviations across all documents.

	wiki	GPT-3.5	GPT-4	LLaMa 2	LLaMa 3	Orca	Falcon
#tokens	243 ± 56	223 ± 35	218 ± 27	152 ± 17	249 ± 35	144 ± 27	152 ± 39
fraction of punctuation [%]	13.5 ± 3.3	11 ± 2.3	12.2 ± 2.3	11.3 ± 2.9	11.8 ± 2.3	12.3 ± 3.4	10.7 ± 3.1
#tokens / sentence	24.0 ± 5.5	23.1 ± 2.8	22.3 ± 2.6	22.0 ± 3.5	24.8 ± 3.1	22.3 ± 3.9	22.3 ± 3.5
#sentences	10.19 ± 0.94	9.7 ± 1.2	9.78 ± 0.79	7.0 ± 1.3	10.04 ± 0.82	6.6 ± 1.7	7.0 ± 2.0
max. #sentences	18	18	14	16	17	16	17

Table 3
Benchmark used.

Benchmark	Human	LLMs	LLMs Type	Language	Domain
AuTexTification 1	~28k	~28k	BLOOM-1B1, BLOOM-3B, BLOOM-7B1, Babbage, Curie, text-davinci-003	English	tweets, how-to, news, legal, reviews

six language models, including the open and API-based ones. We used the ChatGPT API for two models: GPT-3.5-turbo, and GPT-4 (Liu et al., 2023a). LLaMa 2 and 3 with 7 and 8 billion parameters, respectively (Touvron et al., 2023). In this case, we used the Ollama[3] library. For the other two models: Orca (Mukherjee et al., 2023) and Falcon (Almazrouei et al., 2023) we used the GPT4All library (Anand et al., 2023). The models we used had 8 and 11 billion parameters, respectively. We used the GPT4All library to execute LLaMa2, LLaMa3, Orca, and Falcon. We used the default temperature value. Based on the documentation, the temperature was set to 0.7. For GPT3.5 and 4, the temperature setting was 0.7 (currently, the default value for the API for GPT4o and newer is set to 1; OpenAI, 2025).

We used two prompts that were sent to each of the models. The first one is a simple ask for a term explanation in 10 sentences. The exact prompt is the following: *Please describe in 10 sentences as plain text what <term> is*. The second prompt is a request for a text similar to the Wikipedia page. The exact prompt is the following: *Please describe as it would be the Wikipedia page in 10 sentences what <term> is*. The reason for having two prompts is that the term explanation can be potentially easier to be recognized when compared with a model-generated text. That is why the Wikipedia page-like response is compared.

Table 2 shows basic statistics of the dataset. In particular, the GPT models and LLaMa 3 kept very close to the 10-sentence limit, while the other models tended to produce shorter sentences and paragraphs.

3.2. Benchmarks

As external benchmark for machine-generated text detection we have used AuTexTification (Sarvazyan et al., 2023a). AuTexTification contains two shared tasks in two languages (English and Spanish): (i) MGT detection (a binary classification of texts written by human and a language model) and (ii) MGT attribution (classification of six models). Importantly, the first task uses a balanced multi-domain (tweets, how-to articles, legal documents, reviews and news) and multi-model (BLOOM: BLOOM-1B7, BLOOM-3B, BLOOM-7B1, and GPT-3: babbage, curie, and text-davinci-003) corpus, where only the first three domains appear in the training data and the last two in the test data Table 3.

3.3. Stylometry

We use two stylometry libraries: StyloMetrix (Okulska et al., 2023) and CLARIN-PL's stylometric pipeline (Ochab & Walkowiak, 2024).

3.3.1. StyloMetrix

StyloMetrix is an open-source stylometric text analysis library. Covers various grammatical, syntactic, and lexical aspects. StyloMetrix allows allowing feature engineering and interpretability. Stylometry involves the analysis of linguistic features to characterize the style of texts. Previous tools like 'stylo' package in R (Eder et al., 2016) provide quantitative text analysis but lack certain metrics and usability features that StyloMetrix offers. It is based on the spaCy model for English and generates normalized vectors for input texts, allowing comparison across texts of different lengths and genres. Vectors are designed to be interpretable at different levels. Metrics that are available for the English language:

- Detailed Grammatical Forms: Tenses, modal verbs, etc.
- General Grammar Forms: Consolidation of principal grammatical rules.
- Detailed Lexical Forms: Types of pronouns, hurtful words, punctuation, etc.
- Parts of Speech: General frequency calculation.
- Social Media: Sentiment analysis, lexical intensifiers, masked words, etc.
- Syntactic Forms: Questions, sentences, figures of speech, etc.
- General Text Statistics: Type-token ratio, text cohesion, etc.

The version of the library used in this paper provides 195 stylometry features. It also supports model explainability and is available in multiple languages, making it a valuable tool for linguistic analysis and machine learning applications.

3.3.2. CLARIN-PL's stylometric pipeline

We used a modular Python pipeline for interpretable stylometric analysis developed for CLARIN-PL[4] (Ochab & Walkowiak, 2024). The pipeline connects text preprocessing and linguistic feature extraction with various NLP tools, classifiers, an explainability module, and visualization. At present, we use spaCy model 'en_core_web_lg' for preprocessing steps (including tokenisation, named entity recognition, dependency parsing, part-of-speech and morphology annotation), Light Gradient-Boosting Machine (LGBM) (Ke et al., 2017) as the state-of-the-art boosted trees classifier, Shapley Additive Explanations (SHAP) (Lundberg et al., 2020) for computing explanations, and Scikit-learn (Pedregosa et al., 2011) for feature counting and cross-validation. The visualisation functions, showing general and detailed explanations of what linguistic features make texts differ, utilise spaCy and SHAP.

As in previous works (Argasiński et al., 2024; Ochab & Walkowiak, 2024), we decided to use (i) tree models, which are easily interpretable

[3] https://ollama.com/

[4] https://gitlab.clarin-pl.eu/stylometry/cl_explainable_stylo

and for which the explanations can be computed fast, (ii) feature engineering approach, where the features are rooted in linguistic knowledge but can be generated programmatically. Specifically, the features passed to the classifier were the normalised frequencies of:

- lemmas (from uni- to trigrams), excluding named entities,
- part-of-speech tags (from uni- to trigrams), excluding named entities and punctuation,
- dependency-based bigrams,
- morphological annotations (unigrams) excluding punctuation,

No culling (i.e., ignoring tokens with document frequency strictly higher or lower than the given threshold) was performed. We specifically excluded punctuation marks after initial experiments, as the features containing them tended to express some of the Wikipedia preprocessing artefacts. Such features can also be expressive of some artefacts in LLM processing, such as the 'SPACE' token (a redundant whitespace character, e.g., at the beginning of a paragraph or a second one between words), as in the Results. The whitespace token is used in the multiclass classification, but in the binary classification, we remove all 83 features containing it.

3.4. Classification

The first method chosen is a simple decision tree classifier from the popular Python `sklearn`[5] library. It was used with the default parameters such as the Gini impurity method, the minimum samples in the split set to 2, and the split strategy set to *best*. The test and train sets were used in a split of 70 % to 30 % with a 10-fold cross-validation (CV).

The LGBM classifier was used with the following settings: DART boosting, maximal depth of the tree model ("max_depth" = 5), maximal number of leaves per tree ("num_leaves" = 5), default number of boosting iterations, increased "learning_rate" = 0.5, enabled bagging (randomly selecting part of data without resampling with "bagging_freq" = 3 and "bagging_fraction" = 0.8), and number of classes in the multiclass scenario ("num_class" = 7). Further hyperparameter optimisation is possible, but was not performed in this study.

We used the group cross-validation scheme by using 10-fold CV for test error estimation. Group CV makes sure that a given topic of the summary never appears both in the train and test set. The reported scores are averages over the CV loop. Training and test set sizes in each fold were 4390 and 488 samples for binary classification and, respectively, 15365 and 1708 for multiclass classification.

For the binary classification scenario, we provide accuracy, since all the datasets are exactly balanced. For the multiclass scenario, we provide the Matthews correlation coefficient (MCC) as the performance metric.

4. Results

We have performed the classification on the same dataset using two different classifiers and two different stylometric libraries. For the sake of comparison, we also included the recognition of summarization methods with LLMs.

4.1. Binary classification with decision trees

The decision trees performed worse compared to LGBM. This was the first experiment to test if the models can be recognized between each other and the Wikipedia text. The results for two prompts explained in the previous section are given in Table 4.

Decision trees are known to be used to measure feature importance. In our first experiment the most significant stylometric features are as follows:

[5] https://scikit-learn.org

Table 4
Accuracy of binary text classification with decision trees. Each table entry corresponds to a task, where class 1 and 2 are column and row model labels, respectively. Texts generated by different prompts are analysed separately.

	wiki	GPT-3.5	GPT-4	LLaMa 2	LLaMa 3	Orca	Falcon
Prompt #1							
wiki	1.0	0.8170	0.8693	0.9596	0.8324	0.9605	0.9286
GPT-3.5		1.0	0.7154	0.9263	0.6869	0.9273	0.8804
GPT-4			1.0	0.7740	0.5754	0.8124	0.7658
LLaMa 2				1.0	0.8323	0.5693	0.6922
LLaMa 3					1.0	0.8525	0.8081
Orca						1.0	0.6082
Falcon							1.0
Prompt #2							
wiki	1.0	0.8230	0.8419	0.9451	0.7991	0.9475	0.9030
GPT-3.5		1.0	0.6428	0.8884	0.6291	0.8905	0.8271
GPT-4			1.0	0.8380	0.5688	0.8501	0.8008
LLaMa 2				1.0	0.8657	0.5256	0.6809
LLaMa 3					1.0	0.8778	0.8160
Orca						1.0	0.6701
Falcon							1.0

- L_ADJ_COMPARATIVE – adjectives in comparative degree,
- L_FUNC_T – function words types,
- FOS_FRONTING – fronting,
- L_TYPE_TOKEN_RATIO_LEMMAS – type-token ratio for words lemmas.

These four features were used for the binary classifications. The worst results were achieved for the second prompt with the following pair of classes: Orca and LLaMa 2, LLaMa 3 and GPT-4, Falcon and LLaMa 2, and Falcon and Orca. In the first two cases the results were about 52 % and 56 % accordingly. We can conclude that in both cases the recognition is very limited or even fails. Majority of model binary recognitions are between 70 % and 85 %. The best results are for distinguishing LLaMa 2 from GPT-3.5, and Orca from GPT-3.5 for both prompts. The accuracy is about 92 % for the first prompt, and about 89 % for the second prompt. What is worth attention are the results in recognition of models' generated text and the Wikipedia text where the lowest accuracy is about 73 %, but the majority is above 85 %, with best results achieved for Orca and LLaMa 2, 95 % and 96 % accordingly.

4.2. Binary classification with LGBM

4.2.1. StyloMetrix features

Table 5 shows CV-averaged accuracy between all pairs of classes. The LLM most often misclassified as the real Wikipedia are GPT-4 and LLaMa 3 (cf. Tables 5–7. LLaMa 2 and Orca were the hardest to distinguish. GPT models and LLaMa 3, as well as Orca and Falcon are also confused often.

4.2.2. Frequency-based features

Table 5 shows the accuracy between all pairs of classes. LLMs are hardly confused with the real Wikipedia at all. As before, the most often confused pairs of models were GPT models and LLaMa 3, as well as the triplet LLaMa 2, Orca, and Falcon.

4.3. Multiclass classification with LGBM

The performance of LGBM classifier is reported in Table 6. Visibly, it heavily depends on the number and selection of the features used. The small variance of the results across CV folds indicates that the results are robust.

4.3.1. StyloMetrix features

Table 7 shows the normalised confusion matrix. Interestingly, the man-made Wikipedia texts are recognised better than any of the LLMs.

The largest confusion exists between LLaMa 2 and Orca models and between LLaMa 3 and the GPT models. The LLM most often misclassified as the real Wikipedia is GPT-4.

4.3.2. Frequency-based features

Table 7 shows the normalised confusion matrix. Again, Wikipedia has the highest accuracy and the LLM most often misclassified as it is GPT-4. The most often confused pairs of models are Falcon and Orca, GPT-3.5 and GPT-4, LLaMa 3 and GPT-3.5.

4.4. Robustness testing

For brevity, we provide robustness testing only for the LGBM model with frequency-based features, and only for the binary detection of human- and machine-generated texts. Since the test sets contains only one class (machine-generated texts), we provide the value of recall of that class and the validation recall for comparison, see Table 8.

Table 5
Accuracy of binary text classification with LGBM using StyloMetrix features. Each table entry corresponds to a task, where class 1 and 2 are column and row model labels, respectively. The results are averages over 10 CV folds.

	wiki	GPT-3.5	GPT-4	LLaMa 2	LLaMa 3	Orca	Falcon
Stylometrix features							
wiki		0.97	0.94	0.99	0.95	0.99	0.98
GPT-3.5			0.87	0.99	0.88	0.99	0.98
GPT-4				0.99	0.85	0.99	0.98
LLaMa 2					0.99	0.77	0.90
LLaMa 3						0.99	0.98
Orca							0.87
Falcon							
Frequency-based features							
wiki		0.99	0.98	1.00	0.99	1.00	1.00
GPT-3.5			0.90	0.98	0.91	0.98	0.97
GPT-4				0.99	0.93	0.99	0.98
LLaMa 2					0.99	0.79	0.84
LLaMa 3						1.00	0.99
Orca							0.86
Falcon							

Table 6
Multiclass generators performance [MCC].

	StyloMetrix	Frequencies
CV average	0.72	0.87
CV min.	0.71	0.86
CV max.	0.74	0.89
dummy baseline	0.00	0.00
number of features	196	3000

Table 7
Confusion matrix in the multiclass classification scenario for LGBM using StyloMetrix and frequency-based features.

	wiki	GPT-3.5	GPT-4	LLaMa 2	LLaMa 3	Orca	Falcon
Stylometrix features							
wiki	0.90	0.011	0.040	0.0078	0.030	0.0062	0.0082
GPT-3.5	0.017	0.78	0.089	0.0041	0.094	0.0090	0.0082
GPT-4	0.044	0.11	0.73	0.0082	0.10	0.0029	0.0090
LLaMa 2	0.0082	0.0033	0.0057	0.72	0.0033	0.19	0.071
LLaMa 3	0.044	0.097	0.11	0.0049	0.74	0.0016	0.0082
Orca	0.013	0.0049	0.0033	0.22	0.0037	0.67	0.085
Falcon	0.011	0.011	0.0082	0.078	0.011	0.087	0.79
Feature-based features							
wiki	0.98	0.0012	0.011	0.0	0.0033	0.0016	0.0
GPT-3.5	0.0037	0.83	0.078	0.00041	0.063	0.016	0.0057
GPT-4	0.015	0.069	0.85	0.0025	0.041	0.011	0.0074
LLaMa 2	0.0	0.0	0.0	0.96	0.0	0.011	0.031
LLaMa 3	0.015	0.065	0.048	0.00082	0.85	0.0029	0.015
Orca	0.00082	0.0041	0.0049	0.014	0.0	0.88	0.097
Falcon	0.0012	0.0057	0.0033	0.0094	0.0029	0.11	0.87

Table 8
Recall on (i) unseen LLMs and on texts paraphrased with (ii) DIPPER and (iii) Parrot. (i) Unseen LLM test set contained only the given model, while training set contained the other models and the human Wiki summaries. (ii)-(iii) Paraphrased test set contained only the paraphrases, while the training set contained all the models and human texts. The values are mean and standard deviation over CV folds.

Recall [%]	Validation	Test	DIPPER	Parrot
GPT-3.5	99.11 ± 0.36	99.6 ± 0.17	99.95 ± .062	99.971 ± 0.044
GPT-4	99.49 ± 0.24	88.2 ± 1.2	99.95 ± .045	98.81 ± 0.14
LLaMa 2	99.17 ± 0.39	99.61 ± 0.11	99.922 ± .065	99.992 ± 0.017
LLaMa 3	99.24 ± 0.24	94.13 ± 0.72	99.9 ± .064	99.736 ± 0.098
Orca	99.18 ± 0.32	99.79 ± 0.16	99.87 ± .17	99.996 ± 0.013
Falcon	99.14 ± 0.30	99.691 ± 0.056	99.81 ± .11	99.955 ± 0.03

4.4.1. Testing on unseen models

The test assumes that in training the model can only access data on man-made texts and on five out of six LLMs. The features are chosen and fixed at this stage and the training recall is computed. Testing is performed on the single LLM previously unseen by the model. The cross-validation here regards the training only, i.e., for each unseen LLM there were 10 classifiers trained on subsets of the training set (and evaluated on the validation set as shown in Table 8), while the test set remained the same. The standard deviations of recall are computed over these 10 folds.

The largest drop in performance can be seen for GPT-4 and LLaMa 3 models.

4.4.2. Testing on paraphrased texts

Following Sadasivan et al. (2025) we performed a paraphrase attack using DIPPER (Krishna et al., 2023), a 11B paraphrasing model, and Parrot (Damodaran, 2021), a T5-based paraphrase model. Reportedly, in a small sample, DIPPER had shown in human evaluation that the content was satisfactorily preserved in about 70 % of the samples and grammar quality was satisfactory in 88 %. We used no recursive paraphrasing.

The test assumes that, in training, the model can access all unparaphrased data (human and all six LLM-generated texts). The features are chosen and fixed at this stage. Testing is performed on all paraphrases. As above, the cross-validation here regards the training only. The results are shown in Table 8.

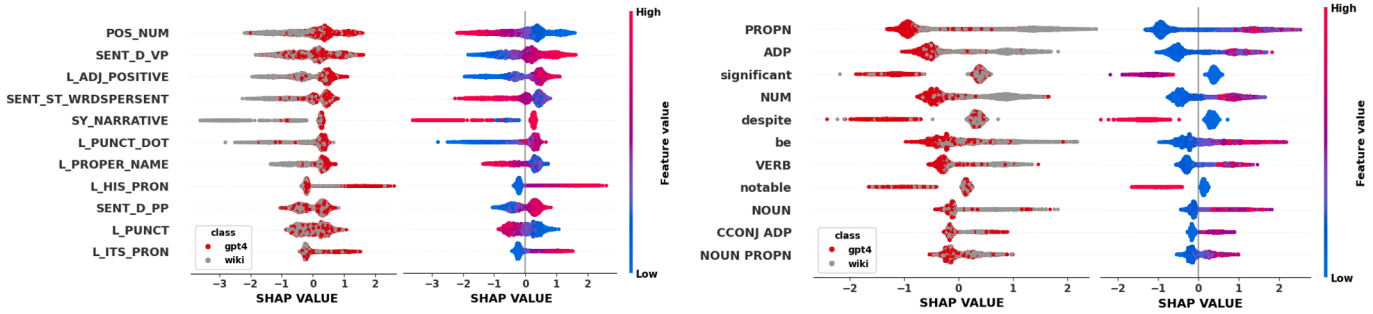
In general, paraphrasing resulted in a higher detection rate than for unparaphrased texts. The only exception is Parrot paraphrasing GPT-4, where a less than 1 % drop in recall occurred.

4.4.3. Testing on cross-domain benchmark

The results obtained on the AuTexTification task are shown in Table 9. Our classifiers used only the training data provided in the task description according to the task constraints.

4.5. Explainability

This section reports the results of SHAP explanations. In the binary classification, we provide only a single example to show the effectiveness of the explanations. For that purpose the GPT-4 model was chosen as the hardest one to detect. In this case, the positive or negative direction of SHAP values points toward one or the other class. In the multiclass classification, the obtained explanations take into account all the LLMs. In that case, the absolute values of SHAP show which features explain which model to what extent. Depending on the model's idiosyncrasies, a feature can explain several models well, but others poorly. Moreover, some models may be explained by few very strong features



(a) (b)

Fig. 1. Explanations for binary classification between the Wikipedia and GPT-4 using (a) StyloMetrix and (b) frequency-based features. Only the first 10 most important features are shown. Each point is a 10-sentence sample describing a given term coloured by: (**Left**) the sample's class, and (**Right**) its feature's intensity. The left plots indicate whether positive or negative SHAPs point toward GPT or the real Wikipedia.

(i.e., with large SHAP values), while others may need numerous features contributing only small fractions to the explanation, as visible in Fig. 2.

For each classification scenario, SHAPs were collected and averaged across all CV folds.

4.5.1. Binary classification

Here we present only the example of classifying the Wikipedia and GPT-4, as shown in Fig. 1(a) and (b), respectively, for StyloMetrix and frequency features. Analogous analyses can be repeated for the other pairs of classes. Let us recall, that punctuation (including the SPACE token) was excluded from the frequency features. Like above in the multiclass scenario, one notices features representing proper names (L_PROPER_NAME, PROPN), dates and other numerals (POS_NUM, NUM), etc. GPT-4 strikingly tends to abuse words like 'significant', 'notable' or 'despite'. Its usage of grammatical features (i.e., POS n-grams), however, tends to be strongly frequency-standardised, visible as the red bulks of the distributions in contrast to the long grey outlying distributions for the Wikipedia.

4.5.2. Multiclass classification

In Fig. 2 the ten most important StyloMetrix and frequency features are shown.

The StyloMetrix features include (in the order of importance): number of function word types, number of words in narrative sentences, the type-token ratio for word lemmas, statistics between noun phrases, fronting, difference between the number of words and the number of sentences, punctuation – dots, punctuation, punctuation – commas, and numerals; see (Okulska et al., 2023) for feature descriptions. The frequency features include single part-of-speech tags such as: whitespace, nouns, adpositions, proper nouns, verbs, adjectives, and determiners; POS bigrams such as: noun followed by a whitespace; and single lemmas such as: 'despite', 'and'.

Table 9
AuTexTification benchmark results. Macro-F1 score. For comparison submissions to Sarvazyan et al. (2023a) are presented: the top-ranked, the logistic regression (LR) baseline and results by Mikros et al. (2023) (an ensemble of stylometric features and transformers).

Classifier	F1
Top	0.81
LR	0.66
(Mikros et al., 2023)	0.61
StyloMetrix	0.48
Frequencies	0.54

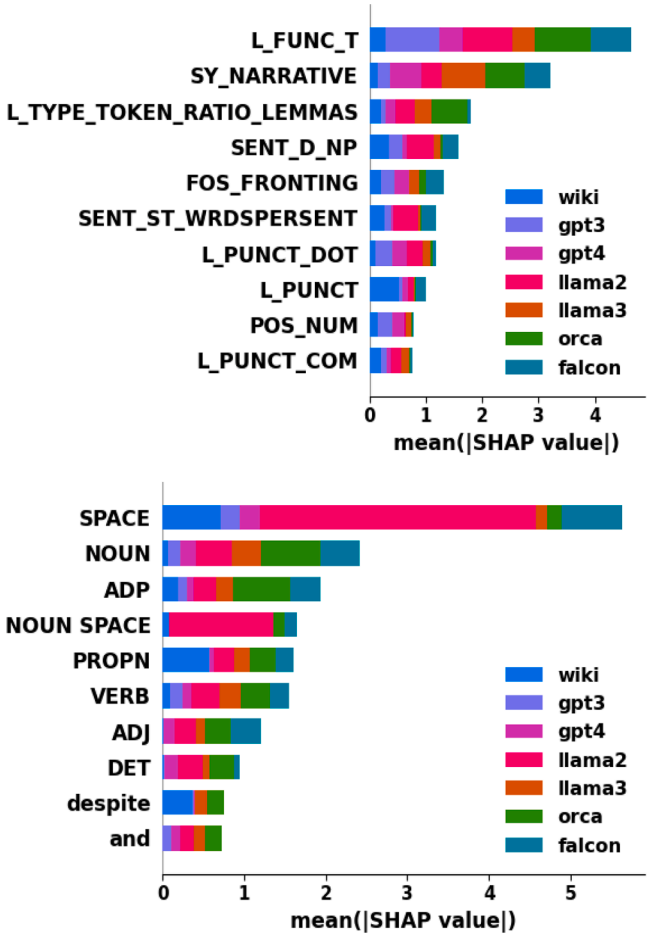


Fig. 2. General explanations for multiclass classification. The first 10 most important features according to the absolute values of SHAP are shown. SHAP values were averaged over CV folds. Colours indicate the importance of a feature for recognising a particular class.

The feature explanations are not model-agnostic. Notice the dates in the Wikipedia sample (POS_NUM), lower number of punctuation marks for LLaMa 2 than for Wikipedia (see numbers next to L_PUNCT in Fig. 3a), SENT_D_NP having similar values in all three cases. Also, looking at Fig. 3b, one notices a significantly larger number of proper nouns and dates in Wikipedia (PROPN – also in bigrams – and NUM),

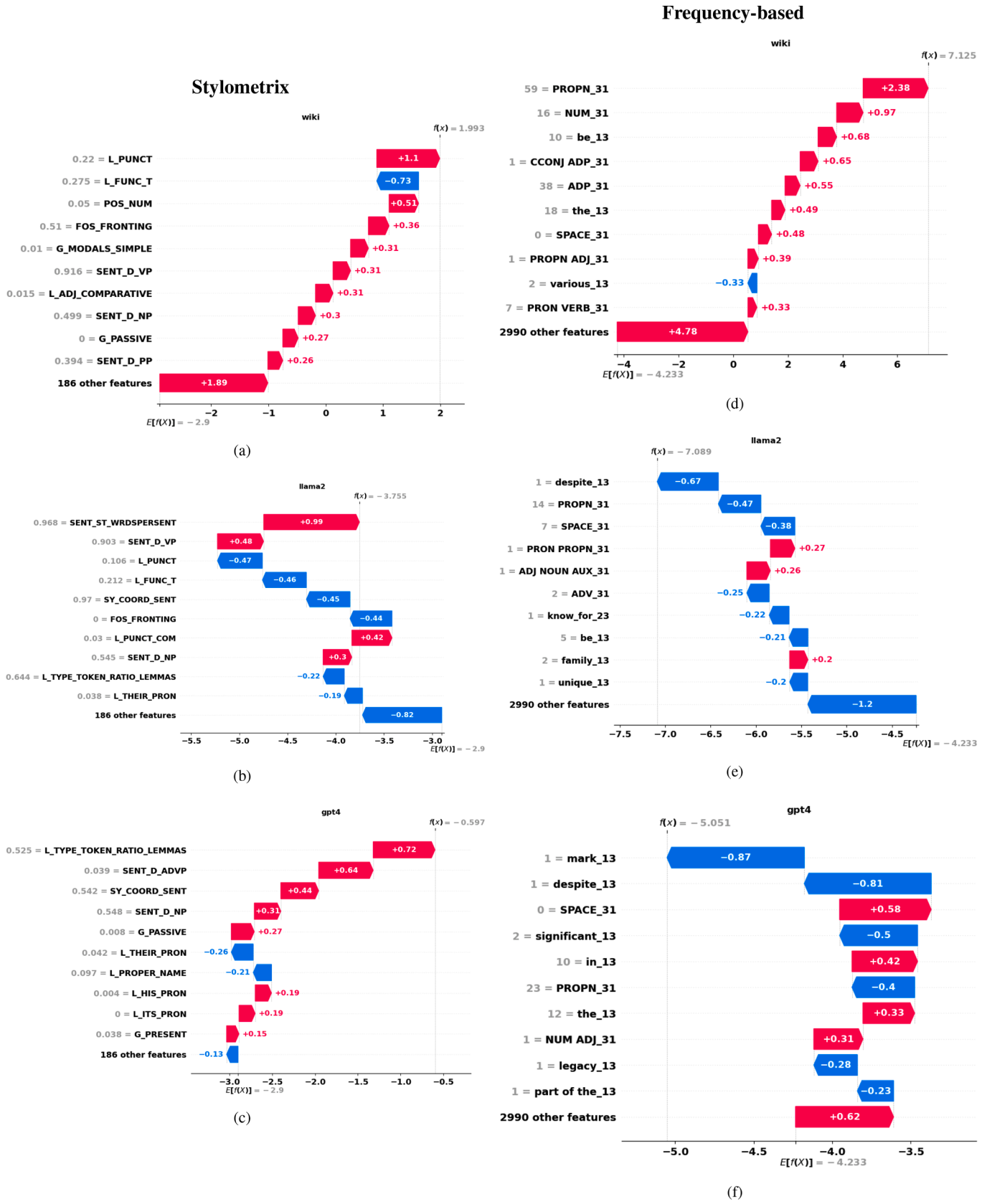


Fig. 3. Local explanations of 10 most important StyloMetrix (a-c) and frequency features (d-f) in multiclass classification for text samples describing the term 'The Swarbriggs'. Only selected models are shown. For this term, the Wikipedia was classified correctly, GPT-4 was misclassified as the Wikipedia, and LLaMa 2 was misclassified as Orca. Grey numbers to the left indicate feature values in this particular text sample. The positive/negative SHAP values do not point strictly to any particular class (in the multiclass scenario) but they tend to be higher for Wikipedia and GPT models and lower for worse models. Fig. 5.

Fig. 4. Text sample from the Wikipedia with highlighted text spans corresponding to important frequency features from Fig. 3. Note that the lack of features (like SPACE) cannot be highlighted but is important to the classifier.

redundant spaces in LLaMa 2 (SPACE), and other singular features. One can notice that the explanations for the GPT models are more distributed (no single feature with a huge SHAP value) than the other models.

It is worth recalling that models trained on different data subsets (CV folds) contribute to the SHAP values in Fig. 2, while the SHAP values presented in Fig. 3 correspond to a single classifier whose test set contained the selected texts. In addition, the SHAP values in Fig. 3 are averaged over classes, but explanations for each class can be obtained separately.

Text samples (corresponding to the term 'The Swarbriggs') are shown in Figs. 4–6, where also the frequency features most important to the classifier have been marked.

4.6. Summarization methods comparison

The text summarization methods are used only for comparison reasons as to what popular methods perform in a binary classification against language models. Similarly to LGBM experiments, this was also performed on the first prompt, because there were no significant differences between both prompts in the decision tree classification. The classification of summarization methods was also performed using the decision tree method. The results are given in Table 10.

The worst-recognized model is GPT-4, as the comparison with the Wikipedia summary is only about 72 %. This indicates that this model can simulate the way human summarizes Wikipedia pages, but it is important to highlight that it was also the most complex model used in

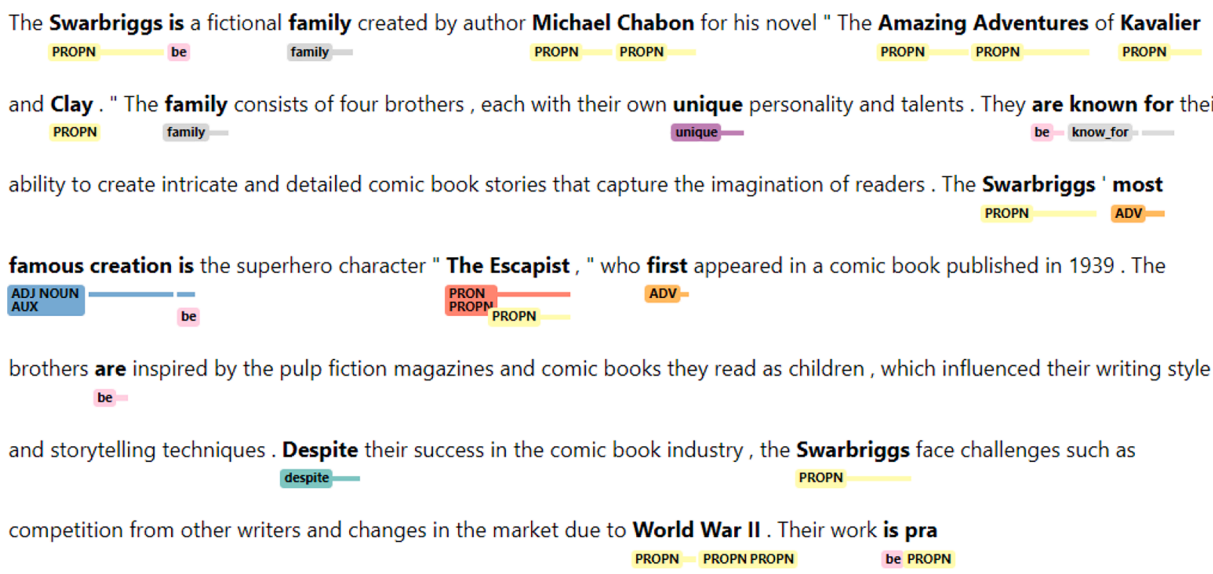


Fig. 5. Text sample from LLaMa 2 with highlighted important frequency features.

our experiment. The other questionable recognitions were obtained for Orca compared to BART summarizer and Sumy summarizer compared to Wikipedia, about 75 % both. The other results vary between 80 % and 92 %. The best results were achieved in a binary classification to recognize the text generated by one summarizing methods from another summarizing method or LLM, like T5 summarizer and Sumy summarizer – with about 92 %, Sumy summarizer and Orca – about 92 %, GPT-4 and BART summarizer – about 95 %, LLaMa 3 and BART summarizer – about

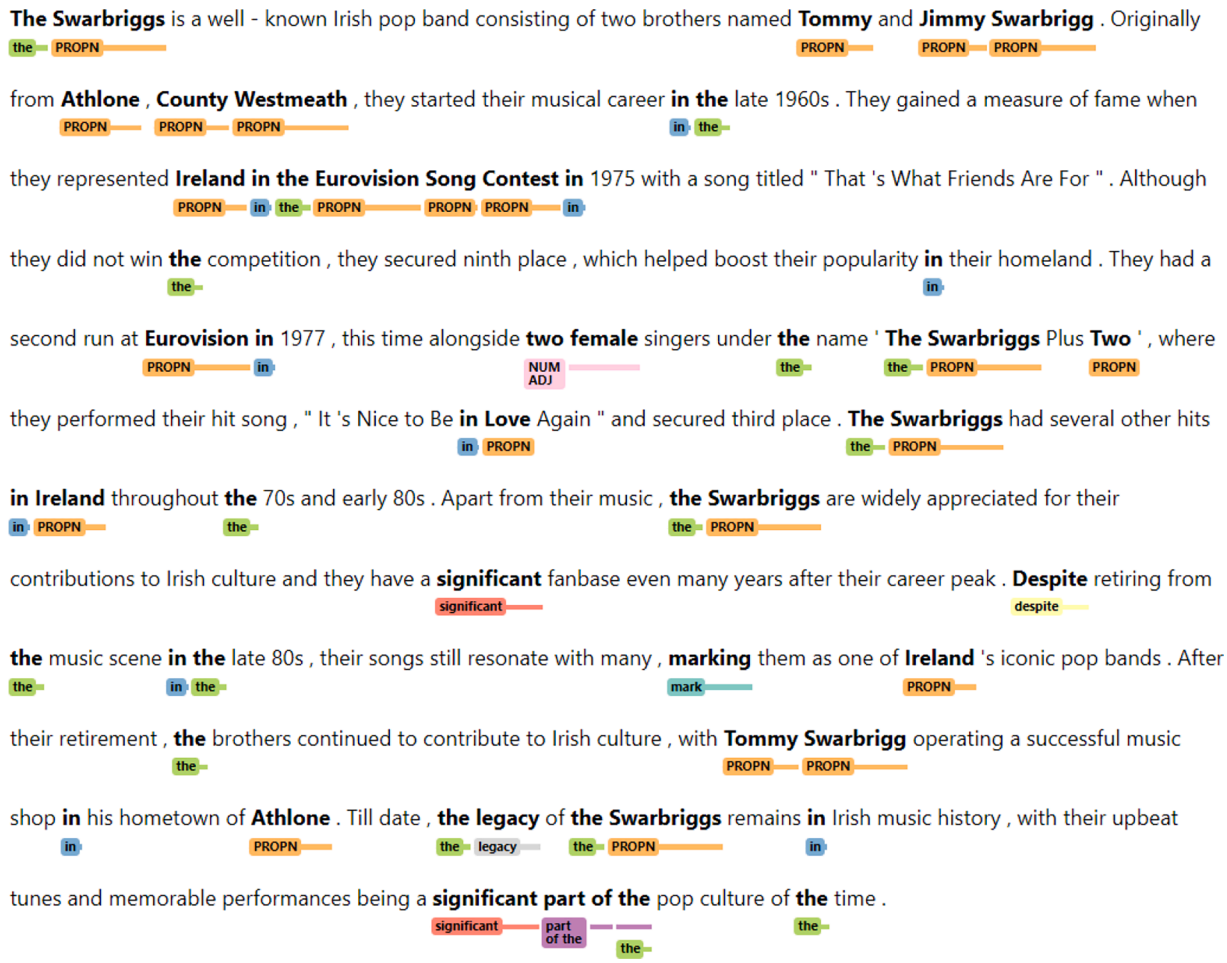


Fig. 6. Text sample from GPT-4 with highlighted important frequency features.

Table 10
Accuracies of summarization methods text generation recognition using decision trees. Average over 10 CV folds.

	Wikipedia sum	Sumy	T5	BART	Gensim
Sumy	0.7540	1.0			
T5	0.864	0.9221	1.0		
BART	0.9664	0.9735	0.9381	1.0	
Gensim	0.9611	0.9753	0.948	0.7954	1.0
GPT-3.5	0.7540	0.8398	0.8889	0.9648	0.9634
GPT-4	0.7283	0.8071	0.8967	0.9501	0.9469
LLaMa 2	0.8924	0.9129	0.9034	0.8135	0.8986
LLaMa 3	0.6865	0.79	0.8757	0.9622	0.9509
Orca	0.9046	0.9223	0.9107	0.7561	0.8717
Falcon	0.8362	0.8875	0.8353	0.7935	0.8769

96 %, BART summarizer and Wikipedia – about 96 %, and BART summarizer and Sumy summarizer – about 97 %.

5. Discussion

Generally, the results show that in a well-defined text generation task LLMs can be easily distinguished from the man-made texts and from each other with a boosted tree classifier even with very few features (196 for StyloMetrix in English) and even for extremely short texts (10 sentences). More features, coming mostly from grammatical tagging, lead to even better – indeed, almost perfect – results.

From multiclass explanations: it seems that well-performing models do not have single strongly recognisable features, but their style is more dispersed among many quantified features. Moreover, the explanations are not general, but may vary depending on the model, hence, the multiclass training is indispensable. These plots summarise all folds in the cross-validation loop, so the results are also stable in terms of different training/test splits. Interestingly, simple features such as the number of punctuation marks matter. The whitespaces found in LLaMa 2 were actually double spaces between tokens or a space at the beginning of the text. The number of full stops appears as a distinguishing feature, possibly because the LLMs tend to stop generating the text in the middle of the sentence. This might also affect 'the difference between the number of words and the number of sentences' (SENT_ST_WRDSPERSENT) as well as some other features. Wikipedia descriptions tend to be more fact-packed (dates and proper nouns) than LLM-generated ones. The distributional plots from binary classification between Wikipedia and GPT-4, suggest that the LLM favours certain individual words and is more standardised than Wikipedia in terms of grammatical structures (represented by frequencies of part-of-speech n-grams) – perhaps an expected outcome since the Wikipedia text samples were authored by many people. These conclusions come from explanations collected in the cross-validation loop, so they are stable in terms of different training/test splits.

The summarisation methods achieve similar results in the decision tree experiment. We can conclude that we will achieve similar results in LGBM for the summarisation methods. It indicates that the summarisers do have a distinctive way of text summarisation that can be found using stylometry. Succinctness of BART and artefacts in T5 explains their high recognisability. Sumy is the most successful due to its flexibility in choosing the summary length.

5.1. Limitations

The limitations of the present paper concern mainly the material of the analysis. Firstly, the results and specific conclusions refer only to the chosen text type, i.e., introductions to Wikipedia articles, which are expected to conform to an encyclopaedic style: plain, factual and partly formulaic. Some of the most distinctive features reflect that, and cannot be generalised to classifying other text types. However, the analytic pipeline is generic, including the engineered features, which have been designed and used in the context of literary texts. Whether the cross-domain classification with this type of model is robust is at this time debatable, taking into account our preliminary results in the AuTexTification task (Sarvazyan et al., 2023a), but also results of others that have tried utilising stylometric features (Mikros et al., 2023) with results below simple baselines. One can frame the issue of domain dependence of the model in various ways: both training and testing in another domain or cross-domain detection (i.e., detection on unseen domains), and which part of the data is unseen (whether the human- or machine-generated texts or both). Depending on these choices, the attack scenario is more or less realistic.

Secondly, the language of the text samples is limited to English only. The precise lexical, grammatical and other complex features will differ for other languages. Performance of stylometric tools has been known to depend heavily on language and specifically on language type (analytic, synthetic, etc.), see, e.g. (Eder, 2011; Evert et al., 2017). However, the LLMs are also best developed in English (Li et al., 2024b) and hence we expect it to be the most challenging setting for classification. The text processing pipeline we used strictly depends on the availability of NLP tools (like POS taggers, dependency parsers, NERs, etc.) for a given language. The frequency features at this moment depend on spaCy, which currently provides more or fewer tools for about 24 languages. In the case of StyloMetrix features, even though they also depend on the models distributed by spaCy, they were custom-designed for Polish, English, German, Ukrainian and Russian only.

Thirdly, the collection of Wikipedia samples is multi-authorial in at least two ways: each article could have been written by a different author, but also a single article probably has been edited by several authors – of various individual styles and linguistic competency. Reproducing this variety has not been explicitly stated in any of the prompts.

Besides the single domain constraint, the robustness testing of the stylometric detection methods and their explanations is limited in terms of the variability of LLM generation. One can envisage generating multiple text versions with: varying hand-crafted or machine-paraphrased prompts, persona-assigned prompts (Liu et al., 2024a; Przystalski et al., 2025; Wang et al., 2024a), as well as the same prompt with varying LLM parameters. In our Wikipedia-based dataset, however, we do not expect much variance by varying the prompts, due to the constraints of the encyclopaedic style. In this case, we consider varying and tuning the prompts a less realistic attack scenario.

One should also note, that the multiclass classification is performed on a closed set of classes. Although adding unseen models does not change the task in binary classification (human vs. machine), in the multiclass case the task would change to an open set problem.

While the binary classification task (human vs. machine detection) remains unchanged with the addition of new generative models, the multiclass setting fundamentally changes: the task becomes an open-set classification problem (Geng et al., 2020), where the classifier additionally has to recognize samples that belong to unknown or novel classes. This issue is out of scope of the present paper, however, having a good close-set classifer is helpful in the open-set problems (Vaze et al., 2022).

The language and type of the human-made texts additionally influence the availability of the training data for the classifier. In our case, the training set for the Wikipedia sample was about a million word tokens (plus another quarter million punctuation marks). Not all text generation tasks allow this large corpora, however, this is still the order of magnitude of a long novel (like classic Samuel Richardson's *Clarissa*, with about 1.1 million tokens with punctuation) or several shorter ones. The frequency-based pipeline has been successfully tested before on two novels of joint size of under 60 thousand word tokens (Ochab & Walkowiak, 2024) and even shorter (Argasiński et al., 2024), three research papers yielding jointly 3400 tokens.

In the subtask 1 of "Voight-Kampff Generative AI Detection at PAN and ELOQUENT 2025" (Bevendorff et al., 2025) (essays, news, and

Table 11
Commercial applications predictions.

Model	Falcon	GPT-3	GPT-4	LLaMA 2	LLaMA 3	Orca	Human
GPTZero prompt #1	98 %	100 %	96 %	98 %	98 %	98 %	100 %
GPTZero prompt #2	100 %	99 %	93 %	95 %	100 %	99 %	N/A
HIX prompt #1	3 %	5 %	3 %	5 %	4 %	0 %	100 %

fiction genres as well as their obsfuscated versions) our pipeline (Ochab et al., 2025b) without hyperparameter optimisation has reached $F_1 = 0.823$ against the top result $F_1 = 0.898$. A recent zero-shot detection solution achieved in Sun and Lv (2025) accuracy of 90,6 %. The average accuracy on three different structured texts datasets is 79,26 %. Both show that stylometric approach achieves better results. In Xu et al. (2024) FreqMark method was proposed for LLM generated text using frequency-based watermark. It shows robustness against paraphrasing and other attack methods. The accuracy of 98 % shows that not only stylometry-based methods perform well on paraphrased text. Stylometric methods were used in Al-Shaibani and Ahmed (2025) for text fingerprints to detect text generated by LLMs. For four models: two Arabic and two general text, the accuracy vary for LLM models text generation detection between 88.23 and 98.07 % for social media content.

5.2. Commercial applications

We have tested two different commercial solutions. The test set was a randomly chosen set of 100 prompt results for each model separately. We also included 100 human-written text. The number of chosen samples is caused due to the high costs of each request of such tools. The results are presented in Table 11. noa (2025) performs very well and predicts if the text is written by a human or a model almost perfectly. The disadvantage of this solution, similar to almost every commercial solution, is that it is a binary classification: human or AI. The models correctly identified the generated text as LLM generated, but there are no details on exactly what LLM was used for generation. Our proposed solution performs slightly worse than GPTZero, but it is a multi-label classifier. For comparison, we have tested the classifier developed by HIX (2025). It performs very poorly and classifies almost every sample as written by humans.

6. Further works

The results show that we can use stylometry for the English language to distinguish between LLMs and human written text. The next steps would be to perform the analysis in other languages, including low-resource languages.

The second way to extend this research is to use other stylometry libraries, classification methods, and more complex language models. Based on the results presented, the more complex models show that they are harder to differentiate from human written text compared to the less complex models.

The third vector of further research is to extend the feature list with features encoding long memory and correlations in text, such as fractal-based features. As stylometry seems to be a good choice, there might be other ones that might be more precise.

Another extension can be the use of stylometry together with neural embeddings. A hybrid approach might increase the accuracy in generated text recognition.

Finally, the classification explanations obtained by different methods and from different classifiers should be verified for their consistency and stability across various domains.

Source code

The data files and code used for text preprocessing and analysis can be found at https://osf.io/dfz6k/ (Ochab et al., 2025a).

The source code for text generation can be found in the repository: https://github.com/kprzystalski/stylometry-llm. It includes: the URLs to the libraries, the code to get the data, preprocess it, and execute the experiment. It comes with setup guidelines, contains all parameters set for each model.

CRediT authorship contribution statement

Karol Przystalski: Supervision, Writing – original draft, Writing – review & editing, Investigation; **Jan K. Argasiński:** Writing – original draft, Writing – review & editing, Investigation; **Iwona Grabska-Gradzińska:** Writing – original draft, Writing – review & editing, Investigation; **Jeremi K. Ochab:** Writing – original draft, Writing – review & editing, Conceptualization, Methodology, Investigation, Visualization, Data curation, Software.

Data availability

We published the data.

Declaration of competing interest

During the preparation of this work, the authors used GPT and Writefull LLM models to improve the readability, style and grammar of the text. After using this tool/service, the author(s) reviewed and edited the content as needed and take full responsibility for the content of the published article.

Acknowledgement

We thank Tomasz Walkowiak for many insightful comments. The research for this publication has been supported by a grant from the Priority Research Area DigiWorld under the Strategic Programme Excellence Initiative at Jagiellonian University.

JKO's research on the stylometric pipeline was financed by European Funds for Smart Economy, FENG program, CLARIN – Common Language Resources and Technology Infrastructure, project no.FENG.02.04-IP.040004/24-00.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857533 and from the International Research Agendas Programme of the Foundation for Polish Science No MAB PLUS/2019/13.

The publication was created within the project of the Minister of Science and Higher Education "Support for the activity of Centers of Excellence established in Poland under Horizon 2020" on the basis of the contract number MEiN/2023/DIR/3796.

References

Al-Shaibani, M. S., & Ahmed, M. (2025). The arabic AI fingerprint: Stylometric analysis and detection of large language models text. arXiv preprint arXiv:2505.23276.

Almazrouei, E., Alobeidli, H., Alshamsi, A., Cappelli, A., Cojocaru, R., Debbah, M., Goffinet, É., Hesslow, D., Launay, J., Malartic, Q., Mazzotta, D., Noune, B., Pannier, B., & Penedo, G. (2023). The falcon series of open language models.

Anand, Y., Nussbaum, Z., Treat, A., Miller, A., Guo, R., Schmidt, B., Community, G., Duderstadt, B., & Mulyar, A. (2023). Gpt4all: An ecosystem of open source compressed language models. https://arxiv.org/abs/2311.04931.

Argamon, S. (2018). Computational forensic authorship analysis: Promises and pitfalls. *Language and Law/Linguagem e Direito*, *5*(2), 7–37.

Argasiński, J. K., Grabska-Gradzińska, I., Przystalski, K., Ochab, J. K., & Walkowiak, T. (2024). Stylometric analysis of large language model-generated commentaries in the context of medical neuroscience. *International Conference*, (pp. 281–295). https://doi.org/10.1007/978-3-031-63775-9_20

Bevendorff, J., Wang, Y., Karlgren, J., Wiegmann, M., Tsivgun, A., Su, J., Xie, Z., Abassy, M., Mansurov, J., Xing, R., Ta, M. N., Elozeiri, K. A., Gu, T., Tomar, R. V., Geng, J., Artemova, E., Shelmanov, A., Habash, N., Stamatatos, E., Gurevych, I., Nakov, P., Potthast, M., & Stein, B. (2025). Overview of the "Voight-Kampff" generative AI authorship verification task at PAN and ELOQUENT 2025. In G. Faggioli, N. Ferro, P. Rosso, & D. Spina (Eds.), *Working notes of CLEF 2025 – conference and labs of the evaluation forum* CEUR Workshop Proceedings. CEUR-WS.org.

Bevendorff, J., Wiegmann, M., Karlgren, J., Dürlich, L., Gogoulou, E., Talman, A., Stamatatos, E., Potthast, M., & Stein, B., et al. (2024). Overview of the "Voight-Kampff" generative AI authorship verification task at PAN and ELOQUENT 2024. In G. Faggioli, N. Ferro, P. Galuscáková, & A. G. S. d. Herrera (Eds.), *Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), Grenoble, France, 9–12 September, 2024* (pp. 2486–2506). CEUR-WS.org (*vol. 3740*). CEUR Workshop Proceedings. https://ceur-ws.org/Vol-3740/paper-225.pdf.

Bhat, M. M., & Parthasarathy, S. (2020). How effectively can machines defend against machine-generated fake news? an empirical study. In A. Rogers, J. Sedoc, & A. Rumshisky (Eds.), *Proceedings of the first workshop on insights from negative results in NLP* (pp. 48–53). Online: Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.insights-1.7

Bhattacharjee, A., Kumarage, T., Moraffah, R., & Liu, H., et al. (2023). ConDA: Contrastive domain adaptation for AI-generated text detection. In J. C. Park, Y. Arase, B. Hu, W. Lu, D. Wijaya, A. Purwarianti, & A. A. Krisnadhi (Eds.), *Proceedings of the 13th international joint conference on natural language processing and the 3rd conference of the Asia-Pacific chapter of the association for computational linguistics (Volume 1: Long Papers)* (pp. 598–610). Nusa Dua, Bali: Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.ijcnlp-main.40

Bozza, S., Roten, C.-A., Jover, A., Cammarota, V., Pousaz, L., & Taroni, F. (2023). A model-independent redundancy measure for human versus chatGPT authorship discrimination using a bayesian probabilistic approach. *Scientific Reports*, *13*(1), 19217.

Brennan, M., Afroz, S., & Greenstadt, R. (2012). Adversarial stylometry: Circumventing authorship recognition to preserve privacy and anonymity. *ACM Transactions on Information and System Security (TISSEC)*, *15*(3), 1–22.

Brooks, C., Eggert, S., & Peskoff, D. (2024). The rise of AI-generated content in wikipedia. In L. Lucie-Aimée, A. Fan, T. Gwadabe, I. Johnson, F. Petroni, & D. van Strien (Eds.), *Proceedings of the first workshop on advancing natural language processing for wikipedia* (pp. 67–79). Miami, Florida, USA: Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.wikinlp-1.12

Chen, Y., Kang, H., Zhai, V., Li, L., Singh, R., & Raj, B. (2023). Token prediction as implicit classification to identify LLM-generated text. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Proceedings of the 2023 conference on empirical methods in natural language processing* (pp. 13112–13120). Singapore: Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.810

Crothers, E., Japkowicz, N., Viktor, H., & Branco, P. (2022). Adversarial robustness of neural-statistical features in detection of generative transformers. In *2022 International joint conference on neural networks (IJCNN)* (pp. 1–8). ISSN: 2161–4407 https://doi.org/10.1109/IJCNN55064.2022.9892269

Crothers, E., Japkowicz, N., & Viktor, H. L., et al. (2023). Machine-generated text: A comprehensive survey of threat models and detection methods. *IEEE Access*, *11*, 70977–71002. https://doi.org/10.1109/ACCESS.2023.3294090

Damodaran, P. (2021). Parrot: Paraphrase generation for NLU. Version Number: v1.0.

De Angelis, L., Baglivo, F., Arzilli, G., Privitera, G. P., Ferragina, P., Tozzi, A. E., & Rizzo, C. (2023). ChatGPT and the rise of large language models: The new AI-driven infodemic threat in public health. *Frontiers in Public Health*, *11*, 1166120.

Dhaini, M., Poelman, W., & Erdogan, E. (2023). Detecting chatgpt: A survey of the state of detecting chatgpt-generated text. arXiv preprint arXiv:2309.07689.

Ding, S. H. H., Fung, B. C. M., Iqbal, F., & Cheung, W. K. (2017). Learning stylometric representations for authorship analysis. *IEEE Transactions on Cybernetics*, *49*(1), 107–121.

Eder, M. (2011). Style-markers in authorship attribution: A cross-language study of the authorial fingerprint. *Studies in Polish Linguistics*, (6), 101–116.

Eder, M., Kestemont, M., & Rybicki, J. (2016). Stylometry with R: A package for computational text analysis. *The R Journal*, *8*(1), 1–15. https://doi.org/10.32614/RJ-2016-007

Evert, S., Proisl, T., Jannidis, F., Reger, I., Pielström, S., Schöch, C., & Vitt, T. (2017). Understanding and explaining delta measures for authorship attribution. *Digital Scholarship in the Humanities*, *32*(suppl_2), ii4–ii16. https://doi.org/10.1093/llc/fqx023

Fagni, T., Falchi, F., Gambini, M., Martella, A., & Tesconi, M. (2021). TweepFake: About detecting deepfake tweets. *PloS ONE*, *16*(5), e0251415. Publisher: Public Library of Science. https://doi.org/10.1371/journal.pone.0251415

Geng, C., Huang, S.-j., & Chen, S. (2020). Recent advances in open set recognition: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *43*(10), 3614–3631. Publisher: IEEE.

GPTZero. (2025). Accessed: 2025-05-20 https://gptzero.me.

Guo, B., Zhang, X., Wang, Z., Jiang, M., Nie, J., Ding, Y., Yue, J., & Wu, Y. (2023). How Close is ChatGPT to human experts? Comparison Corpus, Evaluation, and Detection. *CoRR*, *abs/2301.07597*. arXiv preprint arXiv:2301.07597.

Guo, L., Yang, W., Ma, L., & Ruan, J. (2024a). BLGAV: Generative AI author verification model based on BERT and bilstm. In G. Faggioli, N. Ferro, P. Galuščáková, & A. G. S. Herrera (Eds.), *Working Notes Papers of the CLEF 2024 Evaluation Labs* (pp. 2585–2592). CEUR-WS.org. http://ceur-ws.org/Vol-3740/paper-237.pdf.

Guo, M., Han, Z., Chen, H., & Peng, J. (2024b). A machine-generated text detection model based on text multi-feature fusion. In G. Faggioli, N. Ferro, P. Galuscáková, & A. G. S. d. Herrera (Eds.), *Working notes of the conference and labs of the evaluation forum (CLEF 2024), Grenoble, France, 9–12 September, 2024* (pp. 2593–2602). CEUR-WS.org (*vol. 3740*). CEUR Workshop Proceedings. https://ceur-ws.org/Vol-3740/paper-238.pdf.

Hicke, R., & Mimno, D. (2023). T5 meets Tybalt: Author attribution in early modern english drama using large language models. arXiv preprint arXiv:2310.18454.

HIX, A. I. (2025). HIX AI Detector. Accessed: 2025-05-20 https://bypass.hix.ai/ai-detector.

Hu, X., Ou, W., Acharya, S., Ding, S.,& D'Gama, R., (2023). TDRLM: Stylometric learning for authorship verification by topic-debiasing. *Expert Systems with Applications*, *233*,120745.

Huang, B., Chen, C., & Shu, K. (2024a). Can large language models identify authorship? arXiv preprint arXiv:2403.08213.

Huang, X., Ruan, W., Huang, W., Jin, G., Dong, Y., Wu, C., Bensalem, S., Mu, R., Qi, Y., Zhao, X. et al. (2024b). A survey of safety and trustworthiness of large language models through the lens of verification and validation. *Artificial Intelligence Review*, *57*(7), 175.

Hung, C.-Y., Hu, Z., Hu, Y., & Lee, R. K.-W. (2023). Who wrote it and why? Prompting large-language models for authorship verification. arXiv preprint arXiv:2310.08123.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems*, *30*, 3146–3154.

Krishna, K., Song, Y., Karpinska, M., Wieting, J., & Iyyer, M. (2023). Paraphrasing evades detectors of AI-generated text, but retrieval is an effective defense. arXiv preprint arXiv:2303.13408.

Kumarage, T., Garland, J., Bhattacharjee, A., Trapeznikov, K., Ruston, S. W., & Liu, H., et al. (2023). Stylometric Detection of AI-Generated Text in Twitter Timelines. *CoRR*, *abs/2303.03697*. arXiv preprint arXiv:2303.03697.

Kumarage, T., & Liu, H. (2023). Neural authorship attribution: Stylometric analysis on large language models. In *2023 International conference on cyber-enabled distributed computing and knowledge discovery (cyberc)* (pp. 51–54). IEEE.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2019). BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461.

Lhoest, Q., Villanova del Moral, A., Jernite, Y., Thakur, A., von Platen, P., Patil, S., Chaumond, J., Drame, M., Plu, J., Tunstall, L., Davison, J., Šaško, M., Chhablani, G., Malik, B., Brandeis, S., Le Scao, T., Sanh, V., Xu, C., Patry, N., McMillan-Major, A., Schmid, P., Gugger, S., Delangue, C., Matussière, T., Debut, L., Bekman, S., Cistac, P., Goehringer, T., Mustar, V., Lagunas, F., Rush, A., & Wolf, T. (2021). Datasets: A community library for natural language processing. In *Proceedings of the 2021 conference on empirical methods in natural language processing: System demonstrations* (pp. 175–184). Online and Punta Cana, Dominican Republic: Association for Computational Linguistics. https://aclanthology.org/2021.emnlp-demo.21.

Li, Y., Li, Q., Cui, L., Bi, W., Wang, Z., Wang, L., Yang, L., Shi, S., & Zhang, Y. (2024a). MAGE: Machine-generated text detection in the wild. In L.-W. Ku, A. Martins, & V. Srikumar (Eds.), *Proceedings of the 62nd annual meeting of the association for computational linguistics (Volume 1: Long Papers)* (pp. 36–53). Bangkok, Thailand: Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.acl-long.3

Li, Z., Shi, Y., Liu, Z., Yang, F., Payani, A., Liu, N., & Du, M. (2024b). Quantifying multilingual performance of large language models across languages. Version Number: 2 https://doi.org/10.48550/arXiv.2404.11553

Liu, A., Diab, M., & Fried, D. (2024a). Evaluating large language model biases in persona-steered generation. arXiv:2405.20253 [cs] https://doi.org/10.48550/arXiv.2405.20253

Liu, Y., Han, T., Ma, S., Zhang, J., Yang, Y., & Tian, J., (2023a). Summary of chatgpt/gpt-4 research and perspective towards the future of large language models. *Meta-radiology*, *1*(2), 100017.

Liu, Y., Zhang, Z., Zhang, W., Yue, S., Zhao, X., Cheng, X., Zhang, Y., & Hu, H. (2023b). ArguGPT: evaluating, understanding and identifying argumentative essays generated by GPT models. *CoRR*, *abs/2304.07666*. arXiv preprint arXiv:2304.07666.

Liu, Z., Yao, Z., Li, F., & Luo, B. (2024b). On the detectability of ChatGPT content: benchmarking, methodology, and evaluation through the lens of academic writing. arXiv:2306.05524 [cs] https://doi.org/10.48550/arXiv.2306.05524

Lorenz, L., Aygüler, F. Z., Schlatt, F., & Mirzakhmedova, N., et al. (2024). Baselineavengers at PAN 2024: Often-forgotten baselines for LLM-generated text detection. In G. Faggioli, N. Ferro, P. Galuscáková, & A. G. S. d. Herrera (Eds.), *Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), Grenoble, France, 9–12 September, 2024* (pp. 2761–2768). CEUR-WS.org (*vol. 3740*). CEUR Workshop Proceedings. https://ceur-ws.org/Vol-3740/paper-262.pdf.

Lu, N., Liu, S., He, R., Ong, Y.-S., Wang, Q., & Tang, K. (2024). Large language models can be guided to evade AI-generated text detection. *Transactions on Machine Learning Research*, *2024*. https://openreview.net/forum?id=lLE0mWzUrr.

Lund, B. D., Wang, T., Mannuru, N. R., Nie, B., Shimray, S., & Wang, Z. (2023). ChatGPT and a new academic reality: Artificial intelligence-written research papers and the ethics of the large language models in scholarly publishing. *Journal of the Association for Information Science and Technology*, *74*(5), 570–581.

Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., & Lee, S.-I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, *2*(1), 2522–5839.

Lüdeling, A., & Kytö, M. (Eds.) (2008). Corpus linguistics: An international handbook (vol. 1). Handbücher zur Sprach- und Kommunikationswissenschaft. Berlin New York: Walter de Gruyter.

Macko, D., Moro, R., Uchendu, A., Lucas, J., Yamashita, M., Pikuliak, M., Srba, I., Le, T., Lee, D., Simko, J., & Bielikova, M., et al. (2023). MULTITuDe: Large-scale multilingual machine-generated text detection benchmark. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Proceedings of the 2023 conference on empirical methods in natural language processing* (pp. 9960–9987). Singapore: Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.616

Mikros, G. K., Koursaris, A., & Bilianos, D., (2023). AI-writing detection using an ensemble of transformers and stylometric features. *IberLEF*, *3496*.

Miralles, P., Martín, A., & Camacho, D. (2024). Team aida at PAN: Ensembling normalized log probabilities. In G. Faggioli, N. Ferro, P. Galuščáková, & A. G. S. Herrera (Eds.), *Working notes papers of the CLEF 2024 evaluation labs* (pp. 2807–2813). CEUR-WS.org. http://ceur-ws.org/Vol-3740/paper-268.pdf.

Mosteller, F. (1968). Association and estimation in contingency tables. *Journal of the American Statistical Association*, *63*(321), 1–28.

Mukherjee, S., Mitra, A., Jawahar, G., Agarwal, S., Palangi, H., & Awadallah, A. (2023). Orca: Progressive learning from complex explanation traces of gpt-4. arXiv preprint arXiv:2306.02707.

Neal, T., Sundararajan, K., Fatima, A., Yan, Y., Xiang, Y., & Woodard, D. (2017). Surveying stylometry techniques and applications. *ACM Computing Surveys (CSuR)*, *50*(6), 1–36.

Ochab, J. K., Argasiński, J., Grabska-Gradzińska, I., & Przystalski, K. (2025a). Repository for: Stylometry recognizes human and LLM-generated texts in short samples. Publisher: OSF https://doi.org/10.17605/OSF.IO/DFZ6K

Ochab, J. K., Matias, M., Boba, T., & Walkowiak, T. (2025b). StylOch at PAN: Gradient-boosted trees with frequency-based stylometric features. In J. Carrillo-de Albornoz, J. Gonzalo, L. Plaza, A. G. S. de Herrera, J. Mothe, F. Piroi, P. Rosso, D. Spina, G. Faggioli, & N. Ferro (Eds.), *Experimental IR meets multilinguality, multimodality, and interaction. proceedings of the sixteenth international conference of the CLEF association (CLEF 2025)* Lecture Notes in Computer Science. Berlin Heidelberg New York: Springer.

Ochab, J. K., & Walkowiak, T. (2024). Implementing interpretable models in stylometric analysis. In *Digital humanities 2024: Conference abstracts*. Washington, D.C.: George Mason University (GMU).

Okulska, I., Stetsenko, D., Kołos, A., Karlińska, A., Głąbińska, K., & Nowakowski, A. (2023). Stylometrix: An open-source multilingual tool for representing stylometric vectors. arXiv preprint arXiv:2309.12810.

OpenAI (2025). OpenAI API reference documentation: Create chat completion. (last accessed on 2025-06-30) https://platform.openai.com/docs/api-reference/chat/create.

Patel, A., Rao, D., Kothary, A., McKeown, K., & Callison-Burch, C. (2023). Learning interpretable style embeddings via prompting llms. arXiv preprint arXiv:2305.12696.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Przystalski, K., Argasiński, J. K., Lipp, N., & Pacholczyk, D. (2025). Building personality-driven language models: How neurotic is ChatGPT. Synthesis lectures on engineering, science, and technology. CHAM: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-80087-0

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, *21*(140), 1–67.

Řehůřek, R., & Sojka, P. (2010). Software framework for topic modelling with large corpora. In *Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks* (pp. 45–50). Valletta, Malta: ELRA.

Sadasivan, V. S., Kumar, A., Balasubramanian, S., Wang, W., & Feizi, S., et al. (2025). Can AI-generated text be reliably detected? stress testing AI text detectors under various attacks. *Transactions on Machine Learning Research*. https://openreview.net/forum?id=OOgsAZdFOt.

Sarvazyan, A. M., González, J., Franco-Salvador, M., Rangel, F., Chulvi, B., & Rosso, P. (2023a). Overview of AuTexTification at IberLEF 2023: detection and attribution of machine-generated text in multiple domains. In *Procesamiento del Lenguaje Natural. Jaén, Spain.*

Sarvazyan, A. M., González, J., Rosso, P., & Franco-Salvador, M. (2023b). Supervised machine-generated text detectors: family and scale matters. In A. Arampatzis, E. Kanoulas, T. Tsikrika, S. Vrochidis, A. Giachanou, D. Li, M. Aliannejadi, M. Vlachos, G. Faggioli, & N. Ferro (Eds.), *Experimental IR Meets Multilinguality, Multimodality, and Interaction* (pp. 121–132). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-42448-9_11

Schuster, T., Schuster, R., Shah, D. J., & Barzilay, R., et al. (2020). The limitations of stylometry for detecting machine-generated fake news. *Computational Linguistics*, *46*(2), 499–510. https://doi.org/10.1162/coli_a_00380

Stamatatos, E. (2009). A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology*, *60*(3), 538–556.

Stiff, H., & Johansson, F. (2022). Detecting computer-generated disinformation. *International Journal of Data Science and Analytics*, *13*(4), 363–383. https://doi.org/10.1007/s41060-021-00299-5

Su, Z., Wu, X., Zhou, W., Ma, G., & Hu, S. (2024). HC3 Plus: A semantic-invariant human ChatGPT comparison CORPUS. arXiv: [cs] http://arxiv.org/abs/2309.02731.

Sun, J., & Lv, Z. (2025). Zero-shot detection of LLM-generated text via text reorder. *Neurocomputing*, *631*, 129829.

Touvron, H., Lavril, T., Izacard, G. et al. (2023). Llama: Open and efficient foundation language models.

Uchendu, A., Ma, Z., Le, T., Zhang, R., & Lee, D., et al. (2021). TURINGBENCH: A benchmark environment for turing test in the age of neural text generation. In M.-F. Moens, X. Huang, L. Specia, & S. W.-t. Yih (Eds.), *Findings of the association for computational linguistics: EMNLP 2021* (pp. 2001–2016). Punta Cana, Dominican Republic: Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.findings-emnlp.172

Vaze, S., Han, K., Vedaldi, A., & Zisserman, A. (2022). Open-set recognition: A good closed-set classifier is all you need. In *International conference on learning representations*. https://openreview.net/forum?id=5hLP5JY9S2d.

Wang, S., Cui, S., Zhang, C., Zhang, Z., Wang, J., & Liu, T. (2024a). Towards persona-oriented LLM-generated text detection: Benchmark dataset and method. In M. Wand, K. Malinovská, J. Schmidhuber, & I. V. Tetko (Eds.), *Artificial neural networks and machine learning – ICANN 2024* (pp. 352–367). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-72350-6_24

Wang, Y., Mansurov, J., Ivanov, P., Su, J., Shelmanov, A., Tsvigun, A., Whitehouse, C., Mohammed Afzal, O., Mahmoud, T., Sasaki, T., Arnold, T., Aji, A. F., Habash, N., Gurevych, I., & Nakov, P., et al. (2024b). M4: Multi-generator, multi-domain, and multi-lingual black-box machine-generated text detection. In Y. Graham & M. Purver (Eds.), *Proceedings of the 18th conference of the European chapter of the association for computational linguistics (Volume 1: long papers)* (pp. 1369–1407). St. Julian's, Malta: Association for Computational Linguistics. https://aclanthology.org/2024.eacl-long.83/.

Wu, J., Yang, S., Zhan, R., Yuan, Y., Chao, L. S., & Wong, D. F., et al. (2025). A survey on LLM-generated text detection: necessity, methods, and future directions. *Computational Linguistics*, (pp. 1–64). https://doi.org/10.1162/coli_a_00549

Xu, Z., Zhang, K., & Sheng, V. S. (2024). FreqMark: Frequency-based watermark for sentence-level detection of LLM-generated text. arXiv preprint arXiv:2410.10876.

Yadagiri, A., Kalita, D., Ranjan, A., Bostan, A. K., Toppo, P., & Pakray, P. (2024). Team CNLP-NITS-PP at PAN: Leveraging BERT for accurate authorship verification: A novel approach to textual attribution. In G. Faggioli, N. Ferro, P. Galuščáková, & A. G. S. Herrera (Eds.), *Working notes papers of the CLEF 2024 evaluation labs* (pp. 2976–2987). CEUR-WS.org. http://ceur-ws.org/Vol-3740/paper-290.pdf.

Yao, Y., Duan, J., Xu, K., Cai, Y., Sun, Z., & Zhang, Y. (2024). A survey on large language model (LLM) security and privacy: The good, the bad, and the ugly. *High-Confidence Computing*, (p. 100211).

Yu, P., Chen, J., Feng, X., & Xia, Z. (2025). CHEAT: A large-scale dataset for detecting CHatgpt-written abstracts. *IEEE Transactions on Big Data*, *11*(3), 1–9. https://doi.org/10.1109/TBDATA.2025.3536929

Zellers, R., Holtzman, A., Rashkin, H., Bisk, Y., Farhadi, A., Roesner, F., & Choi, Y. (2019). Defending against neural fake news. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d. Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems*. Curran Associates, Inc. (*vol. 32*). https://proceedings.neurips.cc/paper_files/paper/2019/file/3e9f0fc9b2f89e043bc6233994dfcf76-Paper.pdf.

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z. et al. (2023). A survey of large language models. arXiv preprint arXiv:2303.18223.