Exploring the Limitations of Detecting Machine-Generated Text

Jad Doughman[1], Osama Mohammed Afzal[1], Hawau Olamide Toyin[1], Shady Shehata[1], Preslav Nakov[1], Zeerak Talat[2]
[1]Mohamed bin Zayed University of Artificial Intelligence [2]University of Edinburgh
{jad.doughman, preslav.nakov}@mbzuai.ac.ae, z@zeerak.org

Abstract

Recent improvements in the quality of the generations by large language models have spurred research into identifying machine-generated text. Such work often presents high-performing detectors. However, humans and machines can produce text in different styles and domains, yet the the performance impact of such on machine generated text detection systems remains unclear. In this paper, we audit the classification performance for detecting machine-generated text by evaluating on texts with varying writing styles. We find that classifiers are highly sensitive to stylistic changes and differences in text complexity, and in some cases degrade entirely to random classifiers. We further find that detection systems are particularly susceptible to misclassify easy-to-read texts while they have high performance for complex texts, leading to concerns about the reliability of detection systems. We recommend that future work attends to stylistic factors and reading difficulty levels of human-written and machine-generated text.

1 Introduction

Recent developments for large language models (LLMs) have enabled the generation of text that mimics human writing in coherence and style, which can be used for benign (e.g., drafting an e-mail) or for nefarious (e.g., generating misinformation at scale) purposes. To mitigate the risks of machine-generated text (MGT), research has devoted efforts to building MGT detectors (e.g., Wang et al., 2024; Koike et al., 2024; Abdalla et al., 2023).Such systems often achieve promising performance on in-domain datasets, but may not generalize to out-of-domain data (Wang et al., 2024). This suggests that MGT detection systems may be more apt for some data than for others. Here, we audit two state-of-the-art MGT detection methods by subsampling their evaluation datasets using linguistic features and readability measures and compare model performances on these subsets.

In an effort to investigate the limitations of MGT detectors, we evaluate the sensitivity of current detectors to stylistic variations and text complexity. Specifically, we examine two categories of detection systems trained on different domains, and categorize their test sets using linguistic features, and metrics for lexical composition, sophistication, and diversity, and readability. We then evaluate model performances on different ranges for each feature category and report performances for each subset.

We find that the classifiers are highly sensitive to the distribution of part-of-speech classes, e.g., adverbs, to stylistic features, e.g., average sentence length, and to surface-level artefacts, e.g., punctuation. For instance, we find that the F1-score of detectors drops from $0.4 \rightarrow 0.0$ and $0.6 \rightarrow 0.3$ for different ratios of adverbs in human-written and machine-generated text. Our findings suggest that performance of detectors across domains and styles is likely over-estimated. We therefore call for care in using such tools for critical societal functions, e.g., plagiarism detection in education, and recommend that future work attends to linguistic and stylistic artefacts in benchmark datasets.

2 Related Work

Prior work has sought to detect MGT by using on feature-based (e.g., Fröhling and Zubiaga, 2021; Prova, 2024) and neural network-based (e.g., Gaggar et al., 2023) methods, reporting over 80% and 90% accuracy, respectively. This body of work has primarily used three feature types for MGT detection: statistical distributions (e.g., log-likelihood) (e.g., Gehrmann et al., 2019), features obtained from fact-checking methods (e.g., Wang et al., 2024), and linguistic features (e.g., Tang et al., 2023).

Other work has proposed zero-shot approaches to MGT detection: For instance Mitchell et al. (2023) rely on log-probabilities from the generating model and random perturbation of the text

Proceedings of the 31st International Conference on Computational Linguistics, pages 4274–4281
January 19–24, 2025.

Model	Training Data	Evaluation Data	Macro F1-Score	Drop (%)
LR-GLTR	ArXiv (C-GPT & GPT-3.5)	ArXiv (C-GPT)	0.95	-
	ArXiv (C-GPT & GPT-3.5)	ArXiv (CO)	0.92	↓ 3.16%
	ArXiv (C-GPT & GPT-3.5)	ArXiv (GPT-3.5)	0.79	↓ 16.84%
	ArXiv (C-GPT & GPT-3.5)	OUTFOX (C-GPT)	0.60	↓ 36.84%
	ArXiv (C-GPT & GPT-3.5)	IDMGSP (C-GPT, GA)	0.53	↓ 42.11%
	OUTFOX (C-GPT)	OUTFOX (C-GPT)	0.91	-
	OUTFOX (C-GPT)	IDMGSP (C-GPT, GA)	0.53	↓ 41.76%
RoBERTa	ArXiv (C-GPT & GPT-3.5)	ArXiv (C-GPT)	0.99	-
	ArXiv (C-GPT & GPT-3.5)	IDMGSP (C-GPT, GA)	0.33	↓ 66.00%
GPT-Large	OPEN AI Detector	GPT2 Generations	0.95	-
	OPEN AI Detector	IDMGSP (C-GPT, GA)	0.80	↓ 15.00%
Llama-3.1-8B	Zero Shot	IDMGSP (C-GPT, GA)	0.50	-

Table 1: Comparison of in-domain and out-of-domain performance of detectors. The "Drop" column represets the decrease in F1-score from the in-domain configuration to the out-of-domain configuration.

from another generic LLM; and Guo and Yu (2023) use a black-box LLM to denoise input text with artificially added noise, and then semantically compare the denoised and original text. Yet other work has examined the use of watermarks for MGT as a mechanism for detecting MGT. For example, Kirchenbauer et al. (2023) propose using soft-constraints through *green* and *red* lists of vocabulary to include or exclude from MGT.

Recent work has also conducted comprehensive analyses of MGT detection methods and resources (e.g., Tang et al., 2023; Jawahar et al., 2022; Mitchell et al., 2023; Guo and Yu, 2023). Tang et al. (2023), for example, highlight for the need measures for evaluating MGT detection systems. They argue that current evaluation measures (e.g., AUC and accuracy) are limited for security analysis by only considering the average instance and are limiting for security analysis. Similarly, watermarks for MGT have been called into question, with Zhang et al. (2024) arguing that "strong watermarking of generative models is impossible."

Research has therefore attempted to develop datasets for detecting MGT (e.g., Wang et al., 2024; Koike et al., 2024; Abdalla et al., 2023; Radford and Wu, 2019). Such datasets typically contain human-written texts for given domains, and the generated outputs of LLMs that have been conditioned on partial information from the human written texts (e.g., Wang et al., 2024; Guo et al., 2023; He et al., 2023).

3 Experiments

We evaluate MGT detection using three datasets and four classifiers, and use linguistic features for analysis. Here, we describe our experimental setup.

3.1 Data

We conduct experiments using three datasets: M4, OUTFOX, and IDMGSP.

M4 The M4 dataset (Wang et al., 2024) consists of 147K human-written texts across data sources and languages, paired with human-written and MGTs generated by several LLMs. For our experiments, we use the English subset of the M4 dataset, which consists of 102K human-written texts, sourced from Wikipedia, WikiHow, Reddit, ArXiv and PeerRead, and outputs from GPT-4, ChatGPT and text-davinci-003 (henceforth GPT-3.5).

OUTFOX The OUTFOX dataset (Koike et al., 2024) consists of 15K triplets of essay problem statements, student-written essays, and machine-generated essays. We use human-written and ChatGPT-generated essays for training, and GPT-3.5-generated essays for testing.

IDMGSP The IDMGSP dataset (Abdalla et al., 2023) contains 4K human-written and 4K machine-generated (SCIgen, GPT-2, ChatGPT, and Galactica) scientific papers. We restrict our analysis to abstracts of scientific papers because they are similar in length, which allows for a fair comparative evaluation across different samples.

3.2 Machine-Generated Text Classifiers

We evaluate neural and feature-based methods for MGT detection. The neural methods rely on fine-tuning LLMs, while the feature-based methods rely on machine-generated features for MGT detection.

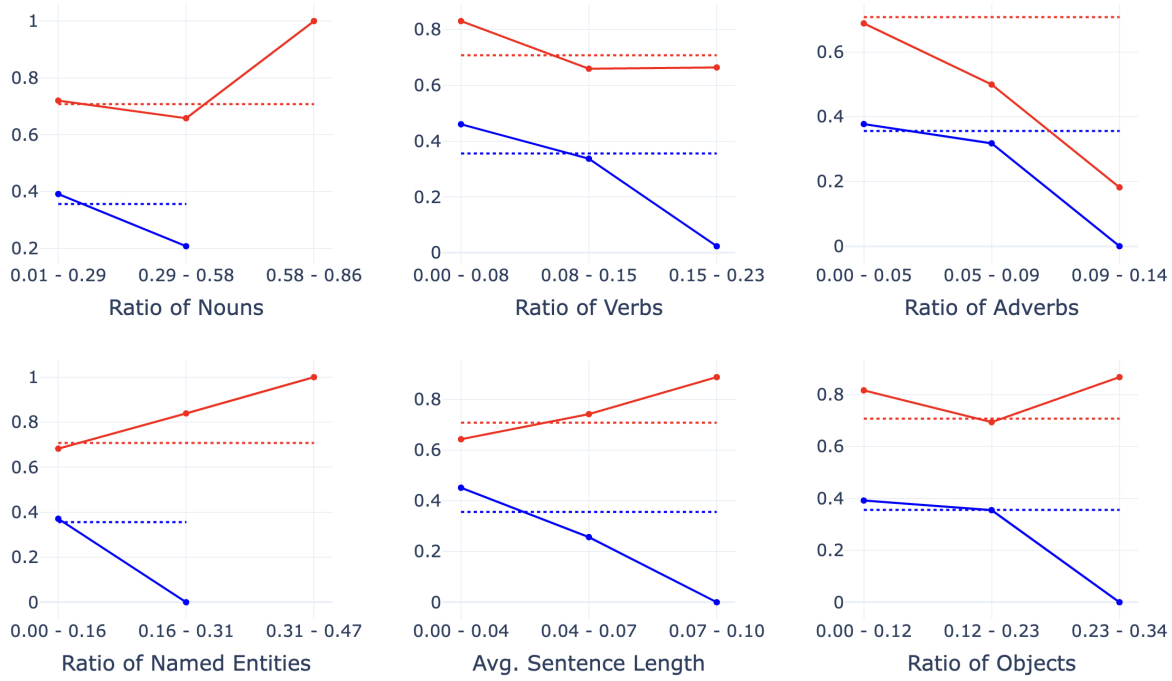

Figure 1: F1-scores for LR-GLTR (trained on M4) for IDMGSP. Red indicates machine-generated, blue human-written data, and dashed lines indicate baselines.

Neural Methods We use two fine-tuned version of RoBERTa (Liu et al., 2019): the **OpenAI Detector**, a RoBERTa-Large model fine-tuned on GPT-2-generated texts, and **RoBERTa-M4**, a RoBERTa-base model fine-tuned on the M4 dataset following Solaiman et al. (2019) and Wang et al. (2024).

Feature-Based Methods Following Wang et al. (2024), we use **LR-GLTR**, a logistic regression model trained using 14 features from Gehrmann et al. (2019). The model uses two sets of features: The number of tokens within the top-{10, 100, 1000, 1000+} ranks from a LM's predicted probability distribution (4 features); the probability distribution for a given word divided by the maximum probability for any word in the same position over 10 bins ranging from 0.0 to 1.0 (10 features). We train one instance of this model on a subset of M4, and another instance on subset of OUTFOX.

Zero-Shot Prediction Methods We use **Llama-3.1-8b** for zero-shot classification of machine-generated versus human-written text.

3.3 Linguistic Features for Analysis

We extract linguistic and extra-linguistic features for analysis. Specifically, we extract **Part-of-Speech (POS)** tags and named entities using spaCy (Honnibal and Montani, 2017). We then compute **average sentence length**, and the **ratio of nouns, verbs, adverbs, named entities, and objects** (direct or prepositional) in a text, i.e., the number of occurrences divided by the total number of tokens in the text. We also compute the **Flesch Reading Ease** score (Flesch, 1948) to assess the reading difficulty of a text. This metric is computed using sentence length, syllable density, and word familiarity. Finally, we compute the lexical diversity of texts, i.e., the variety and range of words used. Specifically, we compute **Hapax (Legomena)**, the number of words that occur only once in a text, and **Dihapax (Dis Legomena)**, the number of words that occur twice in a text. Hapax and Dihapax help illustrate the richness of the vocabulary used.

4 Analysis

Here, we investigate the sources of classification errors (see Table 1 for impacts of domain shifts).

Impact of Surface Form Linguistic Features Ideally, a MGT detector would not overfit to linguistic surface features, however, we find that the **LR-GLTR** model significantly overfits to such features (see Figure 1).[1] For instance, we see that the model performance for both human-written and machine-generated text drops to near zero as the ratio of adverbs increases. Moreover, as we the

[1]Only machine-generated texts appear in the (0.58–0.86] noun ratio range.

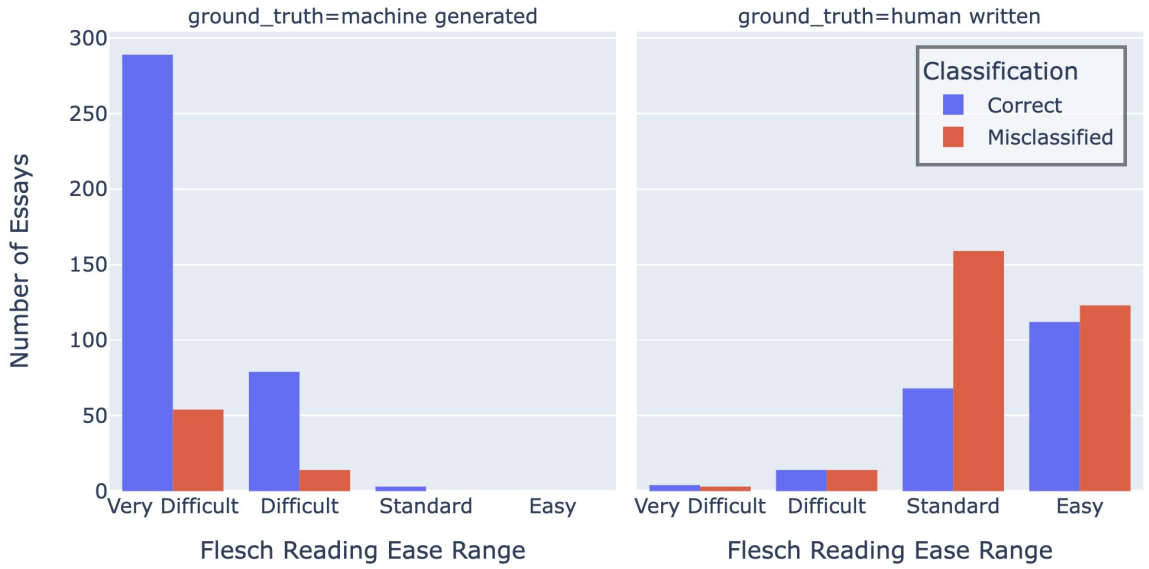

Figure 2: Flesch Reading Ease analysis of LR-GLTR (ArXiv) evaluated on OUTFOX (essays).

ratio of named entities, objects, and the average sentence lengths increase, the model performances drop to zero for human-written texts, while obtaining near perfect scores for machine generated texts. That is, beyond a given ratio of linguistic surface form items, and average sentence length, the model loses the capability to identify human-written text.

4.1 Impact of Readability

Turning our attention to readability (see Figure 2), we find that LR-GLTR has high accuracy for very difficult passages, as it has very few classification errors. However, the model's ability to correctly classify MGT decreases as the reading difficulty decreases. For human-written text, a different pattern emerges: The model struggles to correctly classify human-written texts regardless of text difficulty.

4.2 Impact of Punctuation Marks

To investigate RoBERTa's sizable performance drop (see Table 1) on out-of-domain evaluation sets, we use Shapley Additive Explanations (SHAP) (Lundberg and Lee, 2017), which quantifies the impact of a given feature on a model's performance. We find that punctuation marks and whitespace (see Figure 3) are among the most important features. Such over-reliance on punctuation suggests that the model is overfitting and therefore not learning general features of MGT.

Across both RoBERTa and LR-GLTR models, it appears that surface level features are highly influential for classifier performance. In turn, this suggests that simple adversarial attacks such as changing the ratio of nouns, adverbs, or changing punctuation can render these models ineffective.

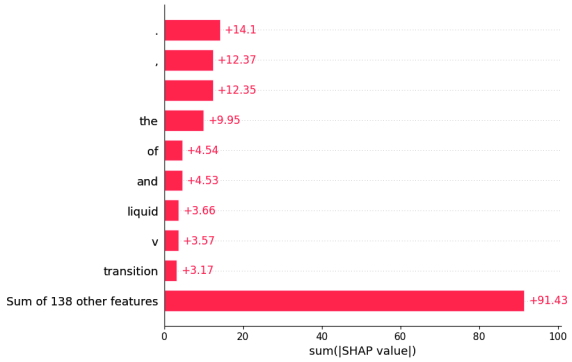

Figure 3: Feature importance for RoBERTa trained on ArXiv using SHAP values.

4.3 Impact of Lexical Diversity

Considering lexical diversity (see Figure 4), the classifier performs best when detecting texts with a narrow vocabulary (low Hapax bins) and specific repetition patterns (high Dihapax bins). For example, the model has high performance for Hapax bins 0 and 1, when combined with Dihapax bins 6 and 7. In contrast, the model struggles with texts that have a rich and varied vocabulary (high Hapax) and certain combinations of repetitions.

4.4 Impact of Named Entities

We conducted an analysis to evaluate the impact of the Named Entity Recognition (NER) ratio—defined as the number of named entities relative to the total token count—on the zero-shot classi-

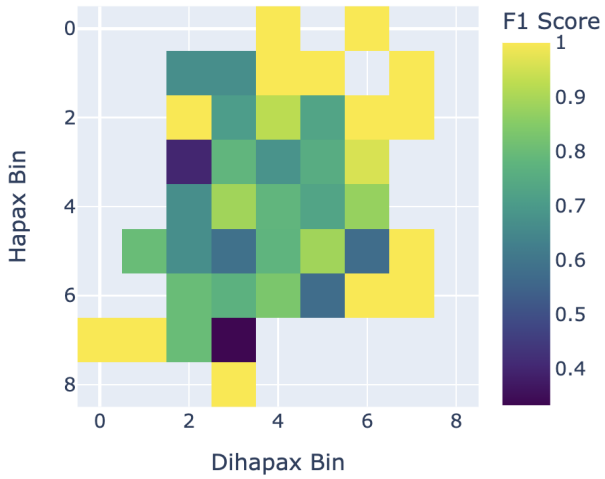

Figure 4: F1 scores for LR-GLTR trained on ArXiv (ChatGPT & GPT-3.5) and tested on ArXiv (GPT-3.5) across bins of hapax and dihapax features.

fication performance of scientific abstracts using Llama-3.1-8B (see Figure 5 and Figure 6). We find that correctly classified abstracts have a broader range of NER ratios, while incorrectly classified abstracts concentrate named entities in the lower rations. That is, the model is more prone to misclassify abstracts with fewer named entities. We test this finding by computing Welch's T-test for NER ratios of correctly and incorrectly classified samples. We find that the hypothesis—that the two distributions have the same mean—to be rejected ($t = -2.289$, *degrees of freedom* $= 1680$, *critical* $t = 1.96$, *p-value*$= 0.022$), which indicates a statistically significant difference in the means ($p < 0.05$). The t-test suggests that the mean NER ratio is lower for incorrectly classified abstracts, i.e., that performance decreases as NER ratio decreases.

Figure 5: NER ratio for classified abstracts.

5 Risk of Deployment

Deploying MGT detectors, such as the ones evaluated above, comes with risks of reliability and fairness. The primary risks associated with deploying such systems are detailed below.

Adversarial attacks The classifier's over-reliance on surface-level features such as writing style, punctuation marks, and whitespace makes it vulnerable to adversarial attacks. Adversaries can intentionally alter these features to cause misclassification.

Bias and fairness Over-fitting to a specific writing style can lead to unfair misclassification of subgroups with a specific writing style. This can result in biased outcomes, particularly against individuals from different cultural, educational, or linguistic backgrounds—especially in contexts that encourage the use of richer vocabulary and longer sentences.

Data drift due to new LLMs The writing style and fluency of MGT change upon the release of new models, which can cause data drift, potentially rendering a classifier ineffective. As a result, classifiers may need regular retraining to accurately detect machine-generated text from newer models.

Domain shift sensitivity The results above indicate that although a classifier performs well within the same domain, it may be sensitive to domain shift. This sensitivity could limit a classifier's applicability and deployment in diverse settings.

6 Conclusion

In this paper, we have examined the limitations of several classifiers for detecting machine-generated text by evaluating their sensitivity to stylistic variation across domains. We find that classifiers show high sensitivity to certain linguistic features, e.g., the distribution of adverbs, sentence length, and readability of the text. Moreover, we find that classifiers overfit to punctuation marks and whitespace. Our results suggest that current datasets for MGT are not robust to stylistic or domain shifts, and are particularly weak when applied to simple writing, e.g., school assignments. We therefore call for the further development of datasets of MGT and critical assessments of MGT detection systems with data from their particular domain of interest to avoid potential negative consequences of misclassification in critical domains.

Limitations

Below are some of our main limitation pertaining to availability of labelled data and the dynamic nature of LLM generation:

- **Dataset limitations:** The datasets used in this paper do not represent the full spectrum of potential domains. This is caused by the limited availability of labeled MGT data.
- **Dynamic nature of LLMs:** We assumed that text generation by a given model are static. However, LLMs are regularly updated and may exhibit changes in their writing style and coherence. However, such changes will typically cause detectors to fail beyond what is described in this work, further emphasizing the need for more careful data analysis.

Ethical Considerations

Our paper investigates the performance of models for the detection of machine-generated text and emphasizes the careful testing and precise reporting of the performance of such systems. This is particularly important, as our examined models struggle on less complex texts, which can have downstream impact if such systems are deployed in educational settings. In light of our findings, we stress the importance of critically evaluating systems for detecting machine-generated text within the domains a given model is to be deployed.

References

Mohamed Hesham Ibrahim Abdalla, Simon Malberg, Daryna Dementieva, Edoardo Mosca, and Georg Groh. 2023. A benchmark dataset to distinguish human-written and machine-generated scientific papers. *Information*, 14(10).

Rudolph Flesch. 1948. A new readability yardstick. *Journal of Applied Psychology*, 32(3):221–233.

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

Raghav Gaggar, Ashish Bhagchandani, and Harsh Oza. 2023. Machine-generated text detection using deep learning.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander Rush. 2019. GLTR: Statistical detection and visualization of generated text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 111–116, Florence, Italy. Association for Computational Linguistics.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*.

Zhen Guo and Shangdi Yu. 2023. Authentigpt: Detecting machine-generated text via black-box language models denoising.

Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2024. Spotting llms with binoculars: Zero-shot detection of machine-generated text. *arXiv preprint arXiv:2401.12070*.

Xinlei He, Xinyue Shen, Zeyuan Chen, Michael Backes, and Yang Zhang. 2023. Mgtbench: Benchmarking machine-generated text detection. *arXiv preprint arXiv:2303.14822*.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Ganesh Jawahar, Muhammad Abdul-Mageed, and Laks Lakshmanan. 2022. Automatic detection of entity-manipulated text using factual knowledge. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 86–93, Dublin, Ireland. Association for Computational Linguistics.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 17061–17084. PMLR.

Ryuto Koike, Masahiro Kaneko, and Naoaki Okazaki. 2024. Outfox: Llm-generated essay detection through in-context learning with adversarially generated examples. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(19):21258–21266.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, NIPS'17, pages 4768–4777, Red Hook, NY, USA. Curran Associates Inc. Number of pages: 10 Place: Long Beach, California, USA.

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

Nuzhat Prova. 2024. Detecting ai generated text based on nlp and machine learning approaches.

Alec Radford and Jeff Wu. 2019. [link].

Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, Gretchen Krueger, Jong Wook Kim, Sarah Kreps, et al. 2019. Release strategies and the social impacts of language models. *arXiv preprint arXiv:1908.09203*.

Ruixiang Tang, Yu-Neng Chuang, and Xia Hu. 2023. The science of detecting llm-generated texts.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Toru Sasaki, Thomas Arnold, Alham Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024. M4: Multi-generator, multi-domain, and multilingual black-box machine-generated text detection. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1369–1407, St. Julian's, Malta. Association for Computational Linguistics.

Hanlin Zhang, Benjamin L. Edelman, Danilo Francati, Daniele Venturi, Giuseppe Ateniese, and Boaz Barak. 2024. Watermarks in the sand: Impossibility of strong watermarking for generative models. In *Forty-first International Conference on Machine Learning*.

A Distribution of NER Ratios

Here, we include a histogram (complementing the violin plot in Figure 5) to illustrate the distribution of the NER ratio for correctly and incorrectly classified abstracts.

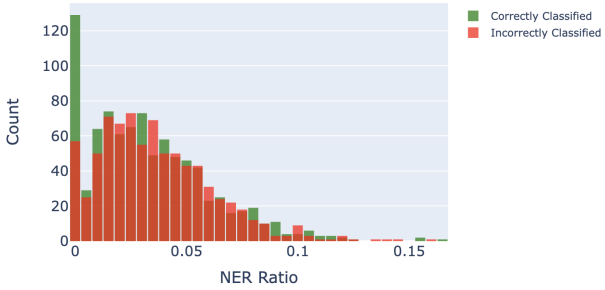

Figure 6: Distribution of NER Ratios (Histogram)

B Part of Speech Analysis

Here, we extend the part-of-speech (POS) subsampling evaluation with a broader coverage of models to explore properties influencing classifier performance across different models and domains. The evaluated models include zero-shot detection methods (LLama 3.1 8b and Binoculars (Hans et al., 2024)) and variants of the LR-GLTR models.

The results indicate that POS features do not generalize to out-of-domain samples (as seen in Figure 7) but retain F scores above 0.5 across in-domain examples (as seen in Figure 8).

The results in Figure 9 and Figure 10 indicate that zero-shot detection methods for identifying machine-generated text, such as Binoculars, are heavily dependent on the length of the sentence. When evaluating longer sentences, the F scores degrade from around 0.9 to 0.3 across both classes. In certain adverb ratios, the F-score drops to 0 for machine-generated text. This suggests that zero-shot detection methods fixate on what are believed to be common features of machine-generated text (longer sentences and more adverbs).

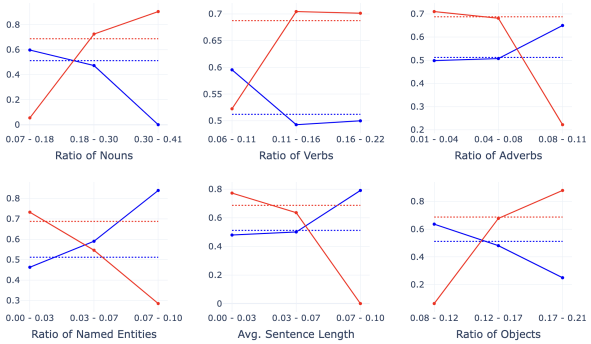

Figure 7: GLTR Logistic Regression: Train ArXiv, Test Essays (ChatGPT). Red indicates machine-generated, blue human-written data, and dashed lines indicate baselines.

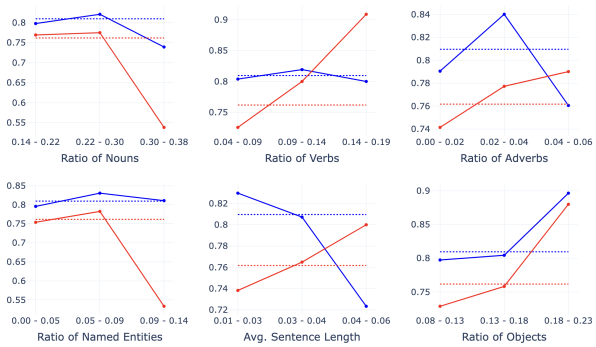

Figure 8: GLTR Logistic Regression: Train ArXiv, Test ArXiv (Davinci). Red indicates machine-generated, blue human-written data, and dashed lines indicate baselines.

Figure 9: Binoculars Zero-Shot Detection. Red indicates machine-generated, blue human-written data, and dashed lines indicate baselines.

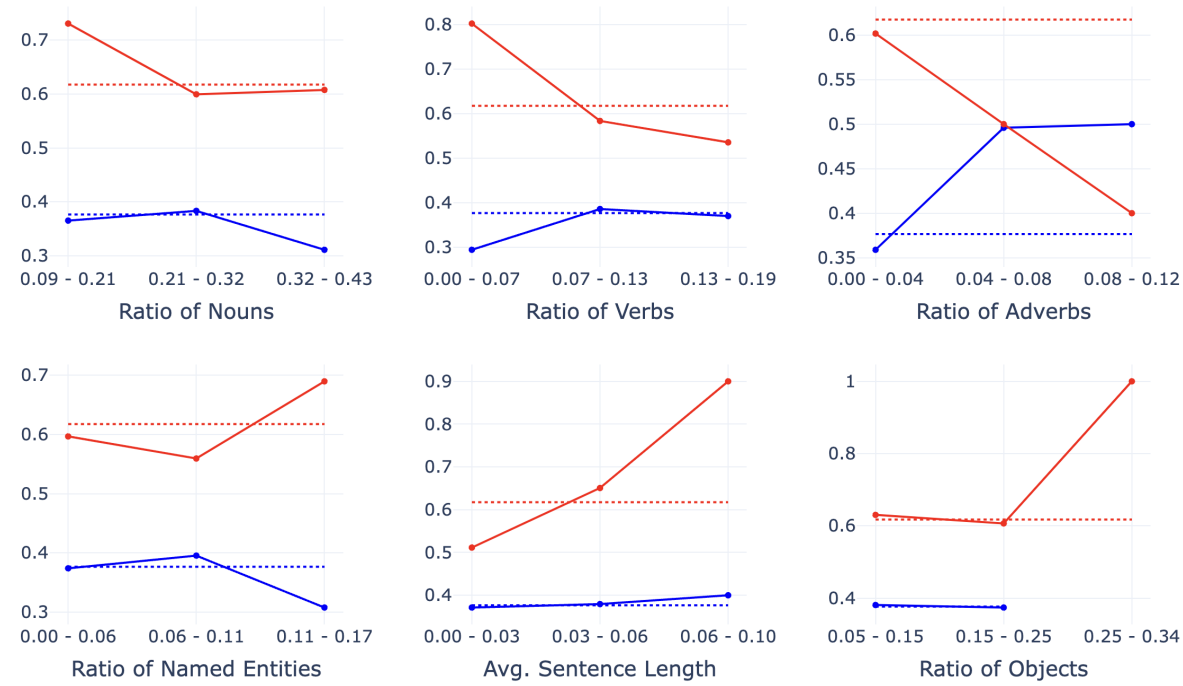

Figure 10: Llama 3.1 8b Zero-Shot Detection. Red indicates machine-generated, blue human-written data, and dashed lines indicate baselines.