Authorial Language Models For AI Authorship Verification

Notebook for the PAN Lab at CLEF 2024

Weihang Huang[1,†], Jack Grieve[1,†]

[1]Department of English Language and Linguistics, University of Birmingham, Edgbaston, Birmingham, B152TT, United Kingdom

Abstract

In this paper, we introduce the use of Authorial Language Models (ALMs) for AI Authorship Verification (AIAV). Given two texts, where one is written by a human and one is written by a machine, AIAV is the task of determining which text was written by the machine (or alternatively by the human). Our approach to resolving this task involves using a support vector machine to predict which text is written by a machine based on perplexity scores for a set of language models that were each fine-tuned on texts generated by a set of language models. We submitted our method as a docker-contained software for independent evaluation on the main testing dataset, and its variants that are obfuscated against detection. On the main dataset, we have been informed that our method has achieved a score of approximately 0.979 on all proposed evaluation measures, including ROC-AUC Brier C@1 F1 and F0.5, beating all baseline methods. And on the variants of main datasets, we achieve a median score of 0.935 which also beats all baselines. We attributes the success of ALMs in this context to the power of using many fine-tuned authorial language models, which we believe improves the resilience of our approach by maximising the amount of potentially discriminating information drawn from the underlying textual data.

Keywords

LLM Detection, Large Language Model, Perplexity, Authorship Verification

1. Introduction

Authorship verification can be defined as the task of predicting whether two input texts are written by the same author[1]. For example, previous PAN labs[2, 3, 4] have led to the development and benchmarking of many excellent methods for human authorship verification. The recent development of Large Language Models (LLMs), however, has introduced new challenges to the field. Since the invention of transformers in 2017[5] and the release of GPT-2 in 2019[6], a number of LLMs are now capable of producing texts with human levels of fluency – even when generated via zero- or few-shot in-context learning (i.e. prompting). In turn, this rapid advance in LLM technology has led to a demand for tools capable of automatically detecting LLM-written texts, extending the problem of authorship identification to the analysis of machine-generated writing for the first time [7, 8, 9, 10, 8, 10]. Notably, although these studies define the task of LLM detection in somewhat different ways, at a basic level, all involve distinguishing machine-written texts from human-written texts.

Building on research in this area, PAN@CLEF2024 released the shared task of Voight-Kampff Generative AI Authorship Verification (AIAV), where the LLM detection problem is reframed as a verification task: given a pair of texts, where one is written by a human and one is written by a machine, the goal is to select the text written by the human (or alternatively the machine) [4, 11]. To resolve this tasks, we extended our authorial language models (ALMs) paradigm for authorship attribution [12] to AIAV. When evaluated on the TIRA dataset [13], our method outperforms all baselines methods with a mean benchmarking score of 0.979.

2. Background

Large Language Models (LLMs) are models, consisting of millions to billions of parameters, that predict the probability distribution of tokens given their observed context. Most LLMs are based on the

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France

wxh207@student.bham.ac.uk (W. Huang); j.grieve@bham.ac.uk (J. Grieve)
0000-0002-0820-4476 (W. Huang); 0000-0003-3630-7349 (J. Grieve)

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

transformer deep learning architecture, which was introduced in 2017 [5]. While LLMs consisting of millions of parameters (e.g. GPT-2) have been capable of producing texts with human-level fluency for years, the more recent development of LLMs with billions of parameters (e.g. GPT-3.5, GPT-4, Llama) has made it possible to generate text via prompting. This now allows almost anyone to easily and quickly generate machine-written texts of very high quality. Although this type of automated writing has great potential value for society, there is also considerable concern about its misuse [7, 8, 9, 10, 8, 10]. While LLM security is a broad topic that requires efforts from across academia and industry to mitigate the risk of LLM misuse and abuse, LLM detection is clearly an indispensable parts of this endeavor.

LLM Detection is a family of tasks within authorship analysis that involve identifying machine-written texts and distinguishing machine writing from human writing. From this broad definition, several more specific types of LLM detection tasks can be identified [11]. Arguably, at the most basic level, the problem is to distinguish between pairs of texts, where one text is written by a human and one text is written by a machine [11]. This task, which is the focus of the PAN@CLEF2024 shared task, is referred to as AIAV. Several solutions to AIAV have been proposed, including PPMd Compression-based Cosine, Authorship Unmasking, Binoculars, DetectLLM LRR and NPR, DetectGPT and Fast-DetectGPT, which act as the baselines for this shared task[11].

Perplexity (PPL) and perplexity-related measures have been at the core of many attempts to automate LLM detection. Perplexity is defined as the exponentiated mean log-likelihood of a text over a LLM, as described in the following formula.

$$PPL\left(M,X\right)=exp\left\{-\frac{1}{t}\sum_{i}^{t}log\left(p_{M}\left(x_i|x_{<i}\right)\right)\right\}$$

where $X=\{x_1,x_2,...,x_t\}$ is the sequence of tokens (e.g., the text), t is the length of the sequence (i.e. number of tokens), M is the LLM, and $p_M\left(x_i|x_{<i}\right)$ is the predicted probability of the i^{th} token given an LLM and the preceding tokens in the sequence. Perplexity measures the predictability of a text over the LLM, and is commonly used in LLM training as a potential loss function and evaluation metric. The higher perplexity is, the less predictable a text is to the LLM.

Perplexity for LLM detection is a common approach because LLM-generated texts are generally assumed to be associated with a lower perplexity than human-written texts for any given LLM. This approach is especially common in hybrid LLM-detection solutions that are designed to assist humans distinguishing human-written texts from LLM-generated ones [8, 14]. LLM detection via perplexity, however, also has clear limitations. From a technical standpoint, it relies heavily on using LLMs for the calculation of perplexity. More fundamentally, it is also entirely possible for human texts to be associated with relatively low perplexity scores for generic LLMs. In general, such approaches therefore appear to be overly simplistic. To mitigate the risks of prediction failure, especially avoiding false positive, where texts written by real human authors are flagged as being machine-generated, researchers have therefore expanded on this approach, for example, by incorporating a pair of pertained LLMs rather than a single LLM into their LLM detection systems[15].

Authorial Language Models (ALMs) is a paradigm for authorship analysis that relies on training a set of fine-tuned authorial models based on the available writing samples for each candidate author[12]. Unlike most previous LLM-based approaches to authorship analysis that use only one LLM[16, 17], ALMs involves using multiple LLMs, one for each candidate author, to better capture authorial variation in token predictability. This makes ALMs more resilient to exceptional or extreme cases because this approach does not relying on a single LLM, while allowing for greater amounts of information to be extracted from the underlying textual data, as the LLMs are fine-tuned on a candidate-by-candidate basis. Furthermore, ALMs is also more interpretable as it can provide token-level predictability metrics for the questioned document for each candidate author. Because of these advantages, we have found that ALMs outperforms all other state-of-the-art methods (N-grams NN, BERT, and PPM) for human authorship attribution on the Blogs50 dataset, while nearly matching the performance of N-grams NN, which achieves the best results, on the CCAT50 datasets[12]. For this shared task, we have therefore modified ALMs for AIAV, as we detail in the next section.

3. Datasets

The PAN@CLEF2024 AIAV shared tasks involves two groups of datasets: the bootstrap group and the testing group.

The bootstrap group was open for method development. In the bootstrap group, one dataset contains 1087 texts that were generated by 13 widely-used LLMs ranging from Llama[18] to GPT-4[19], together with 1087 texts that were authored by humans. Regardless of the author, texts in bootstrap dataset are full or trimmed news articles. In this study, we used the bootstrap datasets for the fine-tuning of authorial language models and the training of the support vector machine classifier. We then developed our method and submitted it to the tira.io[13],

Meanwhile, the testing group is retained by PAN2024 organizers and was not made available to participants in the shared task. Rather, this dataset was used to independently test the systems submitted to tira.io for assessment. Specifically, the PAN2024 organizers tested our system on the testing group of datasets, which includes one main dataset, plus nine variants of datasets that are obfuscated against AI verification methods. For the details of testing group datasets, PAN2024 organizers plan to release the basic facts and compilation details in the overview notebook[11].

4. System Overview

ALMs for AIAV is a version of ALMs that is tailored to the needs of AIAV shared task. ALMs for AIAV is based on the idea of using perplexity for LLM detection, where human-authored texts are assumed to have substantially higher perplexity compared to LLM-authored ones. However, this assumption has exceptions if we only consider perplexity from a single LLM: there are human-authored texts with relative low perplexity, and LLM-authored texts with relatively high perplexity, both of which undermine LLM detection using this approach. Hence, during the development of our method, we hypothesized that these exceptions could result from a lack of any attempt to represent in the styles of *different* LLMs, which we believe could affect LLM detection in two ways.

On the one hand, confounding variables in training corpus, such as genres, registers, and topics, can possibly distort perplexity: for example, a human-written texts in a register that is over-represented in the training corpus would tend to be associated with a relatively low perplexity, whereas an LLM-authored text written in a register that is underrepresented in the training corpus would tend to be associated with a relatively high perplexity. On the other hand, the differences in language modeling and text generating pipelines can also lead exceptional perplexity values: for example, a LLMs that uses a distinctively unique pipeline would potentially generate texts that are more unexpected to other existing LLMs and hence be associated with relatively high perplexity.

Although these issues cannot be completely eliminated, we believe these issues can be mitigated by using not one but many LLMs. By fine-tuning pre-trained LLMs to correspond to each of the potential LLMs in the detection task, we can build perplexity array that take into account the styles of different LLMs. Meanwhile, based on this perplexity array, perplexity values for each of the LLMs can be compared against one another, which further mitigates the risk from under-representing LLM styles.

Like ALMs for human authorship attribution, the first step for ALMs for AIAV is the fine-tuning of a series of pre-trained LLMs that correspond to each of the potential LLM "authors". These fine-tuned authorial language models are then used to extract a perplexity array for each pair of question texts. The perplexity arrays are then used as feature vectors in a pre-trained Support Vector Machine (SVM) classifier to decide which of the two texts is most likely written by a human. Finally, the prediction result is outputted as the is_human score, as requested by the shared tasks[11]. is_human ranges between 0 and 1, where 0 means the first text is considered human-written, and 1 means the second text is considered human-written. The workflow of ALMs for AIAV is described as flowchart in Figure 1, Figure 2, and Figure 3. The details of each step are described in the following subsections.

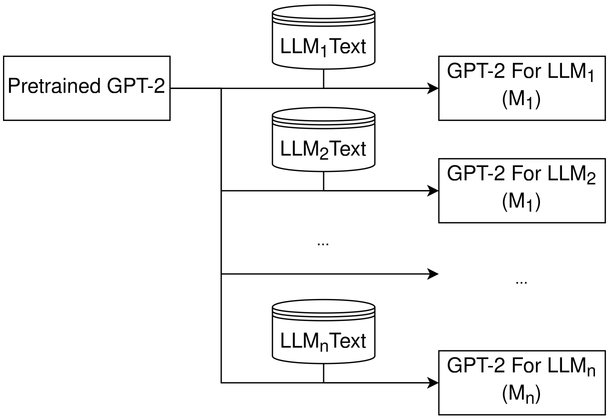
Pretrained GPT-2
LLM$_1$Text
LLM$_2$Text
...
LLM$_n$Text
GPT-2 For LLM$_1$ (M$_1$)
GPT-2 For LLM$_2$ (M$_1$)
...
GPT-2 For LLM$_n$ (M$_n$)

Figure 1: Step 1: Fine-tuning Authorial Language Models

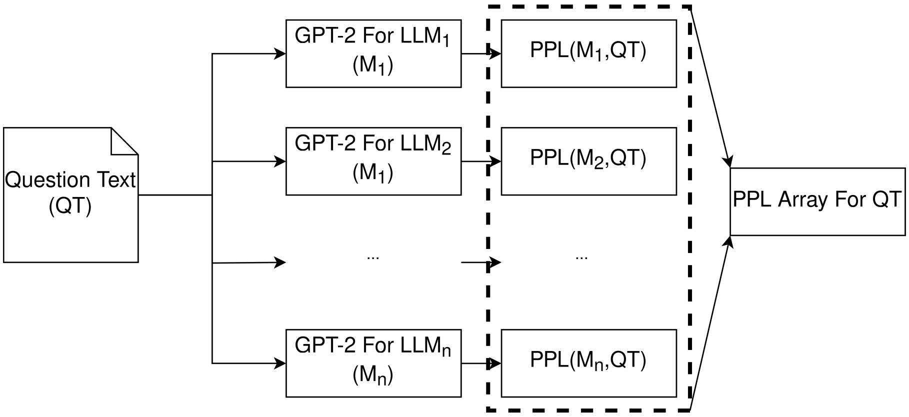

Figure 2: Step 2: Perplexity Array Extraction

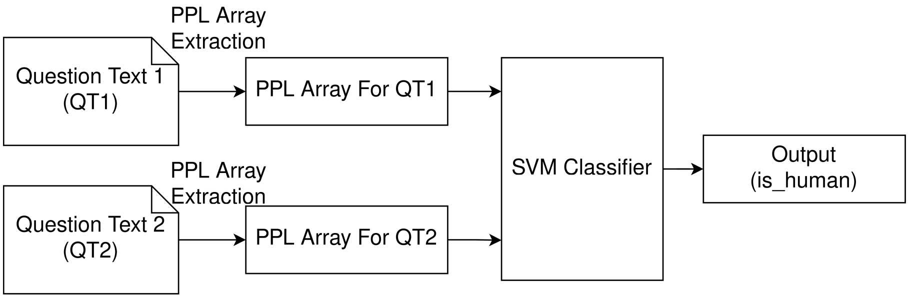

Figure 3: Step 3: Authorship Prediction via Support Vector Machine

4.1. Fine-tuning Authorial Language Models

The first step of the ALMs for AIAV is the fine-tuning of pre-trained LLMs on the texts from each candidate author. However, in the AIAV shared task, candidates are grouped by whether it is human(e.g LLMs group and human group). Though the number of authors in human group is unclear, the number of authors in the LLM groups is specified. Therefore, we can take the LLMs listed in the bootstrap dataset[20] as potential "authors" for the fine-tuning of the authorial language models. We take 80% for each of the LLM datasets as the training data for LLM fine-tuning, and we retain the remaining 20% for use in further steps. As most of these potential models are causal language models, we choose GPT-2 base, a canonical causal language model as the pre-trained model for fine-tuning. We then fine-tune 13 GPT-2 models on the texts from 13 potential LLM "authors". For each case, we fine-tune each LLM for 20 epochs with a weight decay of 0.01, an initial learning rate of 0.00002, and a gradient accumulation step of 16.

4.2. Perplexity Array Extraction

Although perplexity is defined as the exponentiated mean log-likelihood of all tokens in a text, for efficiency purposes, we calculate perplexity based on cross entropy using the formula below:

$$PPL\left(M, X\right) = exp\left\{CrossEntropy\left(Logits, X\right)\right\}$$

Given an input text Q, and a fine-tuned authorial GPT-2 model M, we first pass Q to the GPT-2 BPE Tokenizer to extract a token sequence X. X is then passed to M for language modeling, whose output is $Logits$. Here $Logits$ reflects the predicted probabilities of all tokens in X, where X represents the ground truth. Therefore, in the next step, we measure the predictability of all tokens in X by comparing the predicted $Logits$ and the ground truth X via cross entropy, which we calculate using $torch.nn.CrossEntropyLoss$ from PyTorch. Finally, we obtain the perplexity of Q under M by exponentiated cross entropy.

For each input text Q, we calculate its perplexity under each of the 13 fine-tuned, authorial language models. We store these perplexity values in an 13*1 array, which we flag as a perplexity array for input text Q. The perplexity array is then used in the next step as a feature array to make a prediction on each questioned text pair.

4.3. Authorship Prediction via Support Vector Machine

Given an input question text pair Q_1 and Q_2, we first extract their perplexity arrays from the 13 authorial language models. We then move to authorship prediction based on the two perplexity arrays. For this stage, we trained an SVM using a reconstituted dataset composed of paired perplexity arrays from the human data in bootstrap dataset and the remaining 20% of LLM-generated texts that we retained during ALMs fine-tuning. In this dataset, we paired perplexity arrays following the description of the shared tasks, where, for each pair of texts, we guarantee that one text is human authored and the other text is LLM-generated. We trained the SVM classifier with a radial basis function kernel, a regularization parameter of 1.0, and a tolerance for stopping criterion of 0.001. We did not impose hard training steps or epochs for the SVM classifier.

5. Results

We submitted our ALMs for AIAV as docker contained software for the benchmarking on the tira.io group of testing datasets[13]. During testing, our method was labeled as "greasy-chest". Table 1 shows the results, initially pre-filled with the official baselines provided by the PAN organizers and summary statistics for all submissions to the shared task (i.e., the maximum, median, minimum, and 95-th, 75-th, and 25-th percentiles over all submissions to the task). We find that our method beat all existing baselines for all evaluation measures and performs in the top 25% for all submissions to this shared task.

In addition, Table 2 shows the summarized results averaged (arithmetic mean) over all 10 variants of the test dataset. Each dataset variant applies one potential technique to measure the robustness of the AIAV systems, including but not limited to switching the text encoding, translating the text, switching the domain, and manual obfuscation by humans. Our method achieves a median score of 0.935 over the 9 variants, which also surpasses all existing baselines and is among the top 25% of all submissions to this shared task. However, we also notice that our method has a relative low minimum score over 9 variants, suggesting that further investigations are needed for the most challenging dataset variant. Our submission(as team "jaha") scores 17th out of 30 on the leaderboard with a ranking score of 0.683.

Table 1
Overview of the accuracy in detecting if a text is written by an human in task 4 on PAN 2024 (Voight-Kampff Generative AI Authorship Verification). We report ROC-AUC, Brier, C@1, F_1, $F_{0.5u}$ and their mean.

Approach	ROC-AUC	Brier	C@1	F_1	$F_{0.5u}$	Mean
greasy-chest	0.979	0.979	0.979	0.979	0.979	0.979
Baseline Binoculars	0.972	0.957	0.966	0.964	0.965	0.965
Baseline Fast-DetectGPT (Mistral)	0.876	0.8	0.886	0.883	0.883	0.866
Baseline PPMd	0.795	0.798	0.754	0.753	0.749	0.77
Baseline Unmasking	0.697	0.774	0.691	0.658	0.666	0.697
Baseline Fast-DetectGPT	0.668	0.776	0.695	0.69	0.691	0.704
95-th quantile	0.994	0.987	0.989	0.989	0.989	0.990
75-th quantile	0.969	0.925	0.950	0.933	0.939	0.941
Median	0.909	0.890	0.887	0.871	0.867	0.889
25-th quantile	0.701	0.768	0.683	0.657	0.670	0.689
Min	0.131	0.265	0.005	0.006	0.007	0.224

Table 2
Overview of the mean accuracy over 9 variants of the test set. We report the minimum, median, the maximum, the 25-th, and the 75-th quantile, of the mean per the 9 datasets.

Approach	Minimum	25-th Quantile	Median	75-th Quantile	Max
greasy-chest	0.295	0.731	0.935	0.979	0.995
Baseline Binoculars	0.342	0.818	0.844	0.965	0.996
Baseline Fast-DetectGPT (Mistral)	0.095	0.793	0.842	0.931	0.958
Baseline PPMd	0.270	0.546	0.750	0.770	0.863
Baseline Unmasking	0.250	0.662	0.696	0.697	0.762
Baseline Fast-DetectGPT	0.159	0.579	0.704	0.719	0.982
95-th quantile	0.863	0.971	0.978	0.990	1.000
75-th quantile	0.758	0.865	0.933	0.959	0.991
Median	0.605	0.645	0.875	0.889	0.936
25-th quantile	0.353	0.496	0.658	0.675	0.711
Min	0.015	0.038	0.231	0.244	0.252

6. Conclusion

In this paper, we have introduced ALMs For AIAV, a generative AI verification method that utilizes fine-tuned authorial language models and a support vector machine classifier to predict which text is written by human in a human- and machine-written text pair. We found that our method has a score of 0.979 in ROC-AUC Brier C@1 F1 and F0.5, which is better than all baseline methods. We attribute the excellent performance of ALMs for AIAV's to the ALMs paradigm, which uses many fine-tuned authorial language models, providing greater flexibility and resilience than is possible if only one LLM is used, as has often been the case in previous methods for authorship identification.

Future research may focus on the improvement of authorial prediction methods and use a regressor instead of classifier as proposed in this paper. In addition, it is also worthwhile to experiment with in-context learning (ICL) of LLMs with billions of parameters to see whether ICL could be an effective replacement for the fine-tuning approach we have taken, since ICL would potentially enable a few-shot implementation of our method.

Acknowledgments

We would like to thanks PAN2024 Organizers for their efforts, and Maik Fröbe for the information provided during the submission and evaluation of our software. We would also like to thank Akira Murakami for his support in the development of ALMs.

This research is supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via the HIATUS Program contract #2022-22072200006. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

[1] M. Koppel, J. Schler, S. Argamon, Y. Winter, The "Fundamental Problem" of Authorship Attribution, English Studies 93 (2012) 284–291. URL: http://www.tandfonline.com/doi/abs/10.1080/0013838X.2012.668794. doi:10.1080/0013838X.2012.668794.

[2] J. Bevendorff, B. Chulvi, E. Fersini, A. Heini, M. Kestemont, K. Kredens, M. Mayerl, R. Ortega-Bueno, P. Pęzik, M. Potthast, F. Rangel, P. Rosso, E. Stamatatos, B. Stein, M. Wiegmann, M. Wolska, E. Zangerle, Overview of pan 2022: Authorship verification, profiling irony and stereotype spreaders, and style change detection, in: A. Barrón-Cedeño, G. Da San Martino, M. Degli Esposti, F. Sebastiani, C. Macdonald, G. Pasi, A. Hanbury, M. Potthast, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction, Springer International Publishing, Cham, 2022, pp. 382–394.

[3] J. Bevendorff, M. Chinea-Ríos, M. Franco-Salvador, A. Heini, E. Körner, K. Kredens, M. Mayerl, P. Pundefinedzik, M. Potthast, F. Rangel, P. Rosso, E. Stamatatos, B. Stein, M. Wiegmann, M. Wolska, E. Zangerle, Overview of pan 2023: Authorship verification, multi-author writing style analysis, profiling cryptocurrency influencers, and trigger detection: Extended abstract, in: Advances in Information Retrieval: 45th European Conference on Information Retrieval, ECIR 2023, Dublin, Ireland, April 2–6, 2023, Proceedings, Part III, Springer-Verlag, Berlin, Heidelberg, 2023, p. 518–526. URL: https://doi.org/10.1007/978-3-031-28241-6_60. doi:10.1007/978-3-031-28241-6_60.

[4] A. A. Ayele, N. Babakov, J. Bevendorff, X. B. Casals, B. Chulvi, D. Dementieva, A. Elnagar, D. Freitag, M. Fröbe, D. Korenčić, M. Mayerl, D. Moskovskiy, A. Mukherjee, A. Panchenko, M. Potthast, F. Rangel, N. Rizwan, P. Rosso, F. Schneider, A. Smirnova, E. Stamatatos, E. Stakovskii, B. Stein, M. Taulé, D. Ustalov, X. Wang, M. Wiegmann, S. M. Yimam, E. Zangerle, Overview of PAN 2024: Multi-Author Writing Style Analysis, Multilingual Text Detoxification, Oppositional Thinking Analysis, and Generative AI Authorship Verification, in: L. Goeuriot, P. Mulhem, G. Quénot, D. Schwab, L. Soulier, G. M. D. Nunzio, P. Galuščáková, A. G. S. de Herrera, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fifteenth International Conference of the CLEF Association (CLEF 2024), Lecture Notes in Computer Science, Springer, Berlin Heidelberg New York, 2024.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention Is All You Need, 2017. URL: http://arxiv.org/abs/1706.03762, arXiv:1706.03762 [cs].

[6] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, Language Models are Unsupervised Multitask Learners (????) 24.

[7] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, et al., On the opportunities and risks of foundation models (2022). URL: http://arxiv.org/abs/2108.07258, arXiv:2108.07258 [cs].

[8] S. Gehrmann, H. Strobelt, A. Rush, Gltr: Statistical detection and visualization of generated text, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Association for Computational Linguistics, Florence, Italy, 2019, p. 111–116. URL: https://www.aclweb.org/anthology/P19-3019. doi:10.18653/v1/P19-3019.

[9] Y. Tian, H. Chen, X. Wang, Z. Bai, Q. Zhang, R. Li, C. Xu, Y. Wang, Multiscale positive-unlabeled detection of ai-generated texts (2023). URL: http://arxiv.org/abs/2305.18149, arXiv:2305.18149 [cs].

[10] K. Wu, L. Pang, H. Shen, X. Cheng, T.-S. Chua, Llmdet: A large language models detection tool (2023). URL: http://arxiv.org/abs/2305.15004, arXiv:2305.15004 [cs].

[11] J. Bevendorff, M. Wiegmann, J. Karlgren, L. Dürlich, E. Gogoulou, A. Talman, E. Stamatatos, M. Potthast, B. Stein, Overview of the "Voight-Kampff" Generative AI Authorship Verification Task at PAN and ELOQUENT 2024, in: G. Faggioli, N. Ferro, P. Galuščáková, A. G. S. de Herrera (Eds.), Working Notes of CLEF 2024 - Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2024.

[12] W. Huang, A. Murakami, J. Grieve, Alms: Authorial language models for authorship attribution, 2024. arXiv:2401.12005.

[13] M. Fröbe, M. Wiegmann, N. Kolyada, B. Grahm, T. Elstner, F. Loebe, M. Hagen, B. Stein, M. Potthast, Continuous Integration for Reproducible Shared Tasks with TIRA.io, in: J. Kamps, L. Goeuriot, F. Crestani, M. Maistro, H. Joho, B. Davis, C. Gurrin, U. Kruschwitz, A. Caputo (Eds.), Advances in Information Retrieval. 45th European Conference on IR Research (ECIR 2023), Lecture Notes in Computer Science, Springer, Berlin Heidelberg New York, 2023, pp. 236–241. doi:10.1007/978-3-031-28241-6_20.

[14] S. Chakraborty, A. S. Bedi, S. Zhu, B. An, D. Manocha, F. Huang, On the Possibilities of AI-Generated Text Detection, 2023. URL: http://arxiv.org/abs/2304.04736, arXiv:2304.04736 [cs].

[15] A. Hans, A. Schwarzschild, V. Cherepanova, H. Kazemi, A. Saha, M. Goldblum, J. Geiping, T. Goldstein, Spotting LLMs With Binoculars: Zero-Shot Detection of Machine-Generated Text, 2024. URL: http://arxiv.org/abs/2401.12070, arXiv:2401.12070 [cs].

[16] J. Tyo, B. Dhingra, Z. C. Lipton, On the state of the art in authorship attribution and authorship verification (2022). URL: http://arxiv.org/abs/2209.06869, arXiv:2209.06869 [cs].

[17] G. Barlas, E. Stamatatos, Cross-Domain Authorship Attribution Using Pre-trained Language Models, volume 583 of *IFIP Advances in Information and Communication Technology*, Springer International Publishing, Cham, 2020, p. 255–266. URL: http://link.springer.com/10.1007/978-3-030-49161-1_22. doi:10.1007/978-3-030-49161-1_22.

[18] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, et al., Llama 2: Open Foundation and Fine-Tuned Chat Models, 2023. URL: https://arxiv.org/abs/2307.09288v2.

[19] OpenAI, GPT-4 Technical Report, 2023. URL: http://arxiv.org/abs/2303.08774, arXiv:2303.08774 [cs].

[20] J. Bevendorff, M. Wiegmann, M. Potthast, B. Stein, E. Stamatatos, PAN24 Voight-Kampff Generative AI Authorship Verification, 2024. URL: https://doi.org/10.5281/zenodo.10718757. doi:10.5281/zenodo.10718757.