Your Large Language Models Are Leaving Fingerprints

Hope McGovern [*] [†]
Cambridge Computer Lab

Rickard Stureborg[†]
Duke University
Grammarly

Yoshi Suhara[†]
NVIDIA

Dimitris Alikaniotis
Grammarly

Abstract

It has been shown that fine-tuned transformers and other supervised detectors are effective for distinguishing between human and machine-generated texts in non-adversarial settings, but we find that even simple classifiers on top of n-gram and part-of-speech features can achieve very robust performance on both in- and out-of-domain data. To understand how this is possible, we analyze machine-generated output text in four datasets, finding that LLMs possess unique *fingerprints* which manifest as slight differences in the frequency of certain lexical and morphosyntactic features. We show how to visualize such fingerprints, describe how they can be used to detect machine-generated text and find that they are even robust across text domains. We find that fingerprints are often persistent across models in the same model family (e.g. 13B parameter LLaMA's fingerprint is similar to that of 65B parameter LLaMA) and that while a detector trained on text from one model can easily recognize text generated by a model in the same family, it struggles to detect text generated by an unrelated model.

1 Introduction

Large language models (LLMs) produce text often indistinguishable from human-authored text to human judges (Clark et al., 2021). This unfortunately allows potential misuses such as academic plagiarism (Westfall, 2023) and the dissemination of disinformation (Barnett, 2023), which has therefore prompted interest in machine-generated text detection (MGT). We conduct linguistic analysis on four popular published datasets for MGT, showing that the machine-generated content in each shows linguistic markers in aggregate which make it relatively easy to separate it from human content.

[*] Corresponding Author.
Email: hope.mcgovern@cl.cam.ac.uk.
[†] Work done while at Grammarly.

These discrepancies, which we call a model's "fingerprint", are consistent enough *across domains* and *within model families* that we find we can treat each LLM as if it were a unique author with a distinct writing style. To do so, we use a well-founded method from the field of Author Identification (AID) for a closed set of authors: using handcrafted n-gram features and training a simple machine learning classifier on those features.

	Best Reported Model		N-gram (Ours)	
Paper	F1	AUROC	F1	AUROC
Deepfake	–	99.0	94.7	94.3
HC3	99.8	–	96.7	99.6
Ghostbuster	99.9	100.0	98.0	98.0
OUTFOX	96.9	–	98.7	98.7

Table 1: **Best reported classifier performances (Deep neural networks) versus a decision-tree model with n-gram features.** Best-reported classifier models are from four recent papers which release labeled datasets for MGT. Our model, a decision-tree classifier, uses a combination of character-, word- and POS-n-gram features and outperforms the best-reported model on the OUTFOX benchmark.

As shown in Table 1, the performance of the simple classifier is surprisingly comparable to more complex neural methods, even in a multi-class setting – successfully distinguishing between, e.g. human-, ChatGPT-, and LLaMA-generated text (Table 2). It also proves robust in cross-domain experiments (Figure 2).

In this paper, we empirically uncover and characterize the fingerprints of individual and families of LLMs through a series of comprehensive analyses, and present a new perspective of LLM-content detection as authorship identification.

Proceedings of the 1st Workshop on GenAI Content Detection (GenAIDetect), pages 85–95
January 19, 2025.

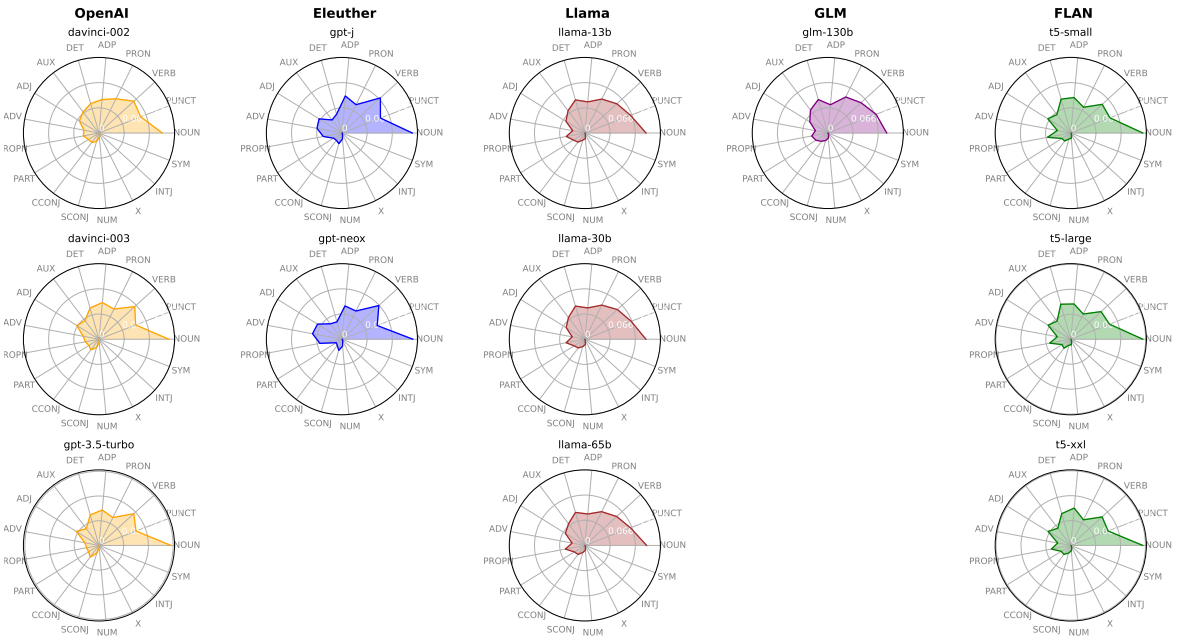


Figure 1: **Visualization of the fingerprints.** We plot frequencies of each part-of-speech (POS) class from the output of several models, sorted by model family. Within each family, the shapes (distributions) look mostly similar regardless of model size. Each radial plot is shown at the same 0% to 20% frequency scale, with POS tags sorted from most to least common among human-written outputs. Jagged/bumpy shapes indicate the fingerprint is more distinct from human distributions. POS is just one component of the full 'fingerprint' we investigate.

2 Methodology

2.1 Fingerprint Features

We use three feature sets: word n-grams ($n \in [2, 4]$), which we expect to be useful in capturing domain-specific vocabulary, but also in capturing function words, which are known to be highly effective for authorship identification; character n-grams ($n \in [3, 5]$), which we intuitively expect to capture subword information broadly aligning with the byte-pair encoding (BPE) tokenization method of many models; and part-of-speech (POS) n-grams ($n \in [2, 4]$), which should capture domain-agnostic information about writing style.

2.2 Classifiers

We use a GradientBoost classifier implemented in the Sklearn library (Pedregosa et al., 2011). The hyperparameters for the classifier were found through grid search, though no extensive hyperparameter sweeps were carried out; this classifier works well out-of-the-box[1]. Initial experiments used a range of ML classifiers, including SVC and logistic regression. These exhibited close or similar performance on our data.

[1]Further hyperparameter tuning could improve classifier performance, but we are primarily interested in exploring why such a simple classifier performs well in the first place.

2.3 Data

We use four publicly available machine-generated text detection datasets for fingerprint analysis as well as training data for supervised sequence classifiers: OUTFOX (Koike et al., 2023), Deepfake-TextDetect (Li et al., 2023), the Human Comparison Corpus (Guo et al., 2023), and Ghostbuster (Verma et al., 2023). We refer to these as 'Outfox','Deepfake', 'HC3', and 'Ghostbuster' in this work, respectively. The Deepfake dataset helpfully provides data splits across 10 text domains and 7 model families. HC3 and Ghostbuster provide data generated by 'gpt-3.5-turbo' across 8 different text domains collectively, while Outfox provides parallel responses to student essay prompts for 'gpt-3.5-turbo', 'text-davinci-003', and 'flan-t5-xxl'. Due to space constraints, complete information on domain coverage and underlying base model(s), may be seen in Table 4.

We only use up to 5,000 training examples of each class (where a class is an individual model or 'human') as we find more data does not improve performance after this point, highlighting a particular advantage of feature-based methods: they are not data-greedy.

3 Experiments

We conduct a series of analyses of LLM fingerprints, finding (1) they are predictive of which model authored a text, (2) consistent across domains, and (3) relatively consistent within model families.

3.1 Characterizing Fingerprints

We visualize fingerprints by looking at the difference of distribution in various linguistic properties. In Figure 1, we report part-of-speech tag distributions of data generated by different models on the same Deepfake data domains[2]. In Appendix A we also include analysis from named entity tags, constituency types, and top-k most frequent tokens. There are, of course, more dimensions of linguistic analysis that could theoretically be applied to uncover model fingerprints.

Distinct patterns emerge when comparing the fingerprint of models *within* the same family compared to models *across* different families. The degree of similarity within families can also vary between families; for example, LLaMA models exhibit a particularly uniform fingerprint across model sizes, while BigScience models (cf. Appendix A) look markedly different.

3.2 Fingerprints for Multi-Class MGT

We take the Ghostbuster and Outfox datasets and perform multi-class classification, considering, e.g. 'ChatGPT' a separate class from 'Flan T5'. Per-class F1 scores and macro-F1 on a held-out test set are reported in Table 2. In both cases, we test a three-way classification and achieve a macro-F1 score greater than 0.91.

The implication of this, then, is that linguistic and morphosyntactic features are effective for distinguishing between texts generated by different LLMs as if they have a unique authorial style, rather than belonging to a generic 'machine-generated' category.

3.3 Robustness to Unseen Data and Models

We intuitively expect that a shift in text domain will impact the efficacy of fingerprints as features. To test this, we take the largest model in each model family of the Deepfake dataset and train a classifier on a set of 9 out of 10 of the text domains available. Specifically, we treat each data source (e.g. 'financeQA', 'cmv', 'reddit eli5', etc.) as a separate text domain. We then compare the F1 score on a held-out test set either of the same training domains, or the held-out 10th domain (downsampled to be the same size), presented in Figure 2. While most models experience a slight dip in performance on OOD data, we find that this difference is not statistically significant.

We conduct a different test in which we select an LLM at random from the Deepfake dataset, train a binary classifier (human vs. machine), and compare the difference of evaluating the trained classifier on either (a) text generated by the same model in a different text domain (OOD) or (b) text from the same domain as the training set, but generated by a different model (OOM). We repeat this experiment $n = 20$ times. As seen in Figure 3, recall for

Dataset	Provenance	F1
Ghostbuster	Human	0.934
	ChatGPT	0.960
	Flan T5	0.927
	Average	0.940
Outfox	Human	0.877
	ChatGPT	0.936
	Claude	0.920
	Average	0.911

Table 2: **F1 scores for each class as the positive class after training under a multiclass classification setting.** Note that even for top models ChatGPT and Claude, our simple n-gram based classifier performs very well (0.936 and 0.920 on the Outfox data). To compare with binary classification results, F1 scores are computed for each class by setting that class to be the 'positive'.

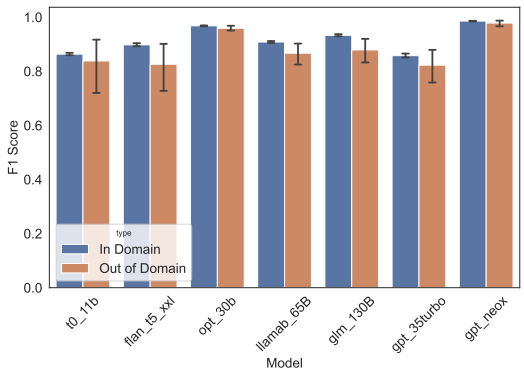


Figure 2: **F1 score of MGT on in-domain versus out-of-domain test sets for the largest model of each model family in the Deepfake benchmark.** We find no statistically significant drop in performance when testing on these 7 models' outputs. 95% confidence intervals are computed through bootstrap sampling at $n = 10,000$.

[2]We choose to report POS results in the main paper as it directly maps to one feature set for our classification experiments, whereas we do not directly use named entity categories, constituency types, or top-k words as features.

the machine class and AUROC drop significantly lower for OOM data compared to OOD data, leading to the interesting insight that *LLMs generate texts across different domains with a consistent, characteristic style that is unique to each model*. In other words, Flan T5 "sounds" like Flan T5 whether it is generating news stories or fan fiction.

We also explicitly test how well a classifier trained on data generated by one model generalizes to (a) other models in the same family and (b) other model families. We find that, on average, the drop in machine recall value (out of 1) from in-domain data to other models in the same family is only 0.01, while the drop to other families is 0.62. We report these results in Table 3.

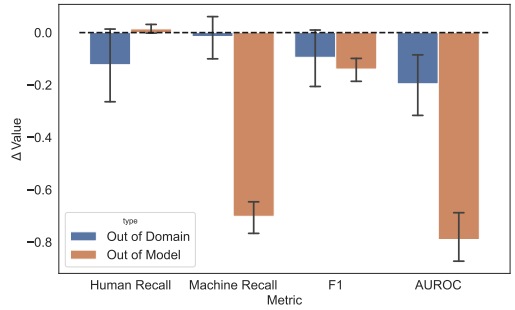


Figure 3: **Average drop in performance when testing on out-of-domain text (blue) versus a text generated by a held-out LLM (brown).** Note that recall of the machine-generated text drops significantly when testing on an unseen model's output, while changing the domain has much less impact.

4 Discussion

4.1 Where Might Fingerprints Come From?

Our work has revealed interesting insights about machine-generated text, namely that LLMs generate in a manner analogous to an individual human author's unique writing style. The origin of these 'fingerprints' is uncertain, but may lie in either the model's training data or model architecture.

It is clear that the model families with the most uniform fingerprint, e.g. LLaMA and Flan T5, are comprised entirely of models trained on the same dataset with the same training method and underlying structure, but with a different number of model parameters. This is also clear in the fingerprint similarity of BigScience's two T0 models contrasted with the one Bloom model, which are trained on different datasets and have different underlying architectures (Encoder-Decoder and GPT3-style, respectively).

A piece of evidence in favor of the influence of training data is that we find that 13B parameter LLaMA chat-tuned model has a *different* fingerprint from its non-chat counterpart, despite having the same architecture (Figure 8). It remains less clear why some families have less uniform fingerprints, and the exact interplay of training data, architecture, and training regime begs further investigation.

4.2 What About Better Models?

Class	F1 Score
GPT-4	0.98
Human	0.98

(a) Binary Classification: GPT-4 vs. Human)

Class	F1 Score
Cohere	0.94
GPT-4	0.96
Human	0.95

(b) Multi-Class Classification: GPT-4 vs. Human vs. Cohere

Figure 4: **F1 Scores for Binary and Multi-Class Classification.**

It might be thought that as language models become larger and generally more capable of producing human-sounding text, their fingerprints will disappear, but we find that our fingerprint-based method performs well even on text produced by more modern models than those contained in the main datasets we test.

We use the data from the COLING 2025 Workshop on Detecting AI-Generated Content[3] to perform both a binary classification of human vs. GPT-4 data, as well as a 3-way classification experiment between human, GPT-4, and Cohere. F1 scores, which may be seen in Figure 4, for all classes in both experiments exceed 0.94.

These scores demonstrate strong performance even with modern models, effectively distinguishing GPT-4 data from human data, as well as differentiating it from other high-capacity models like Cohere. These results suggest that increased model capability alone is insufficient to erase distinctive "fingerprints," highlighting the robustness of our approach in identifying AI-generated content.

5 Related Work

A common approach to machine-generated text detection is to train a supervised binary classifier on

[3]Specifically, we use the data of Subtask a of Task 1, available here https://huggingface.co/datasets/Jinyan1/COLING_2025_MGT_en

labeled data (Guo et al., 2023; Koike et al., 2023; Li et al., 2023). Li et al. (2023) proposed a variety of classification testbeds, finding that pre-trained language models perform the best. While n-gram frequencies have often been used for author identification, only a few recent works examine hand-crafted features or stylometrics in machine-generated text detection (Zaitsu and Jin, 2023). One example is Gehrmann et al. (2019): a unique system that uses the top-k words to highlight text spans to visually aid humans in the task of spotting AI-written text themselves.

Petukhova et al. (2024) finds a combination of fine-tuned neural features and hand-crafted linguistic features effective for MGT on the M4 dataset as part of the SemEval2024 task on machine-generated text detection (Wang et al., 2024).

Li et al. (2023) analyze their corpus DeepfakeTextDetect across linguistic feature axes, but report differences across POS-tag distributions between human and machine data when considering all models and domains in aggregate as insignificant; however, they do find these distributions begin to diverge when considering a subset of models or domains. We demonstrate that these differences extend to every publicly available machine text detection dataset, prove largely consistent within model families, and are very powerful features for training a robust machine-generated text detection classifier.

While linguistic-feature-based approaches have shown promise, other state-of-the-art (SOTA) methods, such as Mitchell et al. (2023); Bao et al. (2024); Tian and Cui (2023), adopt probabilistic and statistical modeling approaches to detect machine-generated text in a training-free setting. We focus purely on manually extracted linguistic features rather than probability curvatures.

6 Conclusion

We demonstrate that in four popular datasets for machine-generated text detection, n-gram features are highly effective for MGT. We uncover that LLMs have unique writing styles that can be captured in lexical and syntactic features, which we characterize as "fingerprints", and show may be effectively harnassed for text-detection in a variety of settings.

Limitations

- **Text length:** we examine outputs of approximately 300-500 words in length. Shorter texts may be difficult to fingerprint or may not provide enough signal.
- **Model choice limitations**: We constrain ourselves to the data and models released as part of text detection corpora, which means that there may be some very good models we simply did not have the data to test at this time.
- **Reflection on real-world use-case.** Analyzing fingerprints in research benchmark datasets is most likely *not* reflective of the true difficulty of deepfake text detection in the wild. For one thing, people don't tend to use LLMs for writing entire articles/essays, etc. A more likely scenario for, e.g. academic plagiarism, is starting from an LLM generated paragraph and making sentence-level rewrites. As this is analagous to a paraphrase attack like DIPPER (Krishna et al., 2023), we expect that it would degrade our classifiers' performance.

Ethics Statement

This research indicates that detecting machine-generated text is easy. However, we want to stress that this does *not* necessarily mean machine-detection is a high-confidence task. Using a single model prediction about one single written text to determine whether or not it was human-written should be evaluated on a different basis than average accuracy, given the potential harms of false positives or false negatives. For example, teachers may wish to use tools to determine if students have cheated on exams or homework using LLMs. We discourage teachers from trusting predictions by any classifier until more investigation is done into the confidence models have for any individual text.

References

Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, Linyi Yang, and Yue Zhang. 2024. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature. *Preprint*, arXiv:2310.05130.

Sofia Barnett. 2023. ChatGPT Is Making Universities Rethink Plagiarism. *Wired*. Section: tags.

Elizabeth Clark, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A. Smith.

2021. All That's 'Human' Is Not Gold: Evaluating Human Evaluation of Generated Text. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7282–7296, Online. Association for Computational Linguistics.

Alex Franklin, Maggie, Meg Benner, Natalie Rambis, Perpetual Baffour, Ryan Holbrook, Scott Crossley, and ulrichboser. 2022. Feedback prize - predicting effective arguments.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander M. Rush. 2019. Gltr: Statistical detection and visualization of generated text. In *Annual Meeting of the 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. Publisher: arXiv Version Number: 1.

Ryuto Koike, Masahiro Kaneko, and Naoaki Okazaki. 2023. Outfox: Llm-generated essay detection through in-context learning with adversarially generated examples. *ArXiv*, abs/2307.11729.

Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. 2023. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. *Preprint*, arXiv:2303.13408.

Yafu Li, Qintong Li, Leyang Cui, Wei Bi, Longyue Wang, Linyi Yang, Shuming Shi, and Yue Zhang. 2023. Deepfake Text Detection in the Wild. Publisher: arXiv Version Number: 1.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. DetectGPT: Zero-Shot Machine-Generated Text Detection using Probability Curvature. arXiv. Version Number: 2.

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

Kseniia Petukhova, Roman Kazakov, and Ekaterina Kochmar. 2024. Petkaz at semeval-2024 task 8: Can linguistics capture the specifics of llm-generated text?

Edward Tian and Alexander Cui. 2023. Gptzero: Towards detection of ai-generated text using zero-shot and supervised methods.

Vivek Kumar Verma, Eve Fleisig, Nicholas Tomlin, and Dan Klein. 2023. Ghostbuster: Detecting text ghostwritten by large language models. *ArXiv*, abs/2305.15047.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, Thomas Arnold, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024. SemEval-2024 task 8: Multigenerator, multidomain, and multilingual black-box machine-generated text detection. In *Proceedings of the 18th International Workshop on Semantic Evaluation*, SemEval 2024, Mexico City, Mexico.

Chris Westfall. 2023. Educators Battle Plagiarism As 89% Of Students Admit To Using OpenAI's ChatGPT For Homework. Section: Careers.

Wataru Zaitsu and Mingzhe Jin. 2023. Distinguishing ChatGPT(-3.5, -4)-generated and human-written papers through Japanese stylometric analysis. *PLOS ONE*, 18(8):e0288453.

A Fingerprint Characterization

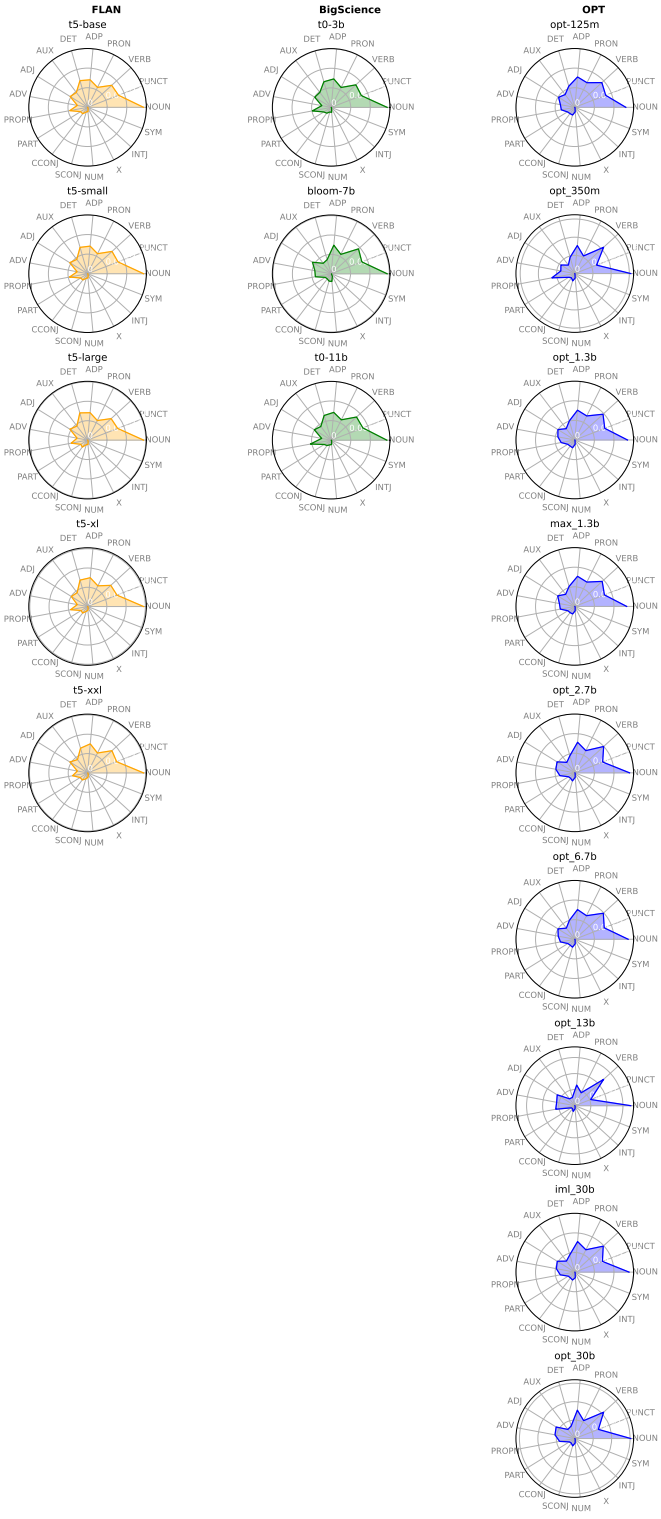
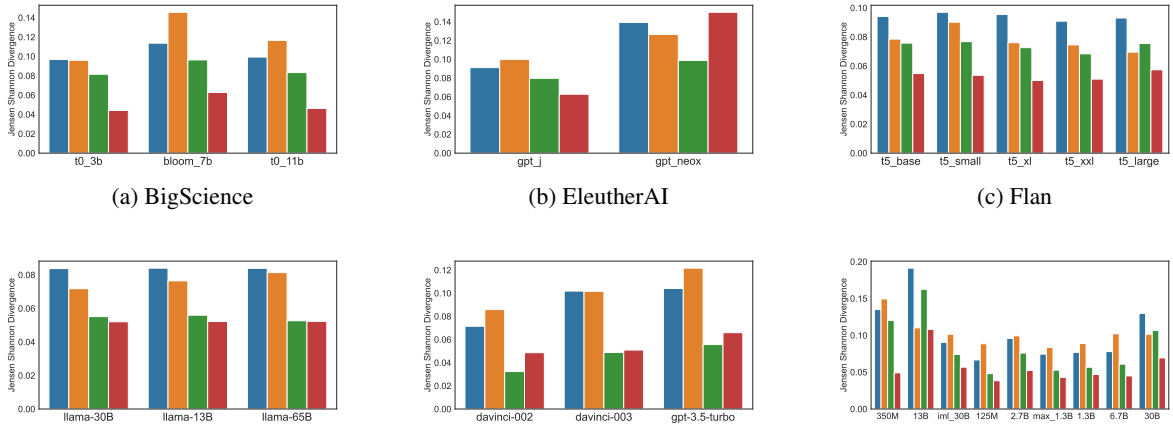


Figure 5: **Additional visualizations of fingerprints.** Note that the POS tag distributions of OPT models are less similar than we observe within other model families. Further investigations could examine what causes these differences, since model size seems to not play a factor in FLAN models.



(a) BigScience (b) EleutherAI (c) Flan

(d) LLaMA (e) OpenAI (f) OPT

Figure 6: **Fingerprint characterization of Deepfake data by model and family.** We report the Jensen-Shannon Divergence of human vs. model for each model in each model family in the Deepfake data across four categories. **Columns from left to right: constituency type, named entity tag, POS tag, top-k word frequency**. We omit the GLM family in this visualization as there is only one model (130B) available. Like in Figure 1, some model families exhibit remarkably consistent fingerprints within families, e.g. LLaMa, Flan, and OpenAI. OPT and EleutherAI in particular have less distinguishable fingerprints within family.

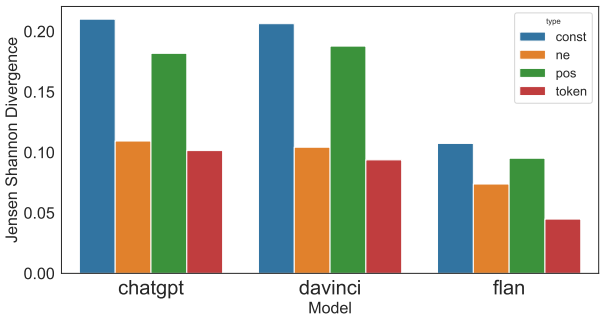


Figure 7: **Fingerprint characterization of Outfox data by model**. **Columns from left to right: constituency type, named entity tag, POS tag, top-k word frequency**. We note again that ChatGPT and davinci, being in the same OpenAI model family, have very similar fingerprints, whereas Flan's fingerprint differs substantially. Note that this fingerprint does look different than the Deepfake davinci's fingerprint, showing us that there is some domain dependence to fingerprints, while underscoring the point that regardless of domain, individual models of the same family do produce similar-sounding texts.

Experiment	Average drop in performance			
	HRec	MRec	F1	AUC
Same Family | Different Domain	−0.03	−0.01	−0.02	0.00
Different Family | Same Domain	0.00	−0.62	−0.21	−0.44

Table 3: **Models exhibit individual writing styles which are more similar across domains than across model families.** We report the average drop in performance of a GradientBoost from a binary classifier trained on Deepfake data. In 7 independent trials, we train a classifier on a randomly selected model and compare its performance on the in-domain test set to: (1) data from a model in the same family but in a held-out domain, and (2) data from a model in a different family but same domains present in the train set (this is made possible by the fact that Deepfake is multi-parallel). Performance drop is low over data from a model in the same family, and high over data from a model in a different family. The human recall value is small but not 0 as the human data is shuffled and downsampled, so the exact same set of prompts is not seen in every trial.

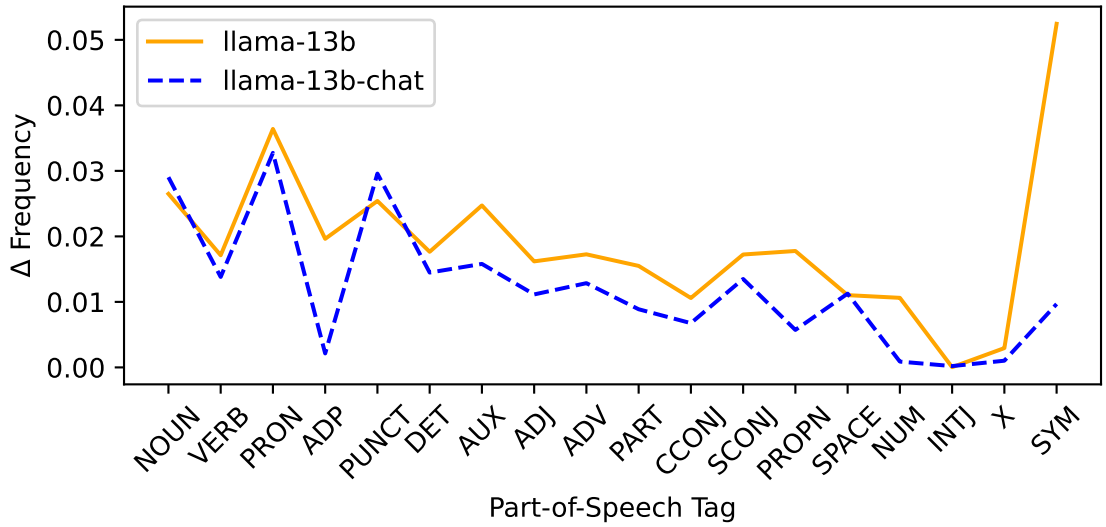


Figure 8: **Absolute difference in POS tag frequencies as compared with human text**. Chat models are slightly more similar to the frequency profile of humans, but are easier to detect than base models. This demonstrates that fingerprints "closer" to human distributions in POS tags does *not* indicate it is less detectable. Further, fine-tuning models for chat clearly alters their fingerprint despite no change in model architecture.

B Implementation Details

B.1 GradientBoost

Parameters: learning rate of 0.2, number of estimators 90, max depth of 8, max features 'sqrt', sum-sample ratio 0.8, random state 10, minimum samples leaf 30 and minimum samples to split 400, these hyperparameters were optimized using Sklearn's gridsearch function. Features: char n-grams:(2,4), word n-grams:(3,5), pos n-grams:(3,5). Maximum 2000 features for each feature set.

C Dataset Information

C.1 Outfox

Outfox is a parallel human-machine dataset built on the Kaggle Feedback Prize dataset (Franklin et al., 2022) and contains approximately 15,000 essay problem statements and human-written essays, ranging in provenance from 6th to 12th grade native-speaking students in the United States. For each problem statement, there is also an essay generated by each of three LLMs: ChatGPT (gpt-3.5-turbo-0613), GPT-3.5 (text-davinci-003), and Flan (FLAN-T5-XXL). Each example contain an instruction prompt ("Given the following problem statement, please write an essay in 320 words with a clear opinion."), a

Dataset	Base Model/Family	Domain	Human	Machine
Domain-Specific	gpt-j-6b	cmv	509	636
		eli5	952	863
		hswag	1000	868
		roct	999	833
		sci_gen	950	529
		squad	686	718
		tldr	772	588
		xsum	997	913
		yelp	984	856
		wp	940	784
		Total	8789	7588
Mixed Model Set	OpenAI GPT	mixed	67k	67k
	Meta Llama	mixed	37k	37k
	GLM-130B	mixed	9k	9k
	Google FLAN-T5	mixed	47k	47k
	Facebook OPT	mixed	80k	80k
	BigScience	mixed	27k	27k
	EleutherAI	mixed	14k	14k
		Total	282k	282k
Ghostbuster	gpt-3.5-turbo	Reuters	500	500
		essay	1000	1000
		wp	500	500
		Total	2000	2000
HC3	gpt-3.5-turbo	eli5	17.1k	17.1k
		open_qa	1.19k	1.19k
		wiki_csai	842	842
		medicine	1.25k	1.25k
		finance	3.93k	3.93k
		Total	24.3k	24.3k
OUTFOX	gpt-3.5-turbo	essay	15k	15k
	text-davinci-003	essay	15k	15k
	flan_t5_xxl	essay	15k	15k
		Total	46k	46k

Table 4: Dataset statistics (number of documents) for publicly available machine-generated text detection datasets.

problem statement ("Explain the benefits of participating in extracurricular activities and how they can help students succeed in both school and life. Use personal experiences and examples to support your argument."), the text of the essay, and a binary label for human or machine authorship.

While we conduct fingerprint analysis on the whole dataset, we use only the human-written subset of the Outfox data as a training corpus for our fine-tuning setup; given an instruction prompt and problem statement, we fine-tune our LLMs of interest to produce text which minimises cross-entropy loss when compared with the original human-written response to the same problem statement. We withhold a test-set of human-written examples from training to be used for evaluation.

C.2 Ghostbuster

Verma et al. (2023) provide three new datasets for evaluating AI-generated text detection in creative writing, news, and student essays. Using prompts scraped from the subreddit `r/WritingPrompts`, the Reuters 50-50 authorship identification dataset, and student essays from the online source IvyPanda, they obtained ChatGPT- and Claude-generated responses and made efforts to maintain consistency in length with human-authored content in each domain.

C.3 HC3

We also analyze data from (Guo et al., 2023), which includes questions from publicly available datasets and wiki sources with human- and ChatGPT-generated responses based on instructions and additional

context. The resulting corpus comprises 24,322 English and 12,853 Chinese questions, of which we only use the English split.

C.4 Deepfake

The Deepfake corpus is a comprehensive dataset designed for benchmarking machine-generated content detection in real-world scenarios (Li et al., 2023). It contains approximately 9,000 human examples across 10 text domains, each paired with machine outputs from 27 models (e.g. GPT-3.5-turbo, text-davinci-002) from 7 different model families (e.g. OpenAI), producing several testbeds designed for examining a detector's sensitivity to model provenance and text domain. Each example contains the text, binary label denoting human or machine, and the source information – which domain, model, and prompting method were used.

Training Data. We primarily use the Deepfake and Outfox data for training classifiers to analyze different aspects of the LLM fingerprints. They are both conveniently multi-parallel: they contain N model responses for each human text sample in the dataset. This has the benefit of removing some uncertainty from our classifier results. Performance on the human class is often identical across trials, as the human data is often identical. This allows a controlled test of how our classifier deals with the machine text samples. Additionally, the different testbeds provided in Deepfake provide convenient, parallel domain and model (/model family) data splits. Specifically, we use the mixed model sets and model-specific, domain-specific testbeds from Deepfake.