BEYOND VIA: ANALYSIS AND ESTIMATION OF THE IMPACT OF LARGE LANGUAGE MODELS IN ACADEMIC PAPERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Through an analysis of more than 3 million arXiv papers, we identify several previously overlooked shifts in word usage that should be driven by large language models (LLMs), such as the increased frequency of "**beyond**" and "**via**" in titles and the decreased frequency of "**the**" and "**of**" in abstracts. Because of the similarities among different LLMs, experiments show that current classifiers struggle to accurately determine which specific model generated a given text in multi-class classification tasks. By adopting a direct and highly interpretable linear approach and accounting for differences between models and prompts, we characterize lexical patterns potentially associated with LLM influence and show that real-world LLM usage is heterogeneous and dynamic. Meanwhile, the continuous evolution of LLMs also poses greater challenges for detection and estimation.[1]

1 INTRODUCTION

The increasing impact of large language models (LLMs) in academic publications has been observed (Liang et al., 2024; Geng & Trotta, 2024; Kobak et al., 2024). As LLMs continue to develop, has their impact evolved more recently?

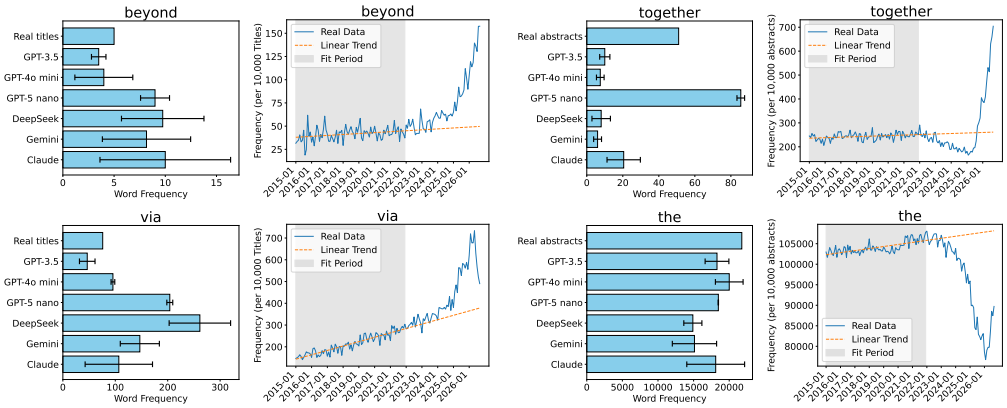
Different LLMs possess their own idiosyncrasies, which can be used to classify the text (Wu et al., 2023; Antoun et al., 2024; Sun et al., 2025). The outputs of various LLMs show some similarities (Jiang et al., 2025), which could reduce the accuracy of predictions. The use of words and expressions in academic writing is constantly evolving (Bizzoni et al., 2020; Wang et al., 2023; Cheng et al., 2024). In parallel, researchers have been monitoring specific indicative words to estimate the usage of LLMs in academic writing (Gray, 2025; He & Bu, 2025). Comparisons between texts generated by LLMs and those written by humans can now be carried out using different techniques (Wu et al., 2025).

We compare the similarities and differences between the texts produced by LLMs and those written by humans from different perspectives. We aim to analyze and estimate the impact of different LLMs on academic publications. For example, as depicted in Figure 1a, certain LLMs have a preference for the word "**beyond**" and "**via**" when generating titles from real arXiv abstracts, whose frequency is also becoming more common in real arXiv paper titles from 2025. Figure 1b shows that the frequency of "**together**" first declines markedly and is then followed by a rapid increase, likely reflecting the contrasting preferences between newer and older LLMs regarding this term. Even common words like "**the**" are likely to have changed noticeably due to the increasing use of LLMs. The dynamics described above are not isolated cases. How the evolution and heterogeneity of LLMs cause variations in real-world word usage patterns remains (relatively) underexplored. When these changes in individual words are considered together, they may also become statistically significant. Using LLM-based simulations, word frequency analysis can be used to further estimate relative stylistic alignment with simulated LLM outputs on academic publishing.

Our goal is not causal attribution of LLM usage, which may be fundamentally difficult in the absence of counterfactual writing trajectories or platform-level intervention data. The main findings of our paper are as follows:

[1]We will make our data publicly available for broader use.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) In arXiv paper titles.

(b) In arXiv paper abstracts.

Figure 1: **Left**: Word frequency comparison for generated titles or rewritten abstracts based on 2,000 real arXiv abstracts; error bars denote variance across models and prompts. **Right**: Temporal trends of word frequencies in real data, with the yellow dashed line fitted on data from 2015 to 2021 and extended to early 2026.

- Different LLMs share certain similarities as well as differences in their language styles, such as word usage. There is a growing tendency for newer models to become more human-like.
- The influence of LLMs in academic writing has become more widespread and varied, as reflected in both paper titles and the changing usage of common terms in abstracts.
- More sophisticated methods do not necessarily produce better estimates, while direct approaches can also offer new insights into the understanding of the societal impact of LLMs.

2 Data and Preprocessing

arXiv data We analyze abstracts of arXiv papers, based on a dataset that is updated weekly on Kaggle[2] and Hugging Face[3]. These data include more than 3.1 million arXiv papers across different fields. Considering the quantity and rapid update rate of arXiv papers, they may well be the best avenue for timely monitoring of LLM influence on academic writing.

PubMed Data This paper incorporates data from the PubMed database[4] to broaden our analysis. If we consider only papers published in or after 2010 that include both titles and abstracts, there are still more than 17 million papers, mainly in the fields of biomedicine and life sciences.

Stopwords Only frequent words are considered to estimate the impact of LLMs. We also used data from the Google Books Ngram dataset to establish a threshold for selecting common words. We are also interested in how stopwords are used in LLMs, with reference to the list in Natural Language Toolkit (NLTK) (Bird, 2006).

Preprocessing Our analysis centers on the publication timing of papers. In particular, for the arXiv dataset, we use only the initial versions of the titles and abstracts and do not update them over time. The details of the data preprocessing are provided in Section A.1 of the appendix.

[2] https://www.kaggle.com/datasets/Cornell-University/arxiv

[3] https://huggingface.co/datasets/librarian-bots/arxiv-metadata-snapshot

[4] https://pubmed.ncbi.nlm.nih.gov/

3 METHODS

3.1 TREND OF WORD FREQUENCY OVER TIME

We denote the observed frequency of word w in the dataset at time t by $f_w^{\mathrm{d}}(t)$, which is defined by the following equation:

$$f_w^{\mathrm{d}}(t) = f_w(t) + \varepsilon_w(t) \tag{1}$$

where $f_w(t)$ refers to the pattern of how word frequency changes over time, while $\varepsilon_w(t)$ represents random variations.

Word frequency is influenced by multiple confounding factors, including shifts in research topics and evolving writing conventions, independent of any LLM influence. We assume that the baseline word frequency changes linearly over time and then model the effect of LLM usage as a factor that modifies this established linear trend, as shown in the formula below:

$$f_w(t) = a_w + b_w t + c_w(t) \tag{2}$$

where a_w and b_w are word-specific coefficients, while $c_w(t)$ represents the effect introduced by LLMs.

Based on observational data from the pre-LLM period $t \in \mathcal{T}_1$, we estimate the parameters a_w and b_w and $\hat{a}_w$ and $\hat{b}_w$. These estimates are then used to predict the word frequency $f_w^{\mathrm{pred}}(t)$ over the later time interval $t \in \mathcal{T}_2$ as follows:

$$f_w^{\mathrm{pred}}(t) = \hat{a}_w + \hat{b}_w t\,. \tag{3}$$

As shown in Figure 1, the linear regression results based on data from 2021 and earlier can also reasonably predict the word frequencies for the first ten months of 2022, i.e., before the emergence of ChatGPT. Therefore, we use the above linear regression to predict how word frequencies would evolve under a counterfactual scenario without LLMs.

3.2 IMPACT ESTIMATION

For different models m, they have varying preferences for words, which also depend on the prompt. Let f_w^0 represent the word frequency in human-written text. For model m under prompt p, the word frequency is $f_w^{m,p}$, and its mean across prompts is denoted by f_w^m.

For the observed values in Equation 1 and the predicted values in Equation 3, we can easily obtain the following ratio:

$$r_w(t) = \frac{f_w^{\mathrm{d}}(t)}{f_w^{\mathrm{pred}}(t)}\,. \tag{4}$$

If we further assume that $\eta_0(t)$ represents the proportion in human-written text at time t, and $\eta_{m,p}(t)$ represents the latent stylistic contribution from model m and prompt p. Then, under the assumption that the observed text is a mixture of these components, we can obtain the following approximate relationship:

$$r_w(t) \approx \eta_0(t) + \sum_{m\in\mathcal{M}, p\in\mathcal{P}} \eta_{m,p}(t)\frac{f_w^m}{f_w^0}, \tag{5}$$

where $\mathcal{M}$ represents the set of all LLMs under consideration, and $\mathcal{P}$ represents the set of all prompts being considered.

If we denote all the unknowns related to η as the vector $\boldsymbol{\eta}(t)$, and define $\mathcal{M}^*$ and $\mathcal{P}^*$ to include both human and non-prompted cases to accommodate η_0. Then for the set of all considered words $\mathcal{W}$, $\boldsymbol{\eta}(t)$ can be determined using the following relation:

$$\begin{aligned} \min_{\boldsymbol{\eta}(t)} \quad & \sum_{w\in\mathcal{W}} \left(r_w(t) \quad - \sum_{m\in\mathcal{M}^*,\, p\in\mathcal{P}} \eta_{m,p}(t)\,\frac{f_w^{m,p}}{f_w^0} \right)^2 \\ \text{s.t.} \quad & \eta_0(t) + \sum_{m\in\mathcal{M}^*,\, p\in\mathcal{P}} \eta_{m,p}(t) = 1. \end{aligned} \tag{6}$$

Model	Date	Model	Date
GPT-3.5 (OpenAI, 2022)	22-11-30	Gemini 2.5 Flash (Google, 2025a)	25-04-17
GPT-4o mini (OpenAI, 2024)	24-07-18	Gemini 2.5 Pro (Google, 2025b)	25-03-25
GPT-5 nano (OpenAI, 2025)	25-08-07	Gemini 3 Flash (Google, 2025)	25-12-17
GPT-5.4 nano (OpenAI, 2026)	26-03-17	Gemini 3.5 Flash (Google, 2026)	26-05-19
DeepSeek V3 (DeepSeek-AI, 2024)	24-12-26	Claude 3 Haiku (Anthropic, 2024)	24-03-13
DeepSeek R1 (DeepSeek-AI, 2025a)	25-01-20	Claude Haiku 4.5 (Anthropic, 2025)	25-10-15
DeepSeek V3.2 (DeepSeek-AI, 2025b)	25-12-01		
DeepSeek V4 Flash (DeepSeek-AI, 2026)	26-04-24		

Table 1: Initial public release dates of the LLMs used in this paper.

It is important to note that the value of η_0 may be very close to 1, for example, before the emergence of ChatGPT. At the same time, other values of η may be close to 0. Therefore, although the loss function can be adjusted, it needs to be done with caution.

The optimal choice of term combinations can be determined in multiple ways, for instance, by identifying which word combinations yield the smallest error in linear regression, or the lowest estimated LLM impact before the emergence of ChatGPT.

3.3 Text Similarity Comparison

We explore two different tasks: generating the title and modifying the abstract. To examine the similarities and differences between human-authored texts and those generated by LLMs, as well as across outputs from different LLMs, we initially assess textual similarity using multiple metrics. We then attempt to train classifiers to categorize texts generated by different LLMs. For example, we use ROUGE (Lin, 2004) to compare the lexical similarity of texts BERTScore (Zhang et al., 2019) to assess their semantic similarity between two pieces of text.

Referring to the design and parameters of a previous study (Sun et al., 2025), we also train classifiers to distinguish texts generated by different LLMs, based on four models: BERT (Devlin et al., 2019), GPT-2 (Radford et al., 2019), T5 (Raffel et al., 2020), and LLM2Vec (BehnamGhader et al., 2024).

The selection of parameters, such as the learning rate is consistent with that of the previous paper (Sun et al., 2025). Although adjusting the model and its parameters might yield better classification results, the purpose of this section is to validate our dataset using the current best open-source classifier, so no modifications were made.

4 Simulation

4.1 Models

Simulations were conducted using different models from the GPT, DeepSeek, Gemini, and Claude. Their release dates range from November 2022 to April 2026, as illustrated in Table 1.[5] More details on the simulation using the API can be found in Appendix Section A.3. We are aware that some LLMs are not covered in this study, and some versions of GPT and Claude are not included in the experiments. But the models we chose are expected to yield fairly stable estimation results.

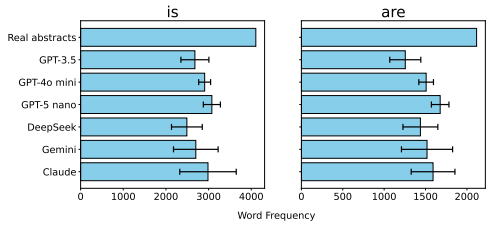
4.2 Prompts

Given the diversity of real-world application scenarios, we adopt two prompts of different lengths, one long and one short, for simulation.

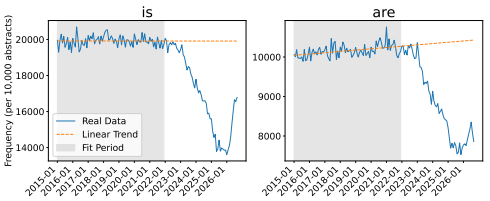
Short Prompt for Abstract Rewriting : Authors may restrict their use of LLM tools to light assistance, such as minor rewrites. Therefore, we consider the short prompt as follows.

```
Revise the following sentences. Please only output the revised text in
JSON format. Only one version. No explanations. Only plain text.
```

[5]The three latest models, DeepSeek V4, Gemini 3.5 and GPT-5.4 nano, are mainly used for the analysis in Figure 3.



(a) Word frequencies in 2,000 abstracts or in the corresponding LLM-processed content.



(b) Temporal trends of word frequencies in real abstracts.

Figure 2: Examples of common words. Some newer models, such as GPT-5.4 nano, show different word-frequency patterns, including an increase in the frequencies of "is" and "are", as shown in Figure 26.

```
EXAMPLE JSON OUTPUT: {"text": "Your revised version of the provided
sentences goes here."}
```

Other Prompts: We can also provide more details in a longer prompt to simulate a researcher maximizing LLM capabilities for editing, like requesting the persona of a professional academic editor. The long prompt for abstract rewriting can be found in Appendix A.2, along with the prompt used for title generation.

5 Results

5.1 Word Frequency Analysis

LLMs have their own preferences for word usage. Some commonly used words are often regarded as stopwords that carry little informational content, and are therefore removed in natural language processing (Sarica & Luo, 2021), but there are differences between LLMs and humans in the usage of many common words. For instance, for the 20 most common words in the abstracts, their word variation frequency defined in Equation 8 is illustrated in Figure 6 in the Appendix. The frequency changes of these common words are likely to be largely unrelated to shifts in research topics, which is also why we focus on them. To better present word preferences across LLMs, we plot the frequency of the 20 words most favored and the frequency of the 20 words least favored by LLMs, based on the coefficient of variation defined in Equation 11, in Figures 7a and 7b in the appendix.

Figure 2 shows a decline in the frequencies of "is" and "are", and demonstrates that most LLMs may indeed avoid using these two words. Figure 8 in the appendix also provides additional examples. Although it is difficult to estimate LLM usage based on individual words, some correlations can still be observed. Thus, in theory, the changes in the frequencies of these common words can also be used to estimate LLM impact.

LLMs are also continuously evolving, and different LLMs have distinct word preferences. The word "together" and "the" in Figure 1b serves as a clear example. Words previously considered characteristic of ChatGPT, like "**delve**" and "**intricate**", are abandoned by newer LLMs, as seen in Figure 9 in the Appendix. The frequency patterns of some common words also show changing trends. For instance, as illustrated in Figure 8 in the appendix, the frequency of "**and**" has recently exceeded the predicted upward trend, whereas the frequency of "**this**" first decreased and then increased. These shifts may reflect differences in word preference between some recent models such as GPT-5 nano and earlier models, as shown in Figure 6.

In addition, Figure 7a indicates that, except for GPT-5 Nano, other models show a strong preference for the word "**furthermore**". The frequency of this term rises sharply but returns to a level close to its previous growth trajectory recently, as illustrated in Figure 8 of the appendix. This pattern may reflect increasing usage of GPT-5 Nano or similar models in academic writing.

The changes in word frequency in abstracts discussed above are observed across the entire arXiv corpus. Similar patterns also emerge when CS and non-CS papers are examined separately. For example, Figures 10 and 11 in the appendix shows the frequency changes of "the", "of", "delve", and "intricate" in these two groups. This phenomena is observed not only in the titles and abstracts

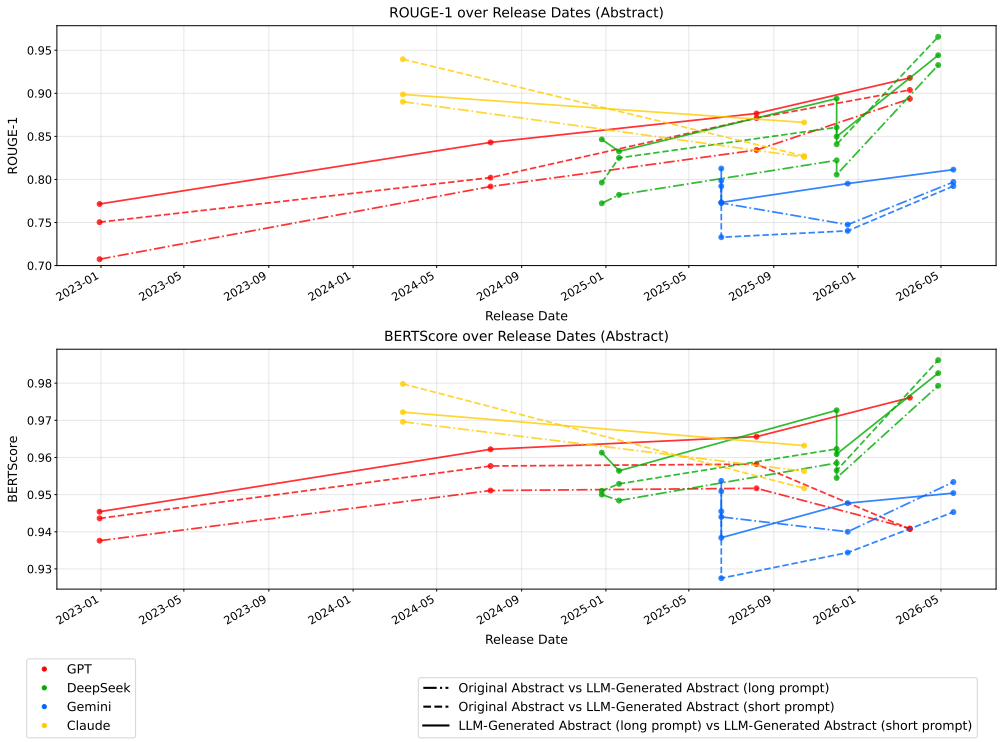


Figure 3: Similarity between the processed texts of the two prompts and their similarity to the real abstracts. The x-axis represents the model release dates.

of papers from arXiv, but also in those from PubMed, as illustrated in Figures 12, 13, and 14 of the appendix. This suggests that the findings of this paper have cross-domain generality and alleviates the concern that the shifts are due to changes in research topics.

The changes and evolution of LLMs also require us to estimate and monitor their impact with a dynamic perspective. The differences among various LLMs offer valuable insights for estimating the usage and influence of different models.

5.2 Text Similarity Analysis

Apart from the differences in word usage frequency, we aim to examine it from a broader perspective of text similarity (e.g., ROUGE-1 and BERTScore). Figures 3 and 15a indicate that the abstracts processed by the two prompts are close to each other, whereas they show a larger difference from the original ones. In general, the results from the two prompts of the new LLMs are gradually aligning more closely with the original abstracts under ROUGE metrics. These results also suggest that, given the same input, outputs from different series of LLMs may tend to become more homogeneous. Semantically, this trend is not as evident.

Similar conclusions can be drawn from the title generation in Figure 15b in the appendix: while the new LLMs are closer to the true titles according to ROUGE, this is not reflected in the semantic results from BERTScore.

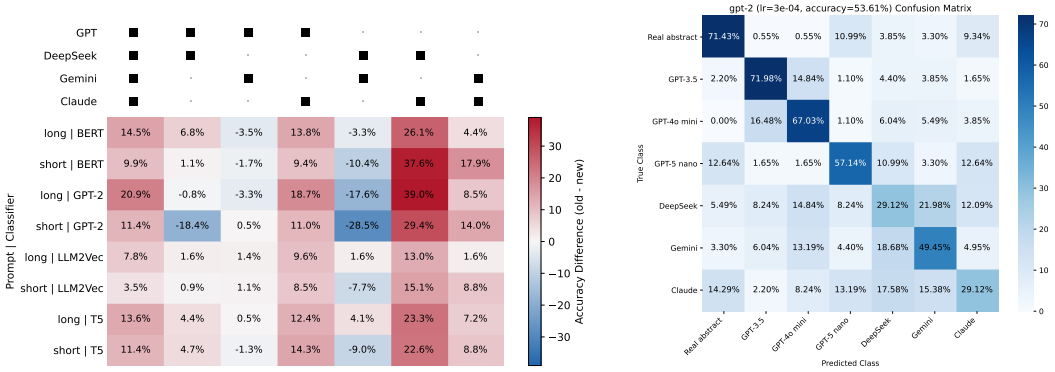
The analysis in this part further highlights the differences between various LLMs, as well as the changes occurring within models of the same series. Higher similarity does not necessarily mean stronger model performance, which may instead indicate that the model has made fewer modifications to the output text.

5.3 Classification

We consider two scenarios in classification tasks: distinguishing between texts generated by different LLMs, and a multi-class classification that includes texts written by humans. The models we use

Table 2: Classification results. Prompt: long, Classifier: BERT, Model Type: old.

GPT	■	■	■	■			
DeepSeek	■	■			■	■	
Gemini	■		■		■		■
Claude	■			■		■	■
Accuracy	74.9%	89.8%	88.5%	**95.1**%	78.8%	91.8%	90.4%



(a) Heatmap of accuracy difference (old - new) across LLM combinations and classifiers.

(b) Confusion matrix for 7-class GPT-2 classification (lr=3e-04)

Figure 4: Classification results. In the training and test data of the right Figure, GPT-3.5, GPT-4o mini, and GPT-5-nano are mixtures of two prompts, while DeepSeek, Gemini, and Claude are mixtures of multiple versions and two prompts. After mixing, each class contains 2,000 abstracts.

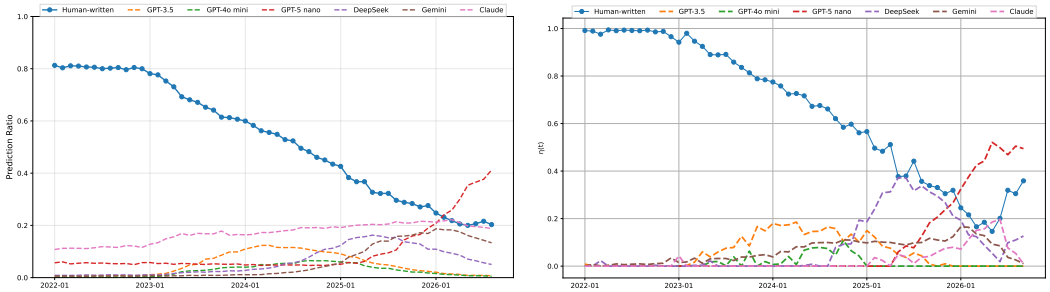
here are based on the existing open-source work of Sun et al. (2025), which is one of the few studies classifying texts from different LLMs. Our main purpose is to test whether the existing classifiers are effective, rather than to achieve the highest accuracy by tuning parameters and structures.

In the first scenario, we consider binary and four-class classification. An earlier version (GPT-3.5, DeepSeek V3, Gemini 2.5 Flash, Claude 3 Haiku) and the most recent version (GPT-5 nano, DeepSeek V3.2, Gemini 3 Flash, Claude Haiku 4.5) from each of the 4 model families (GPT, DeepSeek, Gemini, and Claude) are chosen for comparison.

Table 2 presents the classification results of the Bert-based classifier for texts generated with long prompt, where the values in the table represent the results with the highest accuracy in the test set with different parameters. If the text comes from only two different LLMs, the classifier works remarkably well. But when faced with four categories, its prediction accuracy declines sharply. Table 4 to Table 17 in the Appendix provide more detailed results. Figure 4a shows the gap in classification accuracy of the same classifier on texts generated by new and old LLMs, given a fixed prompt. In most cases, the classifier performs better on texts generated by older models. This may also suggest a trend toward homogenization among texts generated by different LLMs.

For the second scenario, we directly perform a seven-class or thirteen-class classification, including texts written by humans. The results of the seven categories shown in Figures 4b, 16, 17, and 18 of the appendix refer to three GPT models, along with DeepSeek, Gemini, and Claude (with mixed outputs from different models in each series), as well as human-written texts. In Figures 19 to 22 of the appendix, the thirteen categories correspond to the classification of each model separately.Texts generated by different LLMs may have different probabilities of being correctly classified. Accuracy decreases as the number of categories increases. While the accuracy of classifying human-written texts and GPT-processed texts is quite high, it's also fairly common for human-written text to be mistakenly identified as LLM-generated.

In practice, the diversity of prompts and models far exceeds the controlled conditions of our experiment, and these classifiers could easily mistake human-written text for content generated by LLMs.



(a) LLM2Vec-based classifier. (b) Word-frequency based method.

Figure 5: Estimation of LLM impact in arXiv abstracts. The word selection with the smallest LLM impact prediction mean squared error (MSE) in the grid search during the first half of 2022: 435 words used for estimation are drawn from the 15,000 most frequent words in the Google Books Ngram dataset that occur at least 100 times in the 2,000 preprocessed abstracts used for simulation.

5.4 Impact Estimation

We first use the previously trained classifier to process real abstract data. Figure 5a shows that, before ChatGPT appeared, about 20% of the abstracts were still identified as being generated by LLMs, which is consistent with the previous experimental results. Similar results can also be observed for the other classifiers, as shown in Appendix Figure 23.

Therefore, we use the word frequency-based estimation, which is also easier to interpret. Based on Equation 6, the estimated impact of LLMs varies depending on the different words and parameters. For example, we can set

$$\eta_{m,p}(t) \in [0,1], \quad \forall m \in \mathcal{M}^*, \, p \in \mathcal{P}^* \tag{7}$$

To validate our word frequency-based estimation approach, we used SLSQP (Sequential Least Squares Programming) to solve this equation, with the initial value of $\eta_{m,p}(t)$ uniformly distributed.

We select 2021 as the cutoff point for fitting, since ChatGPT appeared in late 2022, and the 2022 data can also serve to verify the accuracy of our method. The word selection in Figure 5b corresponds to the smallest mean square error of the estimates during the first 6 months in 2022. Appendix B.2 provides the parameters used for the grid search. We can also observe in Figure 5b that the proportion of text similar to the GPT-3.5 style first increases and then decreases.

Figure 24 in the appendix also presents results estimated using different combinations of words. For instance, the results in Figure 24a are derived from the stopwords in NLTK that appear at least once in the 2,000 arXiv abstracts. The estimates based on stopwords show larger fluctuations, likely due to the low differentiation of these words across different models, even though they differ from human-written text. Even with numerical fluctuations, the general trend and conclusions from Figure 24 are still consistent, such as the LLM impact estimates being close to zero before October 2022. Hence, estimating the influence of LLMs based on word frequency can reflect the usage of different LLMs.

6 Related Work

LLM-Generated Text Detection. Researchers have proposed many detectors for identifying text generated by LLMs, with a variety of approaches and techniques (Wu et al., 2025). However, in real-world scenarios, such as academic writing, human modifications to LLM outputs can indeed affect the effectiveness of detection (Russell et al., 2025). Considering the variety of models and use cases, it may be an impossible task to accurately estimate the numerical usage of LLMs (Geng & Poibeau, 2025).

LLM Fingerprint. Some LLMs each have their own linguistic fingerprints, such as in vocabulary and lexical diversity, the frequency of certain lexical and morphosyntactic features (Reviriego et al., 2024; McGovern et al., 2025). These features can be used to trace their origins (Nikolic et al., 2025) as well as to detect LLM-generated text (Antoun et al., 2024; Sun et al., 2025) and check API

calls (Gao et al., 2024). Similar studies have also appeared in other languages, although some LLMs may be quite similar and difficult to distinguish (Zaitsu et al., 2025).

LLMs for Scientific Writing. Many people are discussing the use of LLMs in scientific research, and this has become an unstoppable trend (Eger et al., 2025). The disclosure of AI usage may impact how people perceive the article (Hazra et al., 2025). Despite the widespread use of AI in academic writing, there are few papers that disclose its usage (He & Bu, 2025). The scope of LLMs' use in scientific research goes beyond just writing, such as research fields (Hao et al., 2026), citations (Algaba et al., 2025), and more.

Impact Estimation The influence of LLMs on academic publications is diverse (Kusumegi et al., 2025). In addition to the previously mentioned works, some papers attempt to quantify these impacts, such as through the use of words (Liang et al., 2025) and different linguistic shifts (Bao et al., 2025; Lin et al., 2025).

7 Discussion

The focus of this paper is on the impact of LLMs rather than on whether the text was generated by them. Classifiers are likely to perform well in binary classification, but their accuracy significantly decreases in multi-class tasks. In real-world scenarios, however, the latter situation is more common.

Although the method of analyzing word frequency may sound simple, this intuitive approach proves to be quite effective in analyzing the impact of LLMs. In the absence of counterfactual data, linear extrapolation is a reasonable approximation. Focusing on more common words may provide better estimates, and our simple method can fill the gap left by complex classifiers. Our estimates can also be improved, for example, by considering more models and prompts.

In the era of LLMs, the way people use words is continually evolving (Geng & Trotta, 2025; Mak & Walasek, 2025). Real-world usage increasingly involves heterogeneous mixtures of models, prompts, and human revision. The effect of LLMs on text goes **beyond** academic publications. As similar impacts are likely to increase in the future, the methods and findings presented in this paper could serve as important references for subsequent research. The language style of LLMs, including their preference for certain words, requires further exploration (Juzek & Ward, 2025).

The opportunities these tools provide differ for authors from various regions (e.g., Eastern vs. Western) (Khan et al., 2025), though our paper does not explore this aspect. The modification of content and information is more troubling than any change in wording (Mohamed et al., 2025; Abdulhai et al., 2026). The generation of content may require more caution, such as hallucinated references (Sakai et al., 2026).

8 Conclusion

We systematically analyze similarities and differences between arXiv abstracts and titles processed by different LLMs. Compared to human-written texts, LLM-generated outputs exhibit distinct stylistic characteristics, with additional variation observed across different models. The data from PubMed also support these changes.

While complex classifiers can be employed for detection, these black-box methods raise concerns in real-world scenarios and also struggle in our experimental settings. This further suggests that we should be cautious when interpreting individual-level detection results.

Our approach provides a more direct and intuitive illustration of the impact of LLMs on academic writing. The results show that the observed lexical shifts are consistent with increasing editing activity mediated by LLMs. In some cases, simple aggregate lexical statistics may be more reliable than brittle black-box detectors.

Considering both similarity comparisons and detector outcomes, model outputs may indeed become progressively more human-like, raising more challenges for AI-generated text detection. Approaches for tracking and evaluating their impact will need to be continuously refined.

LLM USAGE

LLMs were used to assist with writing and coding, but the core ideas and content of the paper come from the authors, and the outputs produced by LLMs were verified by the authors.

REFERENCES

Marwa Abdulhai, Isadora White, Yanming Wan, Ibrahim Qureshi, Joel Leibo, Max Kleiman-Weiner, and Natasha Jaques. How llms distort our written language. *arXiv preprint arXiv:2603.18161*, 2026.

Andres Algaba, Vincent Holst, Floriano Tori, Melika Mobini, Brecht Verbeken, Sylvia Wenmackers, and Vincent Ginis. How deep do large language models internalize scientific literature and citation practices? *arXiv preprint arXiv:2504.02767*, 2025.

Anthropic. Claude 3 Haiku. `https://www.anthropic.com/news/claude-3-haiku`, 2024. Accessed: 2026-01-25.

Anthropic. Claude Haiku 4.5. `https://www.anthropic.com/news/claude-haiku-4-5`, 2025. Accessed: 2026-01-25.

Wissam Antoun, Benoît Sagot, and Djamé Seddah. From text to source: Results in detecting large language model-generated content. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 7531–7543, 2024.

Tong Bao, Yi Zhao, Jin Mao, and Chengzhi Zhang. Examining linguistic shifts in academic writing before and after the launch of chatgpt: a study on preprint papers. *Scientometrics*, pp. 1–31, 2025.

Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. Llm2vec: Large language models are secretly powerful text encoders. *arXiv preprint arXiv:2404.05961*, 2024.

Steven Bird. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 interactive presentation sessions*, pp. 69–72, 2006.

Yuri Bizzoni, Stefania Degaetano-Ortlieb, Peter Fankhauser, and Elke Teich. Linguistic variation and change in 250 years of english scientific writing: A data-driven approach. *Frontiers in Artificial Intelligence*, 3:73, 2020.

Myra Cheng, Kristina Gligorić, Tiziano Piccardi, and Dan Jurafsky. Anthroscore: A computational linguistic measure of anthropomorphism. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 807–825, 2024.

DeepSeek-AI. DeepSeek V3 release news. `https://api-docs.deepseek.com/news/news1226`, 2024. Accessed: 2026-01-20.

DeepSeek-AI. DeepSeek R1 release news. `https://api-docs.deepseek.com/news/news250120`, 2025a. Accessed: 2026-01-20.

DeepSeek-AI. DeepSeek V3.2 release news. `https://api-docs.deepseek.com/news/news251201`, 2025b. Accessed: 2026-01-20.

DeepSeek-AI. DeepSeek V4 release news. `https://api-docs.deepseek.com/news/news260424`, 2026. Accessed: 2026-05-25.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Steffen Eger, Yong Cao, Jennifer D'Souza, Andreas Geiger, Christian Greisinger, Stephanie Gross, Yufang Hou, Brigitte Krenn, Anne Lauscher, Yizhi Li, et al. Transforming science with large language models: A survey on ai-assisted scientific discovery, experimentation, content generation, and evaluation. *arXiv preprint arXiv:2502.05151*, 2025.

Irena Gao, Percy Liang, and Carlos Guestrin. Model equality testing: Which model is this api serving? *arXiv preprint arXiv:2410.20247*, 2024.

Mingmeng Geng and Thierry Poibeau. On the detectability of llm-generated text: What exactly is llm-generated text? *arXiv preprint arXiv:2510.20810*, 2025.

Mingmeng Geng and Roberto Trotta. Is chatgpt transforming academics' writing style? *arXiv preprint arXiv:2404.08627*, 2024.

Mingmeng Geng and Roberto Trotta. Human-llm coevolution: Evidence from academic writing. *arXiv preprint arXiv:2502.09606*, 2025.

Google. Developers can now start building with Gemini 2.5 Flash. `https://blog.google/products-and-platforms/products/gemini/gemini-2-5-flash-preview/`, April 2025a. Initial preview release. Accessed: 2026-09-26.

Google. Gemini 2.5: Our most intelligent ai model. `https://blog.google/innovation-and-ai/models-and-research/google-deepmind/gemini-model-thinking-updates-march-2025/`, March 2025b. Initial release of Gemini 2.5 Pro Experimental. Accessed: 2026-09-26.

Google. Gemini 3 Flash. `https://blog.google/products-and-platforms/products/gemini/gemini-3-flash/`, 2025. Accessed: 2026-01-20.

Google. Gemini 3.5. `https://blog.google/innovation-and-ai/models-and-research/gemini-models/gemini-3-5`, 2026. Accessed: 2026-05-25.

Andrew Gray. Estimating the prevalence of llm-assisted text in scholarly writing. *arXiv preprint arXiv:2512.01560*, 2025.

Qianyue Hao, Fengli Xu, Yong Li, and James Evans. Artificial intelligence tools expand scientists' impact but contract science's focus. *Nature*, pp. 1–7, 2026.

Sanchaita Hazra, Doeun Lee, Bodhisattwa Prasad Majumder, and Sachin Kumar. Accepted with minor revisions: Value of ai-assisted scientific writing. *arXiv preprint arXiv:2511.12529*, 2025.

Yongyuan He and Yi Bu. Academic journals' ai policies fail to curb the surge in ai-assisted academic writing. *arXiv preprint arXiv:2512.06705*, 2025.

Liwei Jiang, Yuanjun Chai, Margaret Li, Mickel Liu, Raymond Fok, Nouha Dziri, Yulia Tsvetkov, Maarten Sap, Alon Albalak, and Yejin Choi. Artificial hivemind: The open-ended homogeneity of language models (and beyond). *arXiv preprint arXiv:2510.22954*, 2025.

Tom S Juzek and Zina B Ward. Why does chatgpt "delve" so much? exploring the sources of lexical overrepresentation in large language models. In *Proceedings of the 31st international conference on computational linguistics*, pp. 6397–6411, 2025.

Farhan Kamrul Khan, Hazem Ibrahim, Nouar Aldahoul, Talal Rahwan, and Yasir Zaki. Who gets seen in the age of ai? adoption patterns of large language models in scholarly writing and citation outcomes. *arXiv preprint arXiv:2509.08306*, 2025.

Dmitry Kobak, Rita González-Márquez, Emőke-Ágnes Horvát, and Jan Lause. Delving into chatgpt usage in academic writing through excess vocabulary. *arXiv preprint arXiv:2406.07016*, 2024.

Keigo Kusumegi, Xinyu Yang, Paul Ginsparg, Mathijs de Vaan, Toby Stuart, and Yian Yin. Scientific production in the era of large language models. *Science*, 390(6779):1240–1243, 2025.

Weixin Liang, Zachary Izzo, Yaohui Zhang, Haley Lepp, Hancheng Cao, Xuandong Zhao, Lingjiao Chen, Haotian Ye, Sheng Liu, Zhi Huang, et al. Monitoring ai-modified content at scale: A case study on the impact of chatgpt on ai conference peer reviews. *arXiv preprint arXiv:2403.07183*, 2024.

Weixin Liang, Yaohui Zhang, Zhengxuan Wu, Haley Lepp, Wenlong Ji, Xuandong Zhao, Hancheng Cao, Sheng Liu, Siyu He, Zhi Huang, et al. Quantifying large language model usage in scientific papers. *Nature Human Behaviour*, pp. 1–11, 2025.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

Dingkang Lin, Naixuan Zhao, Dan Tian, and Jiang Li. Chatgpt as linguistic equalizer? quantifying llm-driven lexical shifts in academic writing. *arXiv preprint arXiv:2504.12317*, 2025.

Matthew HC Mak and Lukasz Walasek. Style, sentiment, and quality of undergraduate writing in the ai era: A cross-sectional and longitudinal analysis of 4,820 authentic empirical reports. *Computers and Education: Artificial Intelligence*, pp. 100507, 2025.

Hope Elizabeth McGovern, Rickard Stureborg, Yoshi Suhara, and Dimitris Alikaniotis. Your large language models are leaving fingerprints. In *Proceedings of the 1stWorkshop on GenAI Content Detection (GenAIDetect)*, pp. 85–95, 2025.

Amr Mohamed, Mingmeng Geng, Michalis Vazirgiannis, and Guokan Shang. Llm as a broken telephone: Iterative generation distorts information. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7493–7509, 2025.

Ivica Nikolic, Teodora Baluta, and Prateek Saxena. Model provenance testing for large language models. *arXiv preprint arXiv:2502.00706*, 2025.

OpenAI. Introducing ChatGPT. `https://openai.com/index/chatgpt/`, 2022. Accessed: 2026-01-20.

OpenAI. GPT-4o mini: Advancing cost-efficient intelligence. `https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/`, 2024. Accessed: 2026-01-20.

OpenAI. Introducing GPT-5. `https://openai.com/index/introducing-gpt-5/`, 2025. Accessed: 2026-01-20.

OpenAI. Introducing GPT-5.4 mini and nano. `https://openai.com/index/introducing-gpt-5-4-mini-and-nano/`, 2026. Accessed: 2026-09-25.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

Pedro Reviriego, Javier Conde, Elena Merino-Gómez, Gonzalo Martínez, and José Alberto Hernández. Playing with words: Comparing the vocabulary and lexical diversity of chatgpt and humans. *Machine Learning with Applications*, 18:100602, 2024.

Jenna Russell, Marzena Karpinska, and Mohit Iyyer. People who frequently use chatgpt for writing tasks are accurate and robust detectors of ai-generated text. *arXiv preprint arXiv:2501.15654*, 2025.

Yusuke Sakai, Hidetaka Kamigaito, and Taro Watanabe. Hallucitation matters: Revealing the impact of hallucinated references with 300 hallucinated papers in acl conferences. *arXiv preprint arXiv:2601.18724*, 2026.

Serhad Sarica and Jianxi Luo. Stopwords in technical language processing. *Plos one*, 16(8):e0254937, 2021.

Mingjie Sun, Yida Yin, Zhiqiu Xu, J Zico Kolter, and Zhuang Liu. Idiosyncrasies in large language models. *arXiv preprint arXiv:2502.12150*, 2025.

Gui Wang, Hui Wang, Xinyi Sun, Nan Wang, and Li Wang. Linguistic complexity in scientific writing: A large-scale diachronic study from 1821 to 1920. *Scientometrics*, 128(1):441–460, 2023.

Junchao Wu, Shu Yang, Runzhe Zhan, Yulin Yuan, Lidia Sam Chao, and Derek Fai Wong. A survey on llm-generated text detection: Necessity, methods, and future directions. *Computational Linguistics*, 51(1):275–338, 2025.

Kangxi Wu, Liang Pang, Huawei Shen, Xueqi Cheng, and Tat-Seng Chua. Llmdet: A third party large language models generated text detection tool. In *Findings of the association for computational linguistics: emnlp 2023*, pp. 2113–2133, 2023.

Wataru Zaitsu, Mingzhe Jin, Shunichi Ishihara, Satoru Tsuge, and Mitsuyuki Inaba. Stylometry can reveal artificial intelligence authorship, but humans struggle: A comparison of human and seven large language models in japanese. *PLoS One*, 20(10):e0335369, 2025.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019.

A Simulation Details

A.1 Data Preparation

As our study requires analyzing how effects vary over time, we incorporate timestamps into the analysis. The dataset provides two temporal indicators: the paper ID `id`, which indicates the initial submission period, and the last update date `update_date`, which indicates the most recent revision. We initially collected versions 1 (released in August 2020) through 283 (released in April 2026) from Kaggle, retaining only newly added papers in each version. Following arXiv's separation from Cornell University and the discontinuation of the Cornell-managed Kaggle repository, we continued data collection using the Hugging Face repository. A rigorous comparison confirmed that version 283 on Kaggle and version 209 on Hugging Face contained identical data. We therefore extended our collection with 20 additional snapshots from Hugging Face, covering versions 210 (released in May 2026) through 229 (released in September 2026), again retaining only newly added papers in each version. This approach preserves the titles and abstracts as they first appear in the collected snapshots and reduces the temporal mismatch introduced by subsequent revisions. For papers newly submitted during the weekly collection period, the delay between submission and the first recorded snapshot is at most one week. We categorize papers based on arXiv categories, using only the first category for papers assigned to multiple categories. For the LLM simulations, we randomly selected 2,000 papers with update dates between January and October 2022 from Kaggle version 265, predating the release of ChatGPT.

For the PubMed dataset, we utilized the annual baseline data up to the year 2025, which was released on January 29, 2026[6]. In this dataset, the temporal information of the publications is determined utilizing the `pubdate` column. To avoid excessively large data volumes and maintain temporal relevance, we restricted our collection strictly to papers published between 2010 and 2025. Furthermore, we cleaned the dataset by removing any records where either the title (`title`) or the abstract (`abstract`) was missing. After applying these filtering criteria, we retained a total of approximately 17.41 million papers. Unlike the discipline-based categorization used for the arXiv dataset, we substitute the subject classification in the PubMed dataset with the journal name, relying on the `journal` column. Our analysis primarily focuses on the top 100 journals in terms of publication volume. These selected high-volume journals are highly representative, accounting for 21.78% of the total articles published across the more than 20,000 journals present in the dataset.

For textual preprocessing across both corpora, we apply `re.findall(r'\b\w+\b', x.lower())` in Python to strip punctuation and convert all text to lowercase.

A.2 Other prompts

Long prompt for abstract rewriting:

```
Role:
Act as a professional academic editor and reviewer for top-tier scientific
journals (e.g., Nature, IEEE Transactions, ACM). You have expertise in
technical writing, grammar, and scientific logic.
Task:
Refine and polish the specific text provided below. Your goal is to
improve clarity, coherence, and academic tone while strictly maintaining
the original technical meaning.
Constraints & Guidelines:
1. Grammar & Syntax: Correct all grammatical, spelling, and punctuation
errors.
2. Academic Tone: Use formal, objective, and precise language. Avoid
colloquialisms, contractions, or overly flowery language.
3. Clarity & Flow: Improve sentence structure to enhance readability.
Break down overly complex sentences if necessary, but ensure the logical
flow remains smooth.
4. Vocabulary: Replace weak or repetitive words with more precise academic
vocabulary suited for a technical context.
```

[6] https://ftp.ncbi.nlm.nih.gov/pubmed/baseline/

```
5. Preservation:
  - Do NOT change the core scientific meaning or data.
  - Do NOT alter specific variable names, LaTeX formulas, citations (e.g.,
[1]), or technical terminology unless they are clearly incorrect.
6. Conciseness: Remove redundancy and fluff.
Output Format:
Please only output the polished text in JSON format. Only one version. No
explanations. Only plain text.
EXAMPLE JSON OUTPUT:
{"text": "Your polished version of the provided text goes here."}
```

Long prompt for title generation:

```
Read the following abstract of a research paper. Generate a representative
title that accurately reflects the main contribution of the work.

Constraint: You must output ONLY a valid JSON object. Do not include any
introductory or concluding text.

Format: {"title": "Your generated title here"}
```

Short prompt for title generation:

```
Generate a title based on the following abstract submitted to arXiv.

Constraint: You must output ONLY a valid JSON object. Do not include any
introductory or concluding text.

Format: {"title": "Your generated title here"}
```

A.3 API USAGE

All polished texts utilized in this study were generated via cloud-based LLM services rather than local execution. To ensure data consistency and mitigate potential disruptions caused by network instability, all LLM responses were acquired using Python-based batch API processing instead of interactive chat interfaces. The APIs were configured to output data in JSON format to strictly exclude extraneous explanatory text. All other settings were retained at their default values.

The specific model APIs employed are listed below:

OpenAI Models: `gpt-3.5-turbo`, `gpt-4o-mini`, `gpt-5-nano`, `gpt-5-4-nano`.
Access period: January 13, 2026 – September 26, 2026.

DeepSeek Models: DeepSeek-V3.2 (chat and reasoner modes) and DeepSeek-V4-Flash.
Note: For the DeepSeek-V3.2 experiments, `deepseek-chat` was used for the non-thinking mode and `deepseek-reasoner` for the thinking mode. DeepSeek-V4-Flash was evaluated in a separate, later experiment using `deepseek-v4-flash`.
Access period: January 13, 2026 – May 31, 2026.

Alibaba Cloud Models: `deepseek-v3`, `deepseek-r1`.
Note: These models utilized open-source code provided by DeepSeek and were hosted on servers operated by the Alibaba Cloud internet vendor.
Access period: January 13, 2026 – May 31, 2026.

Google Models: `gemini-2.5-flash`, `gemini-2.5-pro`, `gemini-3-flash-preview`, `gemini-3-5-flash`.
Access period: January 13, 2026 – May 31, 2026.

Anthropic Models: `claude-3-haiku-20240307`, `claude-haiku-4-5-20251001`.
Access period: January 21, 2026.

A.4 Computing Resources

Our experiments are conducted on the AutoDL cloud platform using a Docker container environment. Detailed hardware and software specifications are summarized in Table 3.

Table 3: System and Software Configurations.

System	Details	Software	Versions
GPU	NVIDIA RTX 5090	CUDA	12.8
CPU	Intel Xeon 8470Q	Python	3.12
Memory	90 GB	PyTorch	2.8.0
OS	Ubuntu 22.04.5	Transformers	4.44.2
OS Kernel	5.15.0-78	Accelerate	1.13.0
GPU Driver	595.58.03	PEFT	0.19.1

B Results Analysis

B.1 Word Frequency Comparison

Word frequency can also be analyzed from various perspectives. For example, we can define frequency change ratio as follows:

$$r_{w,m} = \frac{f_w^m - f_w^0}{f_w^0}\,. \tag{8}$$

The variance of the frequency of a word w in the same texts given a prompt p across different models m is expressed as:

$$\mathrm{Var}_{w,p} = \frac{1}{|\mathcal{M}^*| - 1} \sum_{m \in \mathcal{M}^*} (f_w^{m,p} - \bar{f}_m^p)^2 \tag{9}$$

where the average value $\bar{f}_m^p$ can be represented as

$$\bar{f}_m^p = \frac{1}{|\mathcal{M}^*|} \sum_{m \in \mathcal{M}^*} f_w^{m,p}\,. \tag{10}$$

Subsequently, the coefficient of variation is defined as

$$\mathrm{CV}_{w,p} = \frac{\sqrt{\mathrm{Var}_{w,p}}}{\bar{f}_m^p}\,. \tag{11}$$

Therefore, we can identify which words behave differently across different LLMs.

B.2 Parameters

To evaluate the sequence classification performance across different model architectures, we conduct fine-tuning experiments utilizing four text embedding backbones: BERT, GPT-2, T5, and LLM2Vec. For reproducibility and strict variance control, all experiments are executed on a single NVIDIA GPU using a fixed random seed of 42. The classification dataset comprises balanced partitions, specifically consisting of 1,818 training samples and 182 testing samples per label extracted from the generated response paths.

During the training phase, all models are optimized for 3 epochs with a mini-batch size of 8 using the AdamW optimizer. To mitigate potential gradient explosion issues and facilitate stable optimization at the onset of training, we enforce a strict gradient clipping threshold of 0.3 and apply a linear warmup schedule over the first 10total training steps (warmup ratio = 0.1). A weight decay coefficient of 0.001 is uniformly imposed across all configurations as a regularization bottleneck.

To account for the distinct architectural properties and parameter scales of the evaluated backbones, the base learning rate is adaptively tuned for each model. Specifically, we employ relatively higher learning rates for the smaller pre-trained language models (BERT, T5, and GPT-2) to facilitate efficient convergence, while applying a more conservative learning rate for the larger LLM2Vec

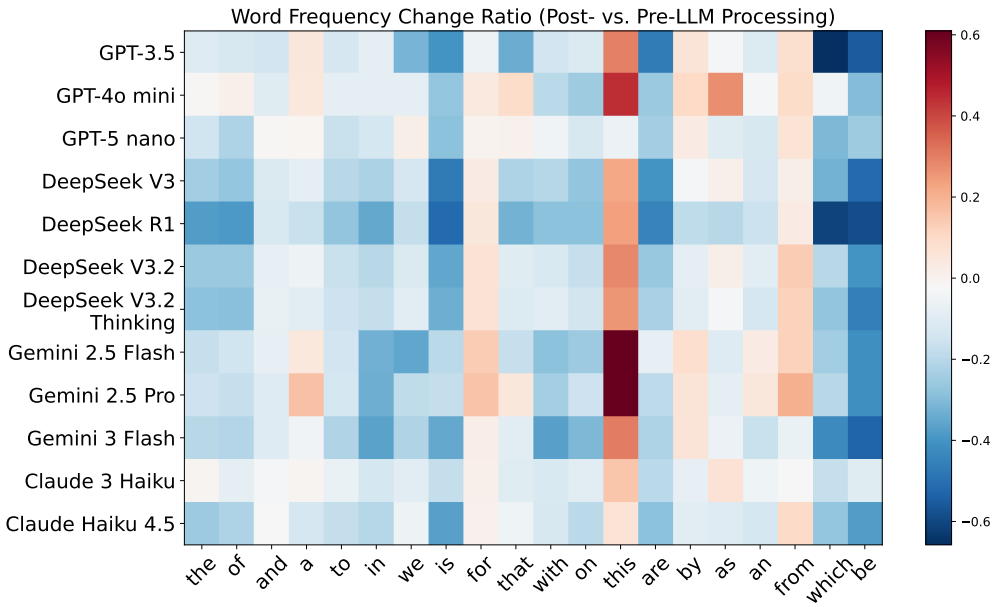


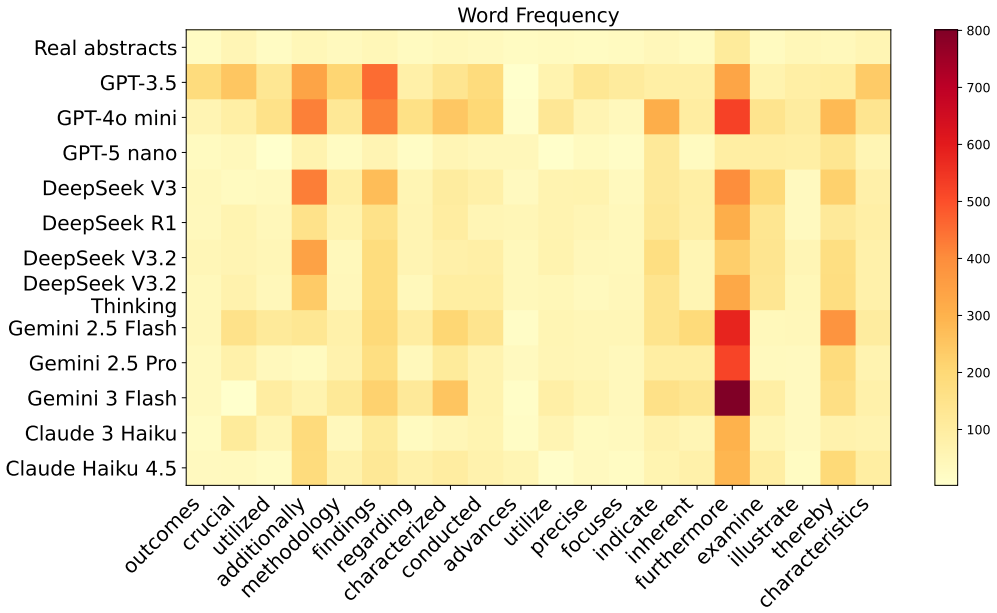
Figure 6: Frequency change ratios of the 20 most frequent words in 2,000 abstracts using multiple LLMs and the shorter prompt.

model to maintain training stability and prevent catastrophic forgetting. All other optimization and regularization parameters remain strictly identical across all models to ensure a fair and controlled evaluation.

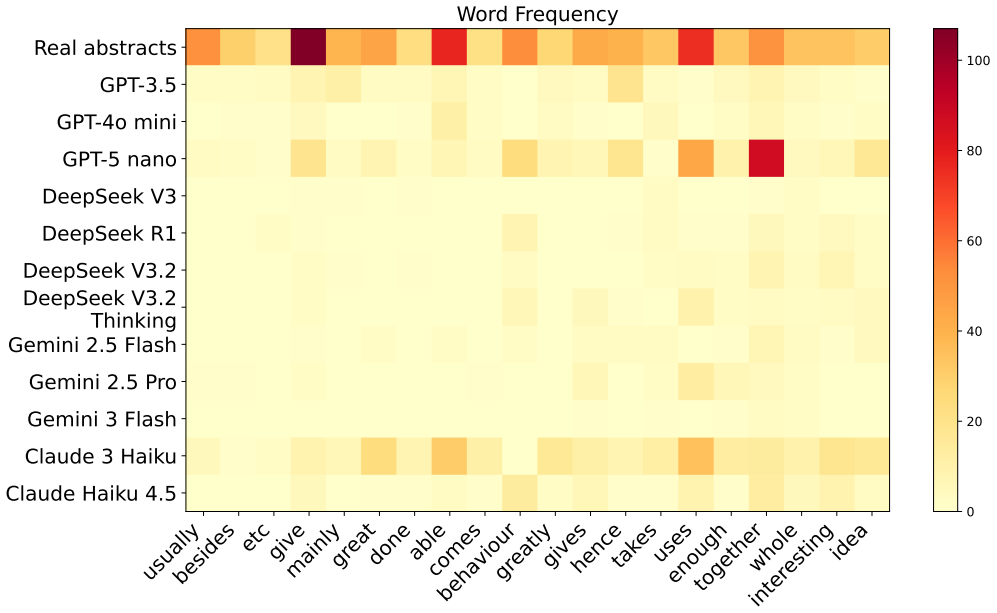
The parameters used in the grid search are as follows:

- Minimum frequency of the words in 2,000 abstracts: 3, 5, 7, 10, 15, 20, 30, 50, 75, 100, 150, 200
- The lowest word frequency ranking in the Google Books Ngram dataset: 500, 1000, 1500, 2000, 3000, 4000, 5000, 7000, 10000, 15000, 20000

B.3 Other Results



(a) The 20 words with the highest change coefficients that LLMs prefer to use.



(b) The 20 words with the highest change coefficients that humans prefer to use.

Figure 7: The frequencies of some words that appeared at least 20 times in the 2,000 abstracts used for the simulation.

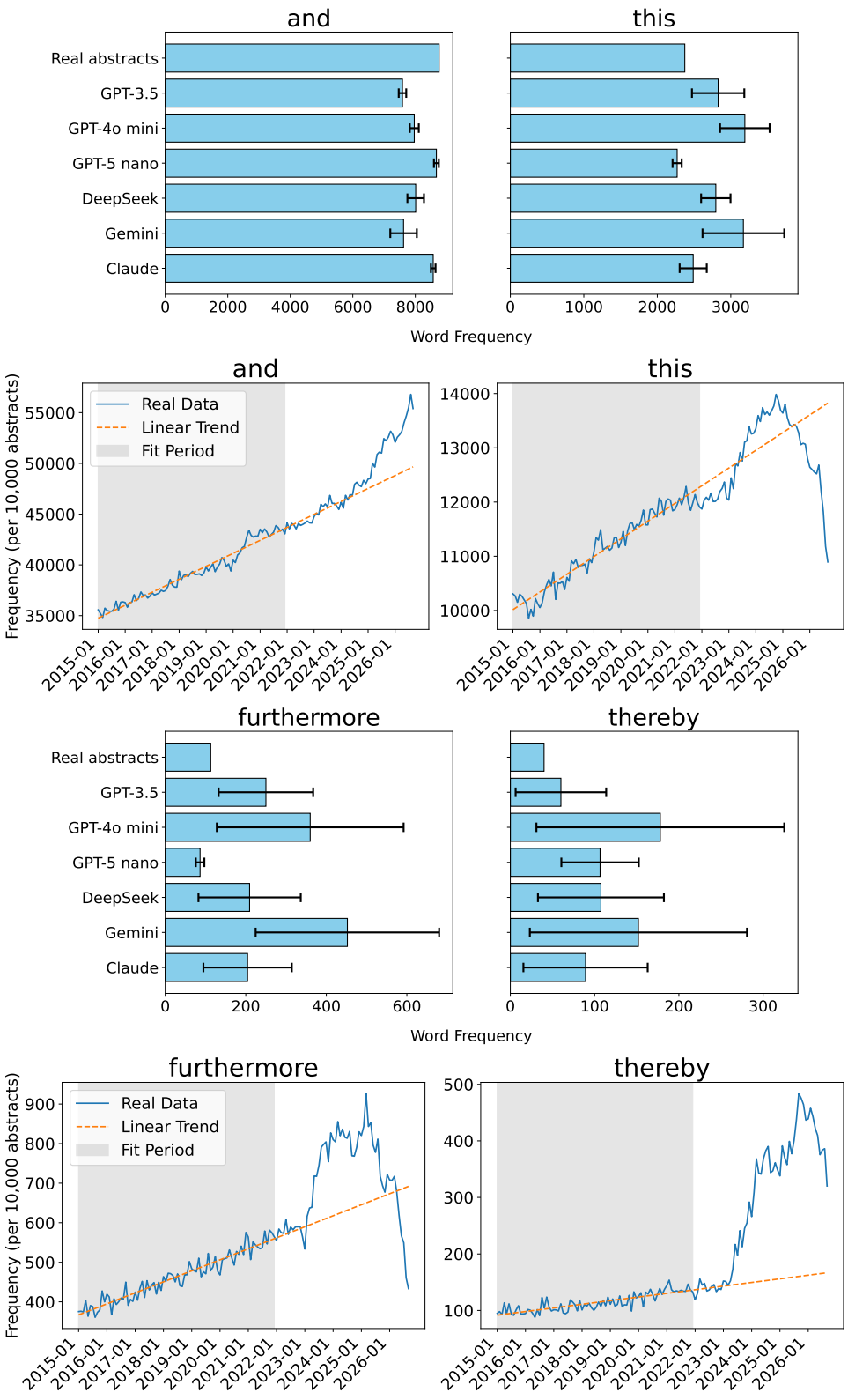


Figure 8: Supplementary comparison results for four common words: *and*, *this*, *furthermore*, and *thereby*.

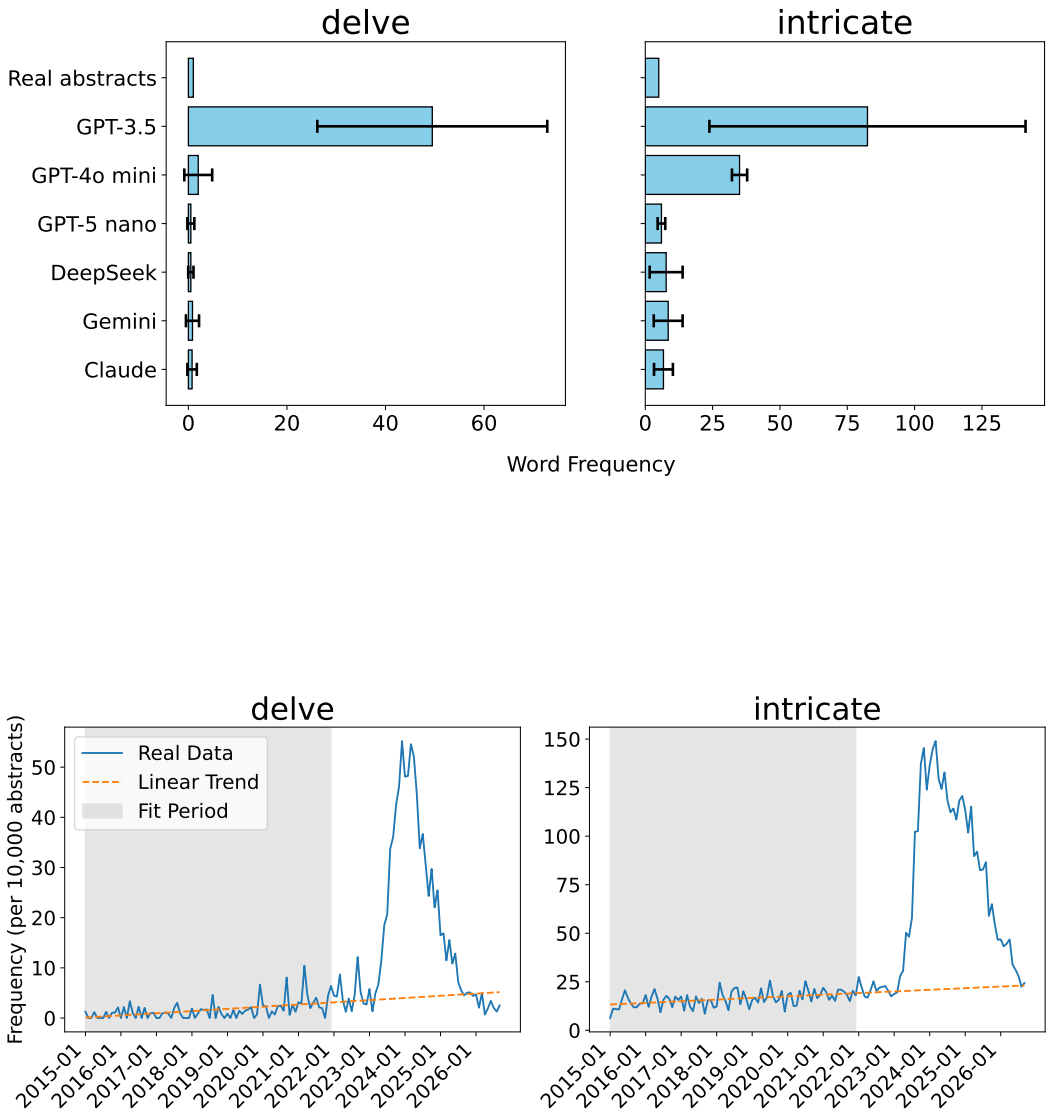


Figure 9: Supplementary comparison results for two words favored by GPT-3.5: *delve* and *intricate*.

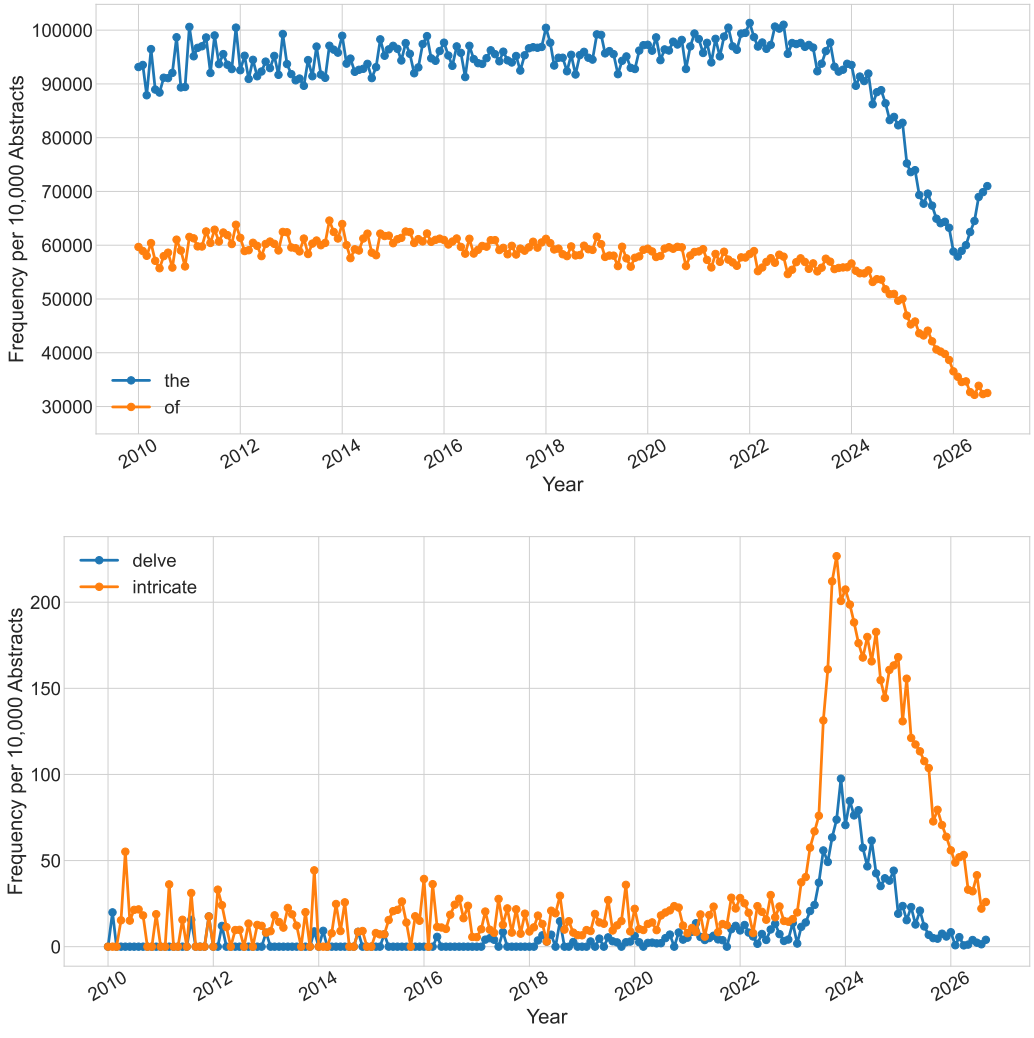


Figure 10: Supplementary comparison results: word frequency in CS abstracts.

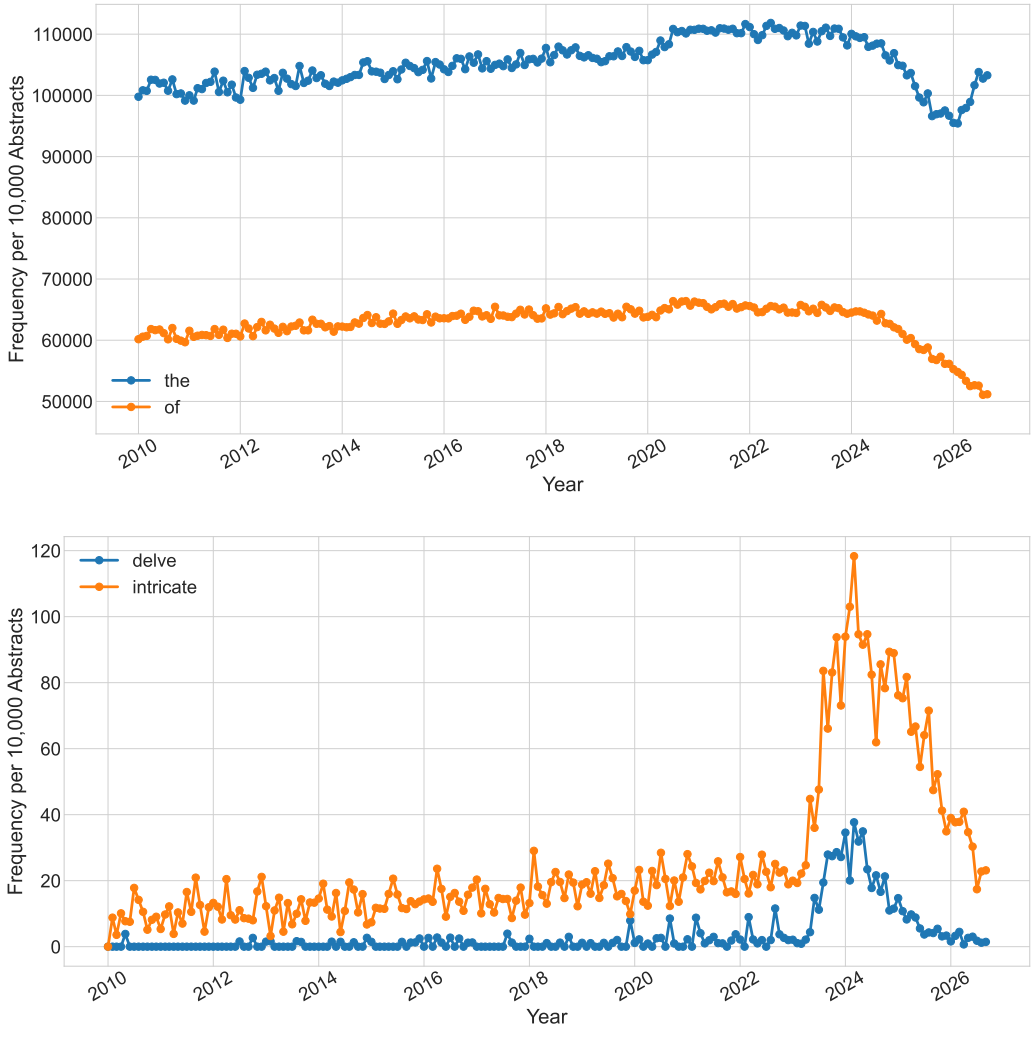


Figure 11: Supplementary comparison results: word frequency in non-CS abstracts.

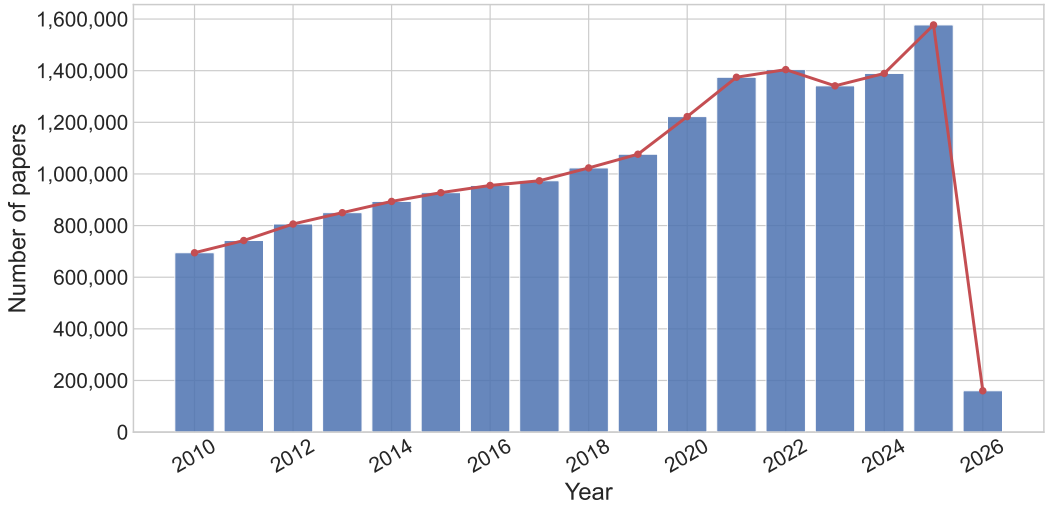


Figure 12: The number of papers in PubMed.

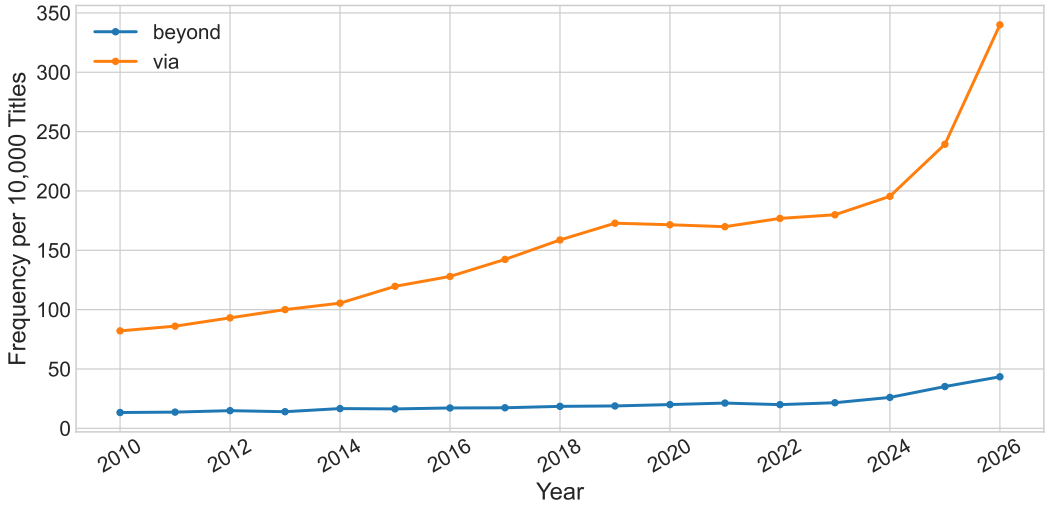


Figure 13: Word frequency in the titles of papers in PubMed.

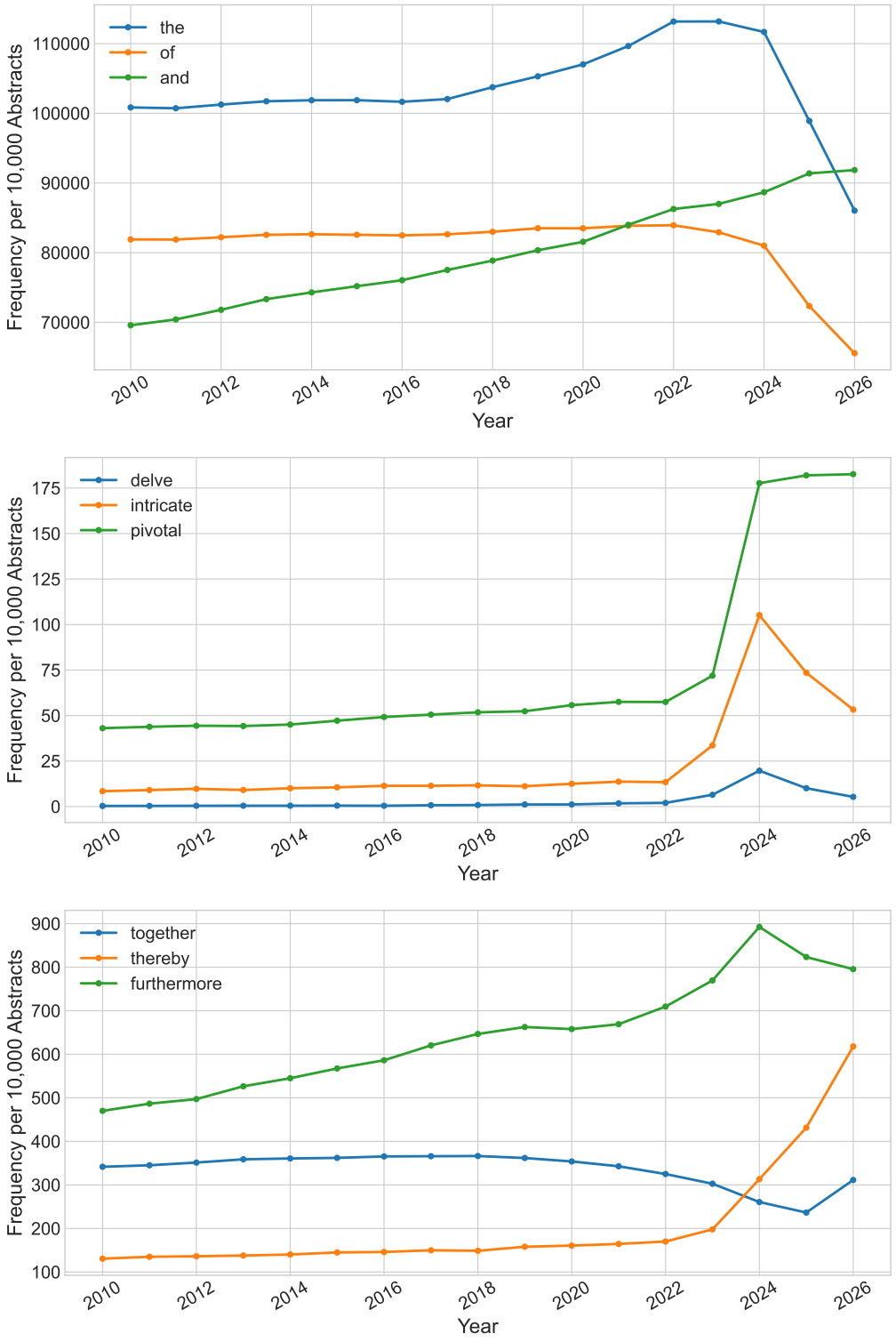


Figure 14: Word frequency in the abstracts of papers in PubMed.

Table 4: Classification results. Prompt: long, Classifier: LLM2Vec, Model Type: new.

GPT	■	■	■	■	·	·	·
DeepSeek	■	■	·	·	■	■	·
Gemini	■	·	■	·	■	·	■
Claude	■	·	·	■	·	■	■
Accuracy	76.8%	91.8%	**94.8**%	86.0%	86.0%	81.0%	90.4%

Table 5: Classification results. Prompt: long, Classifier: LLM2Vec, Model Type: old.

GPT	■	■	■	■	·	·	·
DeepSeek	■	■	·	·	■	■	·
Gemini	■	·	■	·	■	·	■
Claude	■	·	·	■	·	■	■
Accuracy	84.6%	93.4%	**96.2**%	95.6%	87.6%	94.0%	92.0%

Table 6: Classification results. Prompt: long, Classifier: BERT, Model Type: new.

GPT	■	■	■	■	·	·	·
DeepSeek	■	■	·	·	■	■	·
Gemini	■	·	■	·	■	·	■
Claude	■	·	·	■	·	■	■
Accuracy	60.4%	83.0%	**92.0**%	81.3%	82.1%	65.7%	86.0%

Table 7: Classification results. Prompt: long, Classifier: GPT-2, Model Type: new.

GPT	■	■	■	■	·	·	·
DeepSeek	■	■	·	·	■	■	·
Gemini	■	·	■	·	■	·	■
Claude	■	·	·	■	·	■	■
Accuracy	54.1%	77.2%	**89.3**%	75.0%	79.7%	50.3%	80.2%

Table 8: Classification results. Prompt: long, Classifier: GPT-2, Model Type: old.

GPT	■	■	■	■	·	·	·
DeepSeek	■	■	·	·	■	■	·
Gemini	■	·	■	·	■	·	■
Claude	■	·	·	■	·	■	■
Accuracy	75.0%	76.4%	86.0%	**93.7**%	62.1%	89.3%	88.7%

Table 9: Classification results. Prompt: long, Classifier: T5, Model Type: new.

GPT	■	■	■	■	·	·	·
DeepSeek	■	■	·	·	■	■	·
Gemini	■	·	■	·	■	·	■
Claude	■	·	·	■	·	■	■
Accuracy	60.6%	86.5%	**90.4**%	82.4%	70.9%	67.6%	81.3%

Table 10: Classification results. Prompt: long, Classifier: T5, Model Type: old.

GPT	■	■	■	■	·	·	·
DeepSeek	■	■	·	·	■	■	·
Gemini	■	·	■	·	■	·	■
Claude	■	·	·	■	·	■	■
Accuracy	74.2%	90.9%	90.9%	**94.8**%	75.0%	90.9%	88.5%

Table 11: Classification results. Prompt: short, Classifier: LLM2Vec, Model Type: new.

GPT	■	■	■	■	·	·	·
DeepSeek	■	■	·	·	■	■	·
Gemini	■	·	■	·	■	·	■
Claude	■	·	·	■	·	■	■
Accuracy	72.5%	84.3%	**93.4**%	85.7%	86.0%	74.2%	86.3%

Table 12: Classification results. Prompt: short, Classifier: LLM2Vec, Model Type: old.

GPT	■	■	■	■	·	·	·
DeepSeek	■	■	·	·	■	■	·
Gemini	■	·	■	·	■	·	■
Claude	■	·	·	■	·	■	■
Accuracy	76.0%	85.2%	94.5%	94.2%	78.3%	89.3%	**95.1**%

Table 13: Classification results. Prompt: short, Classifier: BERT, Model Type: new.

GPT	■	■	■	■	·	·	·
DeepSeek	■	■	·	·	■	■	·
Gemini	■	·	■	·	■	·	■
Claude	■	·	·	■	·	■	■
Accuracy	60.6%	79.7%	**90.7**%	82.4%	82.1%	50.0%	76.1%

Table 14: Classification results. Prompt: short, Classifier: BERT, Model Type: old.

GPT	■	■	■	■	·	·	·
DeepSeek	■	■	·	·	■	■	·
Gemini	■	·	■	·	■	·	■
Claude	■	·	·	■	·	■	■
Accuracy	70.5%	80.8%	89.0%	91.8%	71.7%	87.6%	**94.0**%

Table 15: Classification results. Prompt: short, Classifier: GPT-2, Model Type: new.

GPT	■	■	■	■	·	·	·
DeepSeek	■	■	·	·	■	■	·
Gemini	■	·	■	·	■	·	■
Claude	■	·	·	■	·	■	■
Accuracy	51.5%	71.4%	**84.1**%	77.5%	77.7%	53.8%	73.9%

Table 16: Classification results. Prompt: short, Classifier: GPT-2, Model Type: old.

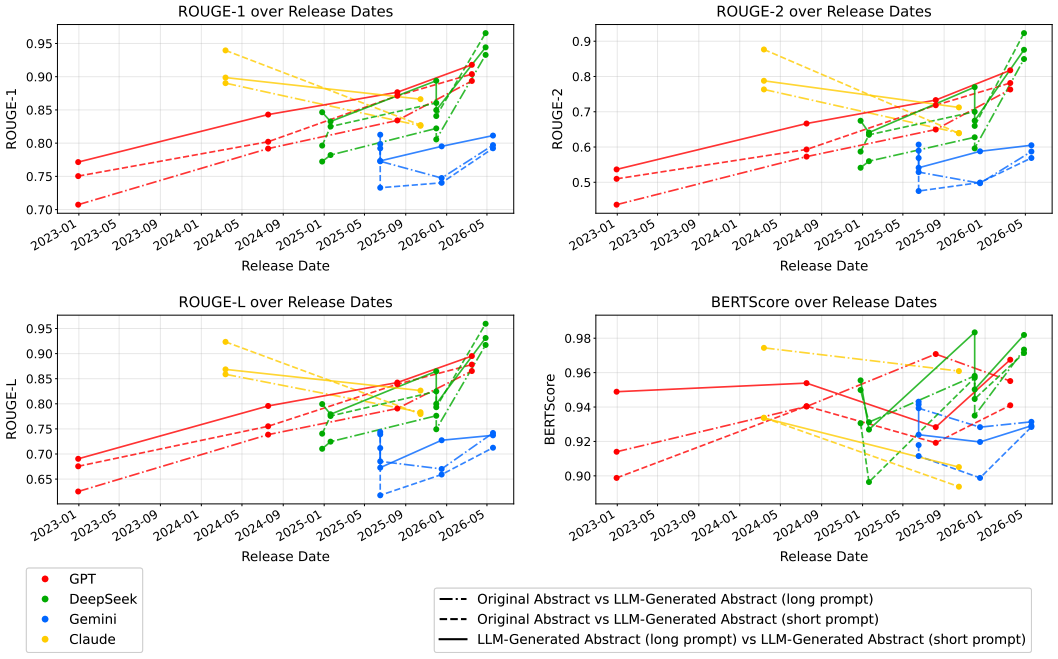
GPT	■	■	■	■			
DeepSeek	■	■			■	■	
Gemini	■		■		■		■
Claude	■			■		■	■
Accuracy	62.9%	53.0%	84.6%	**88.5**%	49.2%	83.2%	87.9%

Table 17: Classification results. Prompt: short, Classifier: T5, Model Type: new.

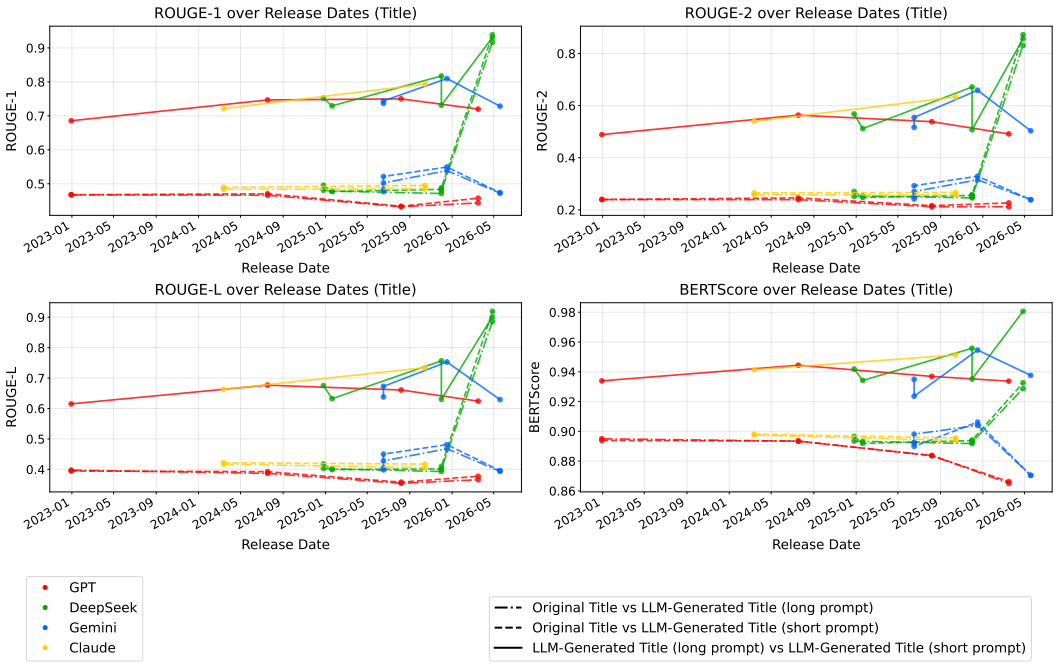
GPT	■	■	■	■			
DeepSeek	■	■			■	■	
Gemini	■		■		■		■
Claude	■			■		■	■
Accuracy	56.6%	76.6%	**87.6**%	76.4%	76.6%	64.8%	82.4%

Table 18: Classification results. Prompt: short, Classifier: T5, Model Type: old.

GPT	■	■	■	■			
DeepSeek	■	■			■	■	
Gemini	■		■		■		■
Claude	■			■		■	■
Accuracy	68.0%	81.3%	86.3%	90.7%	67.6%	87.4%	**91.2**%



(a) Similarity between the processed texts of the two prompts and their similarity to the real abstracts.



(b) Similarity between the titles generated by LLMs and the real titles.

Figure 15: Comparison Results of Text Similarity. The x-axis represents the model release dates.

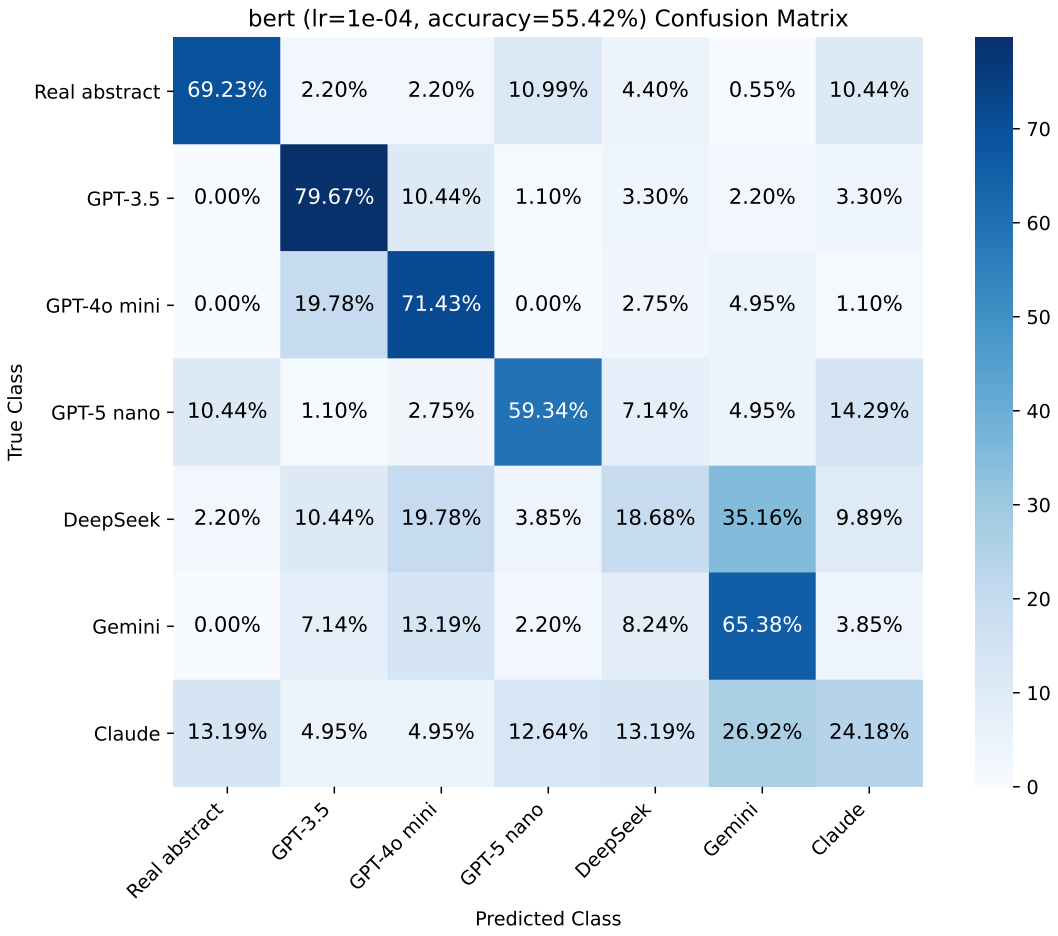


Figure 16: Confusion matrix for 7-class BERT classification (lr=1e-04)

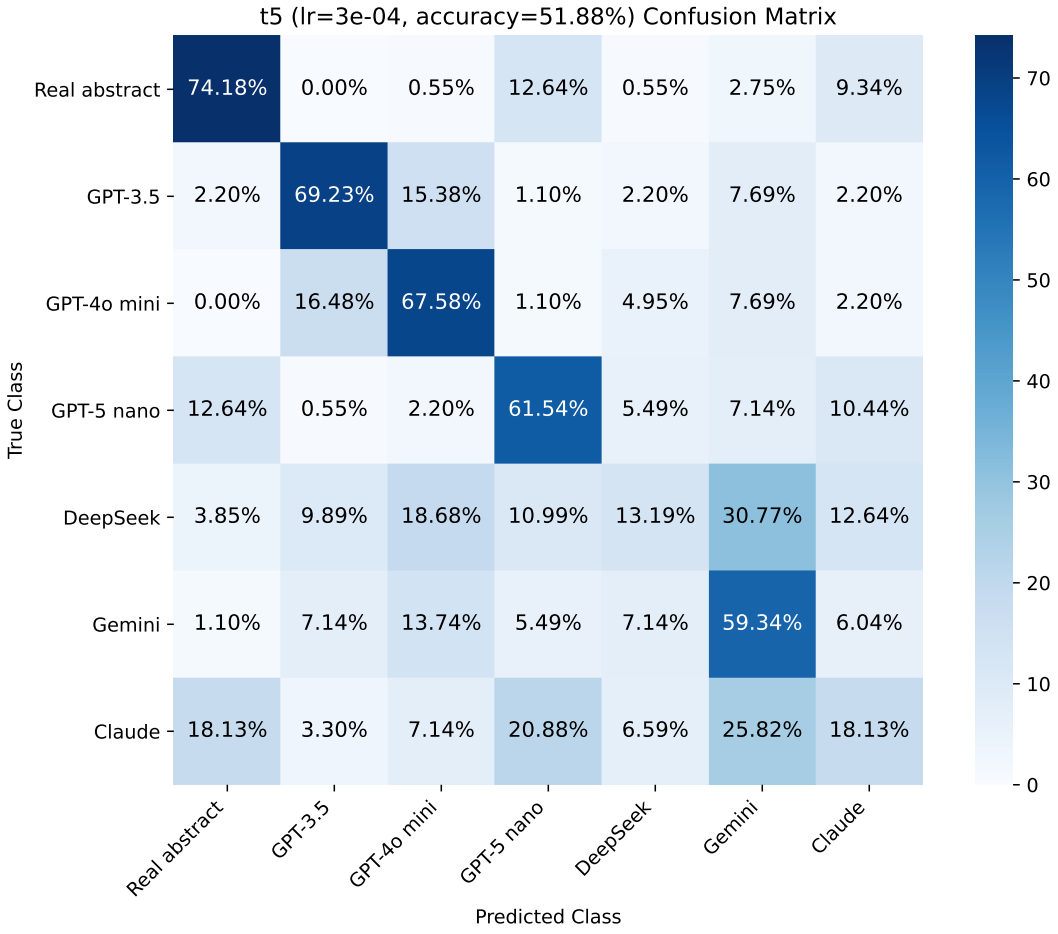


Figure 17: Confusion matrix for 7-class T5 classification (lr=3e-04)

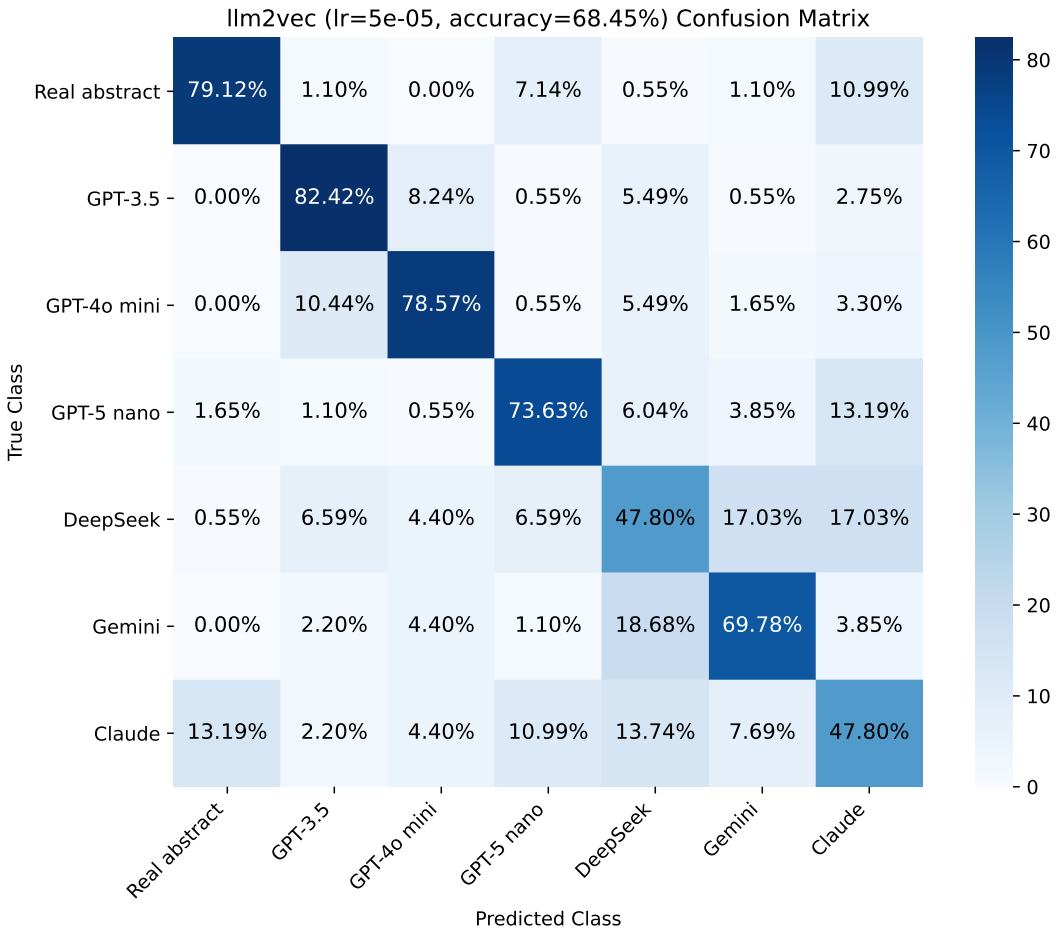


Figure 18: Confusion matrix for 7-class LLM2Vec classification (lr=5e-05)

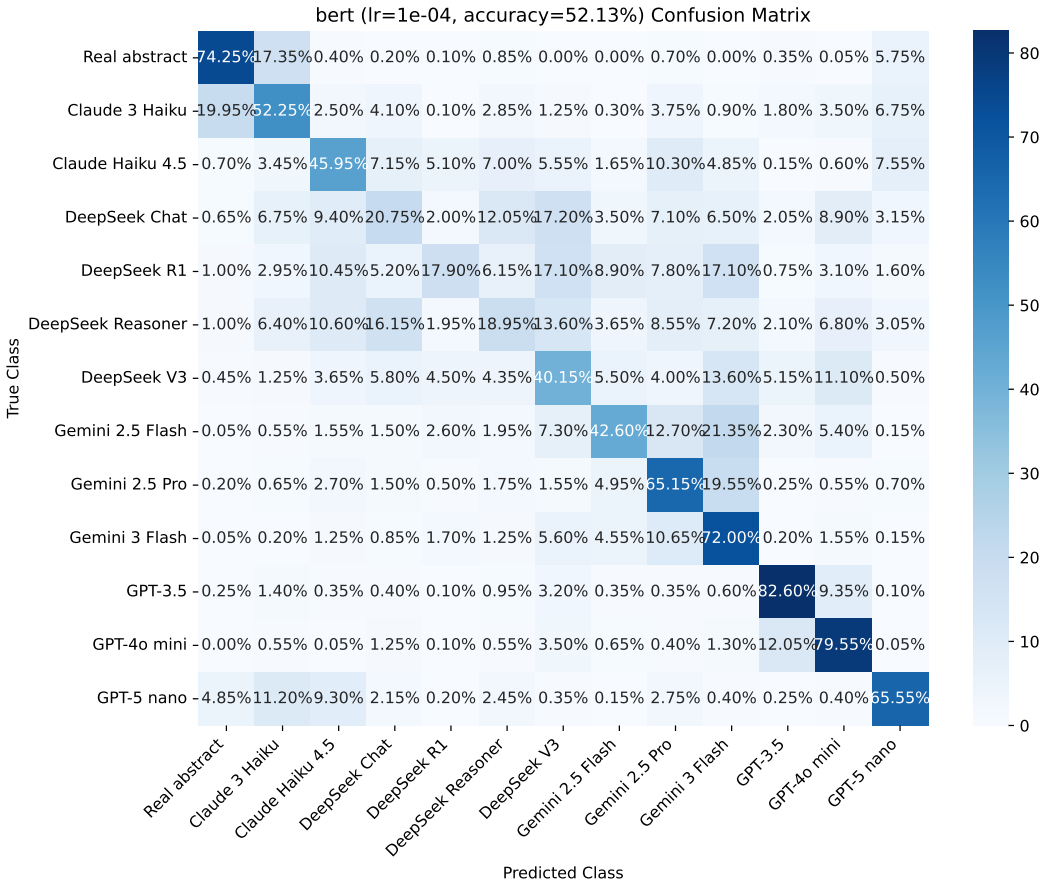


Figure 19: Confusion matrix for 13-class BERT classification (lr=1e-04)

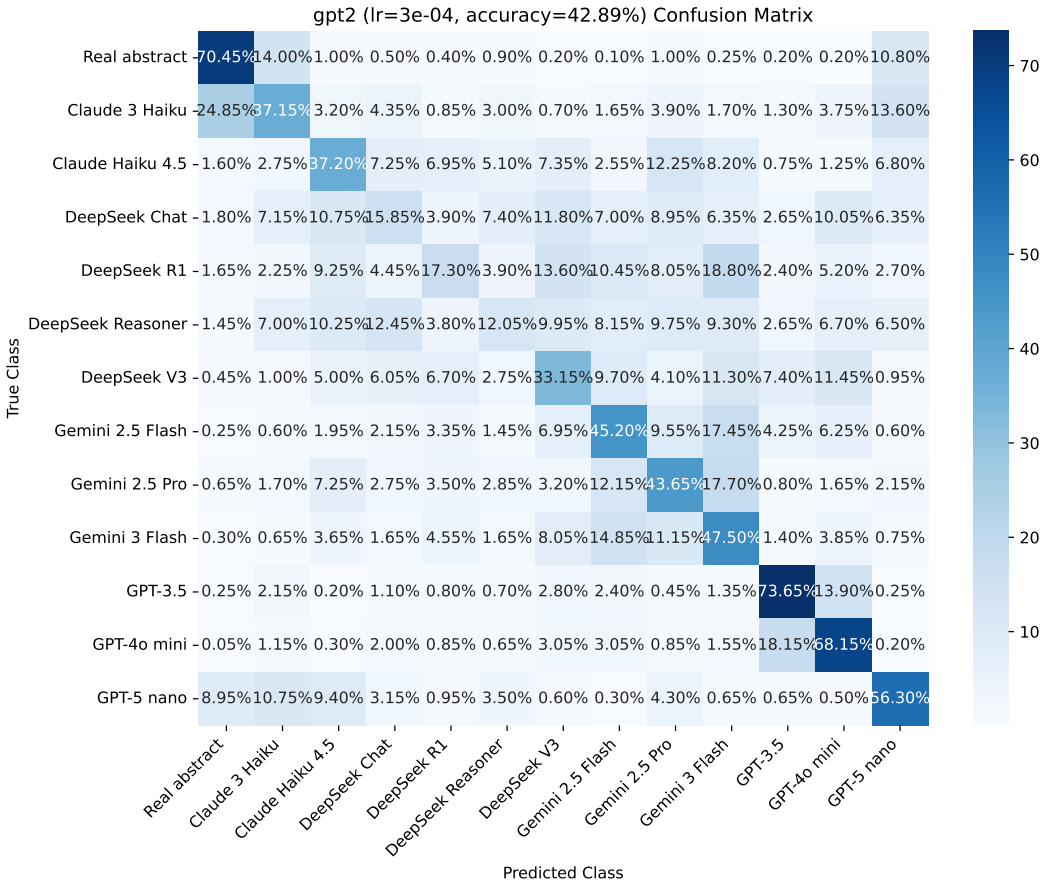


Figure 20: Confusion matrix for 13-class GPT-2 classification (lr=3e-04)

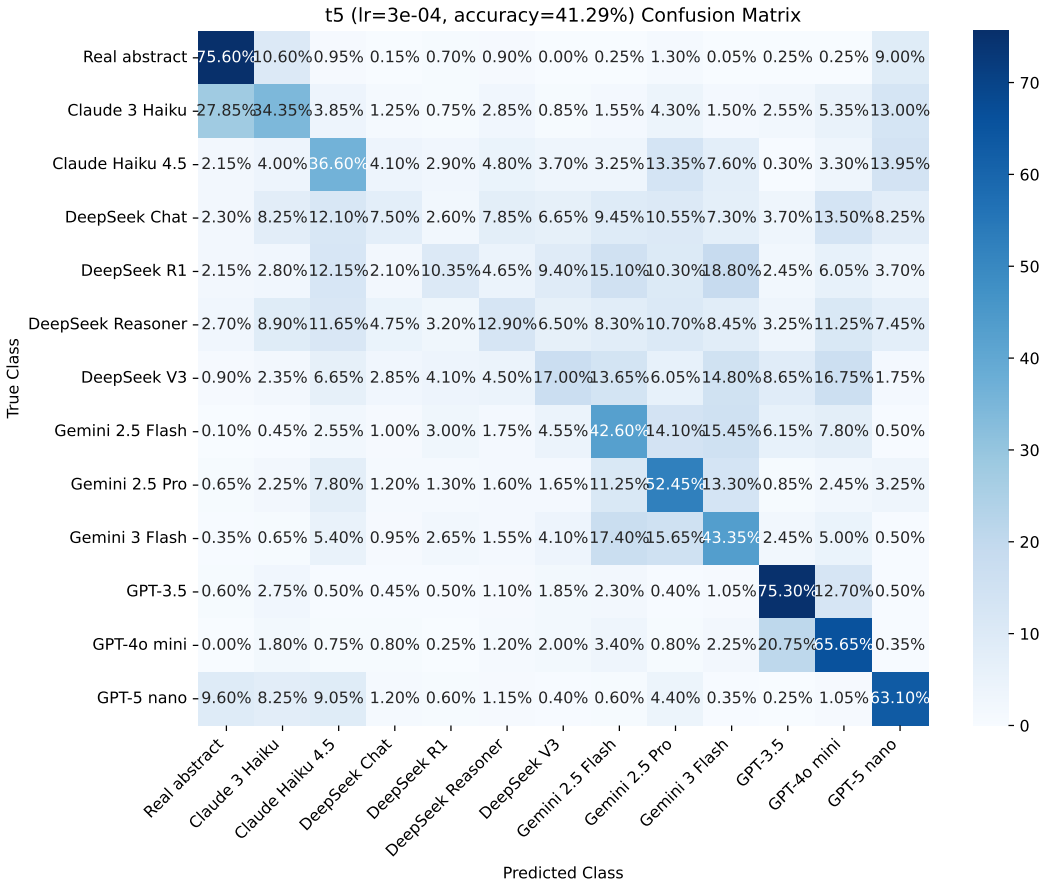


Figure 21: Confusion matrix for 13-class T5 classification (lr=3e-04)

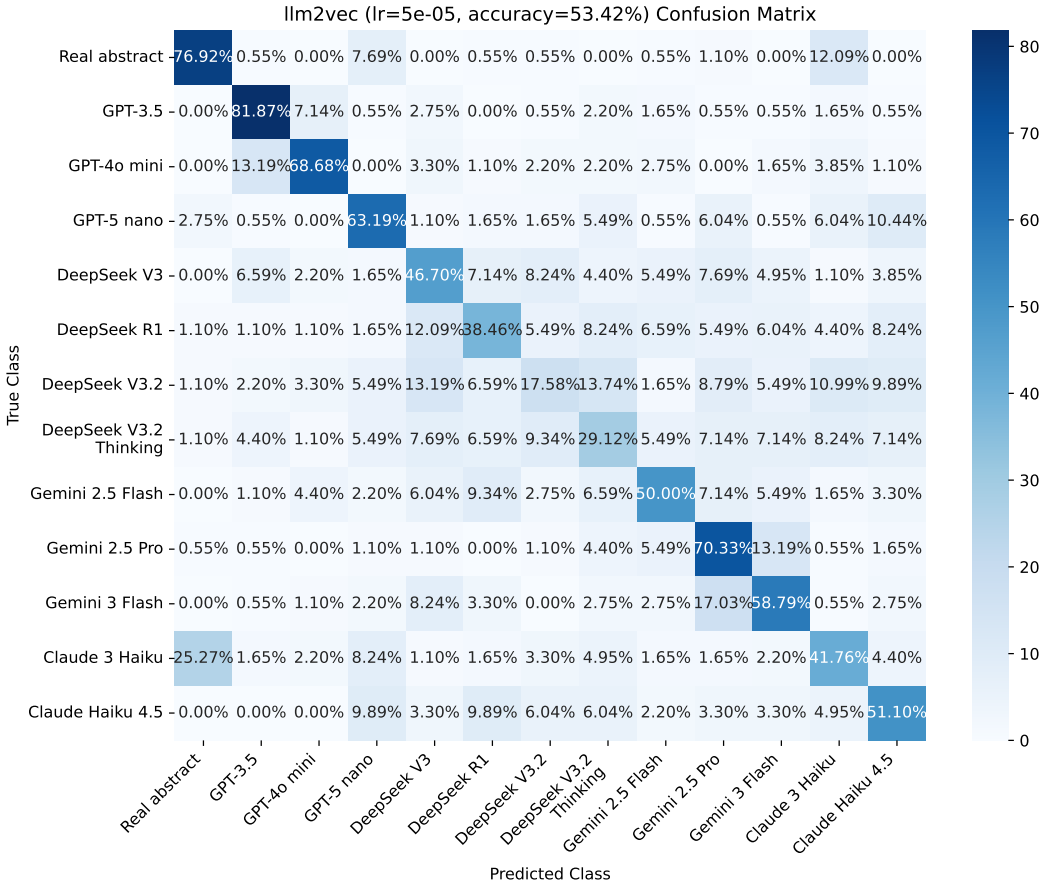
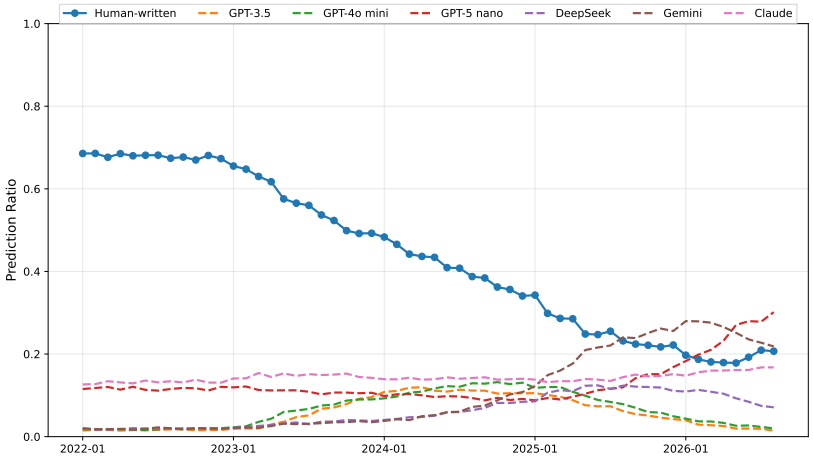
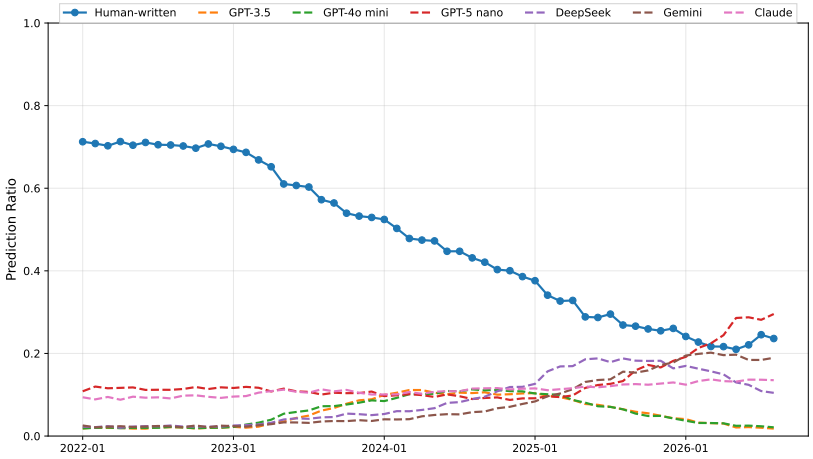


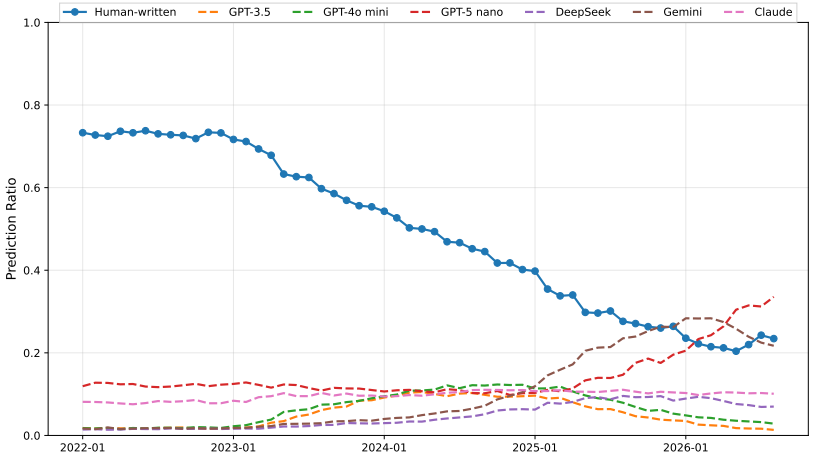
Figure 22: Confusion matrix for 13-class LLM2Vec classification (lr=5e-05)



(a) BERT-based classifier

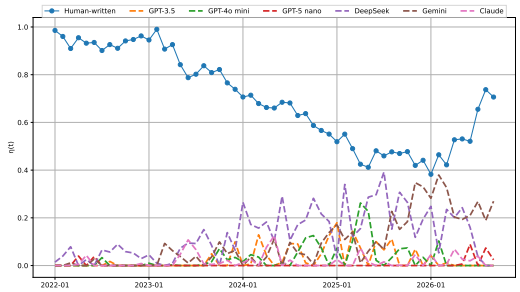


(b) GPT-2-based classifier

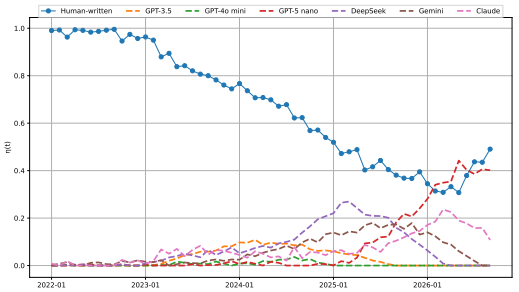


(c) T5-based classifier

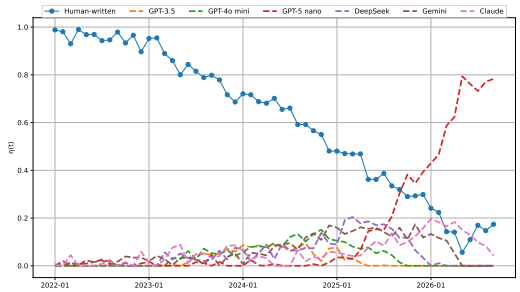
Figure 23: Classification results of real arXiv abstracts using different classifiers.



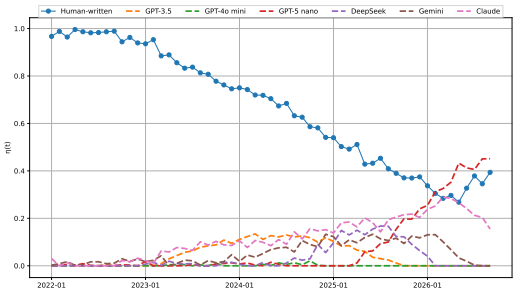
(a) The 126 words used for estimation are drawn from NLTK's English stopwords in the Google Books Ngram dataset that occur in the 2,000 cleaned abstracts used for simulation.



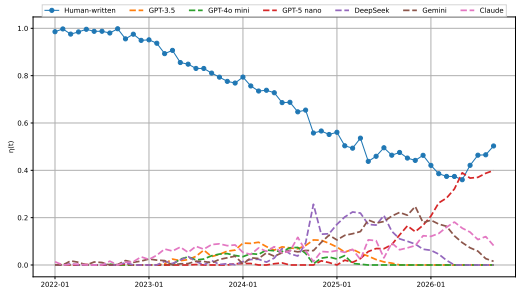
(b) The 2,741 words used for estimation are drawn from the 10,000 most frequent words in the Google Books Ngram dataset that occur at least 10 times in the 2,000 preprocessed abstracts used for simulation.



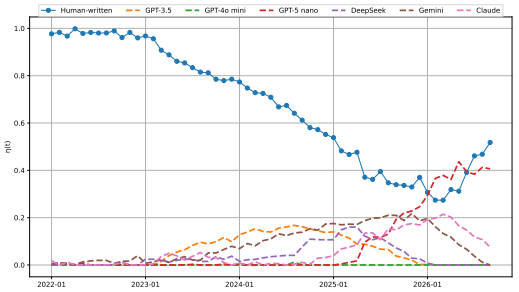
(c) The 695 words used for estimation are drawn from the 1,000 most frequent words in the Google Books Ngram dataset that occur at least 10 times in the 2,000 preprocessed abstracts used for simulation.



(d) The 2,088 words used for estimation are drawn from the 5,000 most frequent words in the Google Books Ngram dataset that occur at least 10 times in the 2,000 preprocessed abstracts used for simulation.

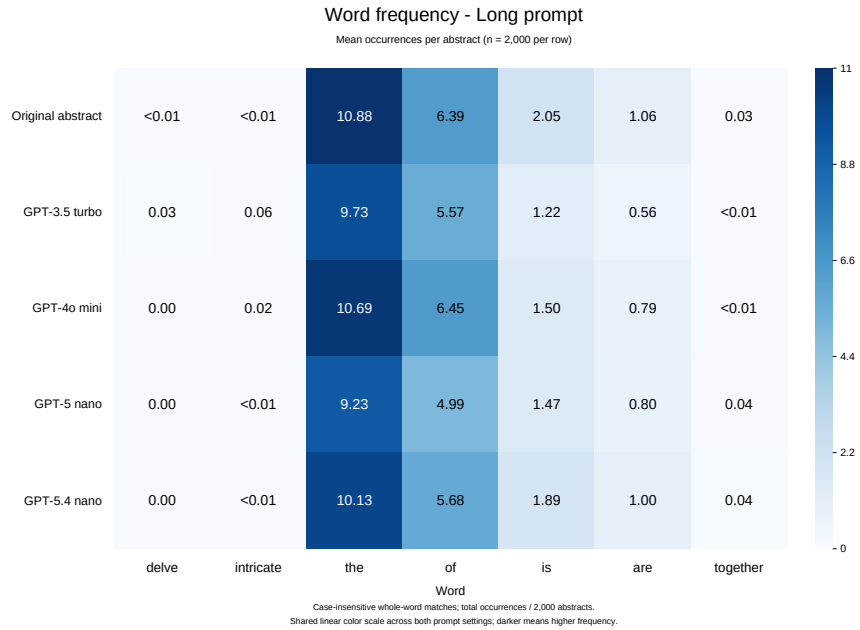


(e) The 3,779 words used for estimation are drawn from the 10,000 most frequent words in the Google Books Ngram dataset that occur at least 5 times in the 2,000 preprocessed abstracts used for simulation.



(f) The 1,885 words used for estimation are drawn from the 10,000 most frequent words in the Google Books Ngram dataset that occur at least 20 times in the 2,000 preprocessed abstracts used for simulation.

Figure 24: Supplementary results for the estimation of LLM impact in arXiv abstracts.

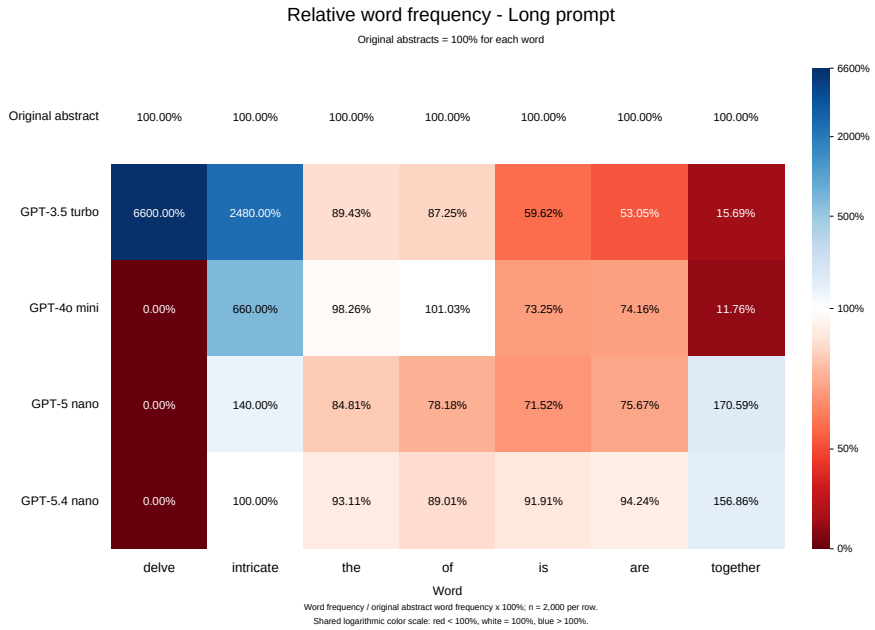


(a) Long prompt.

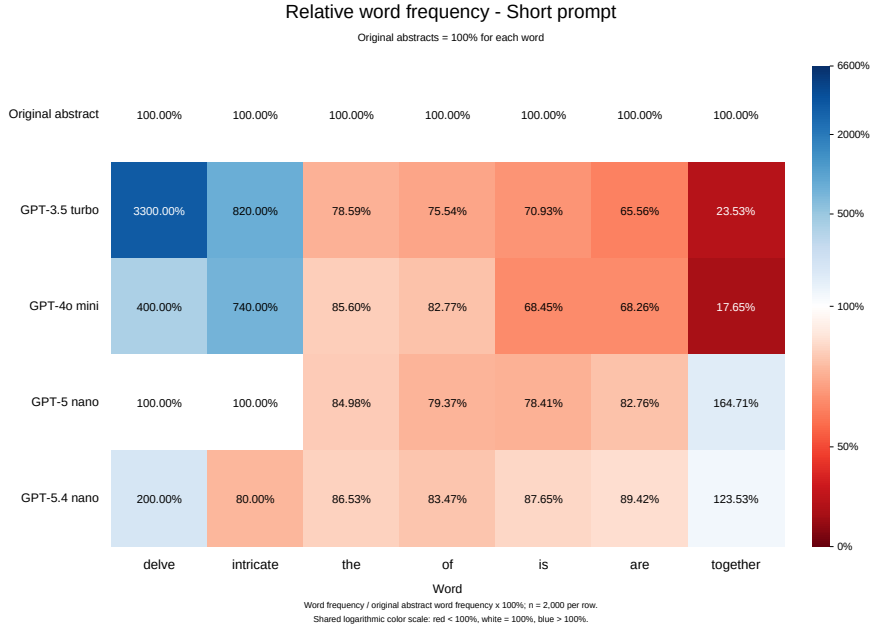


(b) Short prompt.

Figure 25: Mean word frequency across models and prompt settings. Values denote occurrences per abstract over 2,000 abstracts.



(a) Long prompt.



(b) Short prompt.

Figure 26: Relative word frequency across models and prompt settings. Original abstracts provide the 100% baseline for each word within the same prompt setting.