Overview of the DagPap24 Shared Task on Detecting Automatically Generated Scientific Papers

Savvas Chamezopoulos[1] **Drahomira Herrmannova**[1] **Anita de Waard**[1]
Domenic Rosati[2] **Yury Kashnitsky**[1]
Elsevier, USA[1] Dalhousie University, Canada [2]
{s.chamezopoulos, d.herrmannova, a.dewaard}@elsevier.com,

Abstract

This paper provides an overview of the 2024 ACL Scholarly Document Processing workshop shared task on the detection of automatically generated scientific papers. Unlike our previous task, which focused on the binary classification of whether scientific passages were machine-generated or not, one likely use case for text generation technology in scientific writing is to intersperse human-written text with passages of machine-generated text. We frame the detection problem as a multi-class span classification task: given an expert of text, label token spans in the text as human-written or machine-generated We shared a dataset containing excerpts from human-written papers as well as artificially generated content collected by Elsevier publishing and editorial teams. As a test set, the participants were provided with a corpus of openly accessible human-written as well as generated papers from the same scientific domains of documents. The shared task saw 457 submissions across 28 participating teams and resulted in three published technical reports. We discuss our findings from the shared task in this overview paper.

1 Introduction

A big problem with the ubiquity of Generative AI is that it has now become very easy to generate fake scientific papers. This can erode public trust in science and attack the foundations of science: are we standing on the shoulders of robots? One notorious example is how Rafael Luque massively used chatGPT to “polish“ papers that were later found[1] in authorship for sale advertisements. According to a recent study (Gray, 2024), chatGPT “contamination“ is seen in at least 60,000 published papers (slightly over 1% of all articles). For the publishing year 2023, it is found that several specific words like "commendable", "intricate" or "meticulously" show a distinctive and disproportionate increase in their prevalence, which might be an indication of LLM assistance in writing.

The Detecting Automatically Generated Papers (DAGPap[2]) competition aims to encourage the development of robust, reliable AI-generated scientific text detection systems, utilizing a diverse dataset and varied machine learning models in a number of scientific domains.

Building on the DagPap22 competition Kashnitsky et al. (2022), this year’s dataset consisted of 30,000 scientific articles sourced from ScienceDirect[3] that were processed to integrate various alteration methods within the human-written content.

As for the previous challenge, the 2024 DagPap challenge is a collaboration between a publisher (Elsevier) and the research community to attempt a resolution through technical means.

2 Related work

Since the DagPap22 competition, there have been several more efforts on addressing the same problem.

SemEval 2024 hosted a task[4] on multi-generator, multi-domain, and multilingual black-box machine-generated text detection (Wang et al., 2024) (to appear at NAACL 2024). The Shared Task offered 4 subtasks (monolingual and multilingual text classification, multi-way classification and human-machine text boundary detection) and attracted a couple hundred teams. The organizers found that all classification tasks turned out to be

[1] https://tinyurl.com/rafaelluque

[2] https://sdproc.org/2024/sharedtasks.html#dagpap

[3] https://www.sciencedirect.com/

[4] https://github.com/mbzuai-nlp/SemEval2024-task8

Proceedings of the Fourth Workshop on Scholarly Document Processing (SDP 2024), pages 7–11

relatively easy to solve, with many teams beating baselines. The boundary detection problem, however, turned out to be harder to solve.

CLIN33 hosted a similar Shared Task (Fivez et al., 2024) aimed at automatic detection of AI-generated texts in English and Dutch, spanning multiple genres: medium-length news articles, tweets from the social media platform X (previously known as Twitter), product reviews, short-form poetry, and journal columns. Thus, the task focused on a cross-domain multilingual binary classification setup that included entirely new held-out test genres. The results were close to perfect for some genres (e.g., news and reviews) yet unsatisfactory for others (e.g., poetry).

Kaggle, a popular platform for machine learning competitions, also hosted a similar contest[5] "LLM - Detect AI-Generated Text" (King et al., 2023). The competition attracted over 5,000 participants from 4,300 teams and over 110,000 submissions. Although Kaggle is well-known for establishing state-of-the-art in applied machine learning (e.g., the usage of DeBERTa (He et al., 2021) for most NLP tasks), the focus of many competitors of the mentioned contest were on reverse-engineering the prompts used to generate the test set. This may limit the generalizability and applicability of these findings to real-world scenarios.

Some more similar competitions include AuTexTification (Automated Text Identification) (Sarvazyan et al., 2023) and MLMAC (Merkhofer et al., 2023) (Machine Learning Model Attribution Challenge). Interestingly, MLMAC organizers note that the most successful approaches were manual, as participants observed similarities between model outputs and developed attribution heuristics based on public documentation of the base models.

Some notable leaderboards for human- vs. machine-text detection and author attribution are:

- TuringBench (Uchendu et al., 2021), which consists of 20 labels (19 AI text-generators and human) and includes 200K articles, created by prompting AI text-generators with titles of 10K news articles from sources like CNN;
- MULTITuDE (Macko et al., 2023) is a large-scale multi-lingual benchmark comprising 74,000 texts generated by 8 LLMs in 11 languages. Here, authors conclude that detectors struggle with generalizing to the unseen languages, texts from different domains, writing styles, and unknown language models, decoding strategies, or obfuscation efforts. Notably, some commercially available detectors fail at this benchmark. In accordance with Kaggle experience, DeBERTA-v3 is the best base for machine learning types of detectors;
- RAID (Dugan et al., 2024) is probably the largest benchmark at the time of writing: in includes over 6 million generations spanning 11 models, 8 domains, 11 adversarial attacks and 4 decoding strategies. Using RAID, the authors evaluate the out-of-domain and adversarial robustness of 8 open- and 4 closed-source detectors and find that current detectors are easily fooled by adversarial attacks, variations in sampling strategies, repetition penalties, and unseen generative models.

While most of the competitions and leaderboards address the problem of full-text classification into different classes (human/machine or several LLMs as origins; and only SemEval subtask 3 is an exception); for DAGPap24, we focus on a token-level text classification, for a scenario when human writing is interspersed with passages of machine-generated text.

3 Corpus creation

The dataset comprised 30,000 scientific articles sourced from ScienceDirect[6]. These articles' full texts were processed to integrate various alteration methods within the human-written content. For each alteration, the text segment contained between 2,500 to 4,000 characters. Approximately 25% of the entire text was modified using one of the following techniques:

1. **Synonym Replacement using NLTK** (Bird et al., 2009): Approximately 75% of the eligible words were substituted with random choices from their respective synonyms. An eligible word is defined as a non-stop word that has at least one synonym other than itself.

[5] https://www.kaggle.com/competitions/llm-detect-ai-generated-text

[6] https://www.sciencedirect.com/

Column name	Data
index	0
text	Across the world...
annotations	[[0, 3779, human], [3780, 7601, chat_gpt], ...
tokens	[Across, the, world, ...
token_label_ids	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...

Table 1: Example data row

2. **Text Summarization with T5-Small** (Raffel et al., 2020): The T5-small model, implemented through HuggingFace (Wolf et al., 2020), was utilized for text summarization. The summarized text ranges between 80% to 120% of the original text length.

3. **Paraphrasing using ChatGPT** [7]: The ChatGPT 3.5 Turbo model (accessed via the API) was employed for text paraphrasing, refining the original text into alternative expressions.

Before applying any alterations, the texts were segmented into individual sentences. This segmentation was crucial to maintain proper readability and to better simulate scenarios where machine learning and AI are utilized as writing assistants. A significant aspect of this dataset creation is the incorporation of randomness. Both the text chunks where alterations would be made and the selection of the alteration method were determined at pseudo-random intervals. After processing, the text was tokenized into whole words. These tokens, along with their respective annotation spans, constitute the label columns. An example row of the final dataset can be found in Table 1.

4 Competition setup

4.1 Metric and data split

The metric chosen in the competition is Macro F1 score. For each full text, the macro F1 score is calculated and the final score is the average across all scores.

The final data was divided into train, development (dev), and test sets, resulting in 5000 training records 5000 dev records, and 20000 test records for the competition[8].

[7] https://chatgpt.com/

[8] https://www.codabench.org/competitions/2431/#/pages-tab

4.2 Baselines

As organizers, we provided 3 baselines:

- A fine-tuned DistilBERT (Sanh et al., 2019) with an **84%** test set average macro F1 score
- A fine-tuned SciBERT (Beltagy et al., 2019) with an **87%** test set average macro F1 score
- An indicative all-zeros prediction with a **36%** test set average macro F1 score

All instructions and baseline code were made public to the participants via CodaBench [9]. The competition was comprised of two phases: the development and the testing phase. During the development phase, both the training and dev sets were made available to the competitors. Of course, the label columns were not available in the dev set. They could freely measure the performance of their systems on the dev set by providing a set of predictions to be measured against the correct labels in a pre-configured online platform. During the testing phase, the test set (excluding labels) was shared with the competitors, and they could post up to two prediction files to test their models.

4.3 Awards

Elsevier sponsored monetary prizes for a total of USD $5000 to the three top ranking teams. For more details, see the competition website[10].

5 Results

28 teams participated in the task this year, with a total of 457 submissions. Out of these, 19 teams managed to beat the SciBERT baseline on the development set, and 12 teams managed to beat the SciBERT baseline on the test set. The performance of the top 4 teams was particularly impressive, all exceeding an F1-score of 99% on the test set.

3 papers got accepted at the Scholarly Document Processing workshop dedicated to the shared task (Gritsai et al., 2024; Andreev et al., 2024; Zhao et al., 2024).

In (Zhao et al., 2024), they address the challenge by using two tokenization methods, fine-tuning various language models, introducing an Anomalous Label Smoothing (ALS) method and

[9] https://github.com/ChamezopoulosSavvas/DAGPap24

[10] https://www.codabench.org/competitions/2431/#/pages-tab

employing a majority voting method for ensembling model predictions. ALS is similar to a Conditional Random Fields (CRF) in filtering out unreasonable labels. It ensures that the distribution of predicted labels aligns with that of the training set. The team finished at the 2nd place on the final leaderboard with 0.9948 and 0.9944 F1 scores during the development and testing phases respectively. We are interested to see how similar techniques can be applied to the problem in general, when test data distribution does not always match the training data distribution.

The authors of (Andreev et al., 2024) achieved an F1 score of 89.83 during the competition (landing at the 6th place on the leaderboard) and 99.46 afterward by fixing a tokenization issue. Their solution was also to ensemble several models, here they ensembled several DeBERTa models with varying the number of layers frozen, and input token lengths.

Finally, (Gritsai et al., 2024) provided a multi-task setting where one linear layer was trained for the binary prediction task (human- or machine-written), and the other was multi-class. They used the binary classifier to guide whether they used the multi-class classifier.

6 Discussion

During the previous iteration of the DAGPap challenge (Kashnitsky et al., 2022), the participants were able to achieve $> 99\%$ F1 scores. Our hypothesis was that one of the reasons for this could be that the techniques used for generating "fake" examples tended to generate content that often included specific patterns. For example, the model used for summarization tended to open with phrases like "This paper is focused on ..." or "In this paper, the authors ...". To alleviate this issue, for this round of the competition, we leveraged more robust techniques for generating the competition corpus, specifically, gpt-3.5-turbo models, T5-Small, and NLTK synonym replacement. The task was also reformulated from sequence classification in the previous round to token classification in this round.

Despite these changes, some participants were still able to achieve over 99% F1 scores. We believe this demonstrates the difficulty in creating representative training corpora for the detection of machine-generated scientific content – in the experience of our teams at Elsevier and as reported in (Rosati, 2022; Macko et al., 2023; Dugan et al., 2024), detection of machine-generated scientific content is still a challenging task despite these encouraging results. However, we believe that the DAGPap24 shared task did offer a step forward to explore this challenging problem, and we hope to work together with the community on resolving this pernicious issue.

For future work, we plan to design a competition that focuses on real-world scenarios where human-written and machine-generated scientific texts are mixed. We also plan to investigate the robustness of detection systems against adversarial attacks in the scientific domain, where the generated text is intentionally modified to evade detection.

Acknowledgements

The work is partially supported by the European Research Council (ERC) grant agreement no. 951393.

References

Nikita Andreev, Alexander Shirnin, Vladislav Mikhailov, and Ekaterina Artemova. 2024. Papilusion at DAGPap24: Paper or illusion? detecting AI-generated scientific papers. In *The 4th Workshop on Scholarly Document Processing @ ACL 2024*.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Liam Dugan, Alyssa Hwang, Filip Trhlik, Josh Magnus Ludan, Andrew Zhu, Hainiu Xu, Daphne Ippolito, and Chris Callison-Burch. 2024. Raid: A shared benchmark for robust evaluation of machine-generated text detectors.

Pieter Fivez, Walter Daelemans, Tim Van de Cruys, Yury Kashnitsky, Savvas Chamezopoulos, Hadi Mohammadi, Anastasia Giachanou, Ayoub Bagheri, Wessel Poelman, Juraj Vladika, Esther Ploeger, Johannes Bjerva, Florian Matthes, and Hans van Halteren. 2024. The clin33 shared task on the detection of text generated by large language models. *Computational Linguistics in the Netherlands Journal*, 13:233–259.

Andrew Gray. 2024. Chatgpt "contamination": estimating the prevalence of llms in the scholarly literature.

German Gritsai, Ildar Khabutdinov, and Andrey Grabovoy. 2024. Multi-head span-based detector for AI-generated fragments in scientific papers. In *The 4th Workshop on Scholarly Document Processing @ ACL 2024*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Yury Kashnitsky, Drahomira Herrmannova, Anita de Waard, George Tsatsaronis, Catriona Catriona Fennell, and Cyril Labbe. 2022. Overview of the DAGPap22 shared task on detecting automatically generated scientific papers. In *Proceedings of the Third Workshop on Scholarly Document Processing*, pages 210–213, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Jules King, Perpetual Baffour, Scott Crossley, Ryan Holbrook, and Maggie Demkin. 2023. Llm - detect ai generated text.

Dominik Macko, Robert Moro, Adaku Uchendu, Jason Lucas, Michiharu Yamashita, Matúš Pikuliak, Ivan Srba, Thai Le, Dongwon Lee, Jakub Simko, and Maria Bielikova. 2023. MULTITuDE: Large-scale multilingual machine-generated text detection benchmark. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9960–9987, Singapore. Association for Computational Linguistics.

Elizabeth Merkhofer, Deepesh Chaudhari, Hyrum S. Anderson, Keith Manville, Lily Wong, and João Gante. 2023. Machine learning model attribution challenge.

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

Domenic Rosati. 2022. SynSciPass: detecting appropriate uses of scientific text generation. In *Proceedings of the Third Workshop on Scholarly Document Processing*, pages 214–222, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Areg Mikael Sarvazyan, José Ángel González, Marc Franco-Salvador, Francisco Rangel, Berta Chulvi, and Paolo Rosso. 2023. Overview of autextification at iberlef 2023: Detection and attribution of machine-generated text in multiple domains.

Adaku Uchendu, Zeyu Ma, Thai Le, Rui Zhang, and Dongwon Lee. 2021. Turingbench: A benchmark environment for turing test in the age of neural text generation. In *Proceedings of the Findings of the 2021 Empirical Methods in Natural Language Processing (EMNLP)*, Punta Cana, Dominican Republic.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, Thomas Arnold, Chenxi Whitehouse, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. 2024. Semeval-2024 task 8: Multidomain, multimodel and multilingual machine-generated text detection.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.

Yuan Zhao, Junruo Gao, Junlin Wang, Gang Luo, and Liang Tang. 2024. Utilizing an ensemble model with anomalous label smoothing to detect generated scientific papers. In *The 4th Workshop on Scholarly Document Processing @ ACL 2024*.