# Overview of the "Voight-Kampff" Generative AI Authorship Verification Task at PAN and ELOQUENT 2025

Janek Bevendorff[1,2,*], Yuxia Wang[3], Jussi Karlgren[4], Matti Wiegmann[2,5], Maik Fröbe[6], Akim Tsivgun[7], Jinyan Su[8], Zhuohan Xie[3], Mervat Abassy[9], Jonibek Mansurov[3], Rui Xing[3,10], Minh Ngoc Ta[11], Kareem Ashraf Elozeiri[12], Tianle Gu[13], Raj Vardhan Tomar[14], Jiahui Geng[3], Ekaterina Artemova[15], Artem Shelmanov[3], Nizar Habash[16], Efstathios Stamatatos[17], Iryna Gurevych[3,18], Preslav Nakov[3], Martin Potthast[5,19] and Benno Stein[2]

[1]*Leipzig University, Germany*
[2]*Bauhaus-Universität Weimar, Germany*
[3]*Mohamed bin Zayed University of Artificial Intelligence, UAE*
[4]*University of Helsinki, Finland*
[5]*University of Kassel, Germany*
[6]*Friedrich-Schiller-Universität Jena, Germany*
[7]*Nebius AI, Netherlands; KU Leuven, Belgium*
[8]*Cornell University, USA*
[9]*Alexandria University, Egypt*
[10]*The University of Melbourne, Australia*
[11]*BKAI Research Center, Hanoi University of Science and Technology, Vietnam*
[12]*Zewail City of Science and Technology, Egypt*
[13]*Tsinghua University, China*
[14]*Cluster Innovation Center, University of Delhi, India*
[15]*Toloka AI, Netherlands*
[16]*New York University Abu Dhabi, UAE*
[17]*University of the Aegean, Greece*
[18]*TU Darmstadt, Germany*
[19]*hessian.AI, Germany; ScaDS.AI, Germany*

*pan@webis.de* *https://pan.webis.de* *https://eloquent-lab.github.io*

## Abstract

The "Voight-Kampff" Generative AI Authorship Verification task aims to determine whether a text was generated by an AI or written by a human. The 2025 edition of the task explores two subtasks:

Subtask 1 tests the detection of purely AI generated text with potentially unknown obfuscations, and as such continues our research from 2024. The task is again organized as a builder-breaker challenge together with the ELOQUENT lab. The PAN participants submitted 24 detectors. The best system archives a mean score of 0.99, the best baseline achieves a score of 0.92. ELOQUENT participants submitted 13 new test datasets with 22 obfuscated texts each. The most difficult dataset archives a mean $C@1^{-1}$ score of 0.63.

Subtask 2 investigates texts with six degrees of human-AI collaboration: (i) fully human-written, (ii) human-written, then machine-polished, (iii) machine-written, then machine-humanized (obfuscated), (iv) human-initiated, then machine-continued, (v) deeply mixed text,where some parts are written by a human and some are generated by a machine, and (vi) machine-written, then human-edited. The dataset contains over half a million examples in total and is composed from several relevant AI-detection datasets across multiple text genres. PAN participants submitted 21 detectors to subtask 2. The best system archives an $F_1$ score of 0.65, the best baseline a score of 0.48.

The data, baselines, and the code used for creating the datasets and evaluating the systems are available.[1]

## Keywords

Generative AI Detection, LLM Detection, Human-AI Collaboration, Workshop, PAN

[1] Code and data for subtask 1 and subtask 2 are available on GitHub.

*CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain*

*Corresponding author.

# 1. Introduction

Authorship verification is a fundamental task in author identification. PAN has continuously been organizing authorship verification tasks for years [1, 2, 3, 4] and with generative AI / LLM detection being fundamentally also an authorship verification task [5], we decided to "delve" into that realm.

For the 2025 edition of this task, we increase the difficulty of the binary detection setting and add a new subtask with a focus on detecting human-AI collaboration. In the first edition of this task [6], "Voight-Kampff" was organized as a classic authorship verification task: Given two texts, one authored by a human, one by a machine: pick out the human. Against our initial assumptions, the task was of little challenge to the participants' systems even in the face of text obfuscation. Thus, for this year, we elevate the challenge in two directions.

First, we reduce the original "Voight-Kampff" setting to a binary task: *Given a text, determine if it was written by a human or an AI model.* In addition, we extend our selection of obfuscation methods to test model sensitivity and add larger, newer, and more difficult-to-detect models. As before, the PAN and ELOQUENT labs jointly organize this task: ELOQUENT participants provide very strongly obfuscated texts from a set of summaries, and PAN participants develop systems to robustly detect AI generated text. The ELOQUENT contributions are described in Section 2 and the PAN contributions are described in Section 3. As in the previous year, all systems are submitted as immutable docker containers via TIRA [7] for easy reproducibility.

Second, we organize a new subtask aiming to detect human-AI collaboration, since joint writing and editing of documents with AI tools has become commonplace and such cases are presumably harder to detect than in the binary case. The subtask is a logical continuation of previous shared tasks organized at SemEval and GenAIDetect [8, 9]. Subtask 2 asks to distinguish six cases of joint human-AI authorship: (i) fully human-written, (ii) human-written, then machine-polished, (iii) machine-written, then machine-humanized (obfuscated), (iv) human-initiated, then machine-continued, (v) deeply mixed text, in which some parts are written by a human and some are generated by a machine, and (vi) machine-written, then human-edited. This subtask is described in Section 4.

# 2. ELOQUENT: Generating Hard-to-detect AI-texts

ELOQUENT participants generate datasets of machine text, attempting to break the classifier systems built by PAN participants.

In its first edition, the classifiers submitted by participants to the PAN lab handily classified the texts into human vs. machine. We found that of the submitted datasets in 2024, all were able to fool some of the classifier systems some of the time; but no generative model was consistently able to convince the better classifier systems that it was human. It was clear that machine-generated texts appeared to consistently hold to certain detectable stylistic indicator features. [6]

## 2.1. Dataset

For the test set, 22 texts written by human authors, of between 350–700 words were selected. Most original texts were longer and a suitably long section of the text was selected. Summaries of each text were generated by the organizers using OpenAI's ChatGPT service using the prompt "Summarize the main points of the following text and give an overall description of the genre and tone of the text." Those summaries were then shared to the participants for their systems to generate short texts on the basis of the summaries. A sample summary test item is given in 1 and a list of item titles is given in 1.

A suggested prompt was given with the participants—"Write a text of about 500 words which covers the following items"—but the participants were free to formulate their own prompts as they saw fit. The generated texts were submitted by the participants through a submission form, and then further submitted by the organizers to the PAN lab for classification.

**Id**: 041

**Genre and Style**:

*Genre:* Mythological and Linguistic Ethnography / Cultural Anthropology

*Tone:* Scholarly, reverent, and lyrical, blending academic analysis with a poetic appreciation of language, mythology, and cultural worldview.

**Content**:

Finnish language and culture are deeply intertwined with nature, with precise and acoustically rich verbs used to describe natural elements like snow, wind, and animals.

Ancient Finns practiced animistic nature-worship, viewing all visible forces sun, moon, sea, earth as living, conscious beings.

Over time, belief evolved to include invisible spiritual beings, or habitat (genii/regents), who governed natural elements and had both form and spirit, though lesser ones were more formless and abstract.

These haltiat were immortal and hierarchical, often ranked based on the significance of their domain (e.g., Tapio of the forest outranking Pilajatar, daughter of the aspen).

Finnish mythology emphasizes the independence and dignity of each deity, regardless of power; even a minor god rules absolutely within their sphere.

Deities were typically paired and familial, with the sky and celestial bodies being the earliest and most revered objects of worship, leading to the concept of Jumala, the thunder-home, as the supreme god.

**Figure 1:** A sample test item for the Voight-Kampff Task.

**Table 1**
Voight Kampff 2025 test data items. All original texts were taken from sites with documented pre-2020 versions of text sources available or directly sourced from the author.

| Id | Title | Source |
|---|---|---|
| 030 | 419 letter | archive.org |
| 031 | 419 letter | archive.org |
| 032 | The banker and the bear, 1900 | gutenberg.org |
| 033 | Baths and Bathing, 1879 | gutenberg.org |
| 034 | Two years' captivity in German East Africa, 1919 | gutenberg.org |
| 035 | JR Cigars, 2012 | archive.org |
| 036 | Session moderator instructions, 1990 | lingvi.st |
| 037 | Book of Esther, ˜400 BC (En translation 1901) | readbibleonline.net |
| 038 | Maastricht Treaty, 1992 | cvce.eu |
| 039 | The Blue Varient, 2011 | fanfic.net |
| 040 | Wisdom of Father Brown, 1914 | gutenberg.org |
| 041 | Kalevala, foreword of En translation, 1888 | gutenberg.org |
| 042 | What is Free Software?, 1990 | gnu.org |
| 043 | Gripes about reviewing, 2008 | lingvi.st |
| 044 | Letters to Guy, 1885 | gutenberg.org |
| 045 | Intro to LLMs, 2025 | acm.org/cacm |
| 046 | Nobel Peace Prize acceptance speech, 2014 | nobelprize.org |
| 047 | Norse Mythology, 1876 | gutenberg.org |
| 048 | Baths and Bathing, 1879 | gutenberg.org |
| 050 | Steppenwolf, (En translation 1929) | gutenberg.org |
| 051 | Free trade | wikipedia.org |
| 052 | Saffron | wikipedia.org |

**Table 2**
Accuracy of classifiers at distinguishing human-authored from machine-generated text as measured by the inverse C@1 score averaged over all participating classifiers. A low score indicates successful classification of a text to be human-generated or machine-generated; a high score indicates that classifiers misclassified an item to be human authored when it in fact was machine-generated, i.e., that a generative model was able to fool the classifiers more often.

| Team | Experiment | Model | C@1$^{-1}$ | FPR | FNR | Ref. |
|---|---|---|---|---|---|---|
| *Human* | | | 0.136 | 0.148 | – | |
| JUNLP_SS | | GPT-2 | 0.477 | – | 0.465 | [10] |
| HumanAIzers | Translation Back-Translation | Qwen3-8B | 0.618 | – | 0.609 | [11] |
| HumanAIzers | Homoglyph Attack | Qwen3-8B | 0.436 | – | 0.428 | |
| HumanAIzers | ADHD Writing Workflow | Qwen3-8B | 0.142 | – | 0.136 | |
| HumanAIzers | Persona-Based Generation | Qwen3-8B | 0.083 | – | 0.081 | |
| Moa | | | 0.178 | – | 0.175 | |
| PJS-team | En-Hi-En v2 | Claude Sonnet3.5 | **0.626** | – | 0.625 | [12] |
| PJs-team | En-Hi-En | Claude Sonnet3.5 | 0.280 | – | 0.276 | |
| UTK | v2 | | 0.222 | – | 0.214 | [13] |
| UTK | v1 | | 0.177 | – | 0.172 | |

### 2.2. Submissions

Of the 49 registered participants, five teams submitted 10 experiments [10, 11, 12, 13, 14]. The attrition rate is great, similar to last year, and we will investigate the possibility of turning this task into a continuously open experiment with asynchronous submission.

Table 2 lists the participating systems and the classification results per system using the C@1 accuracy score aggregated from the PAN lab participants.

### 2.3. Observations

This year, we find that of the submitted ELOQUENT-generated datasets, all were able to fool some of the classifier systems some of the time, and several were able to do so better than chance. This is a considerable improvement from last year, and reflects a more directed experimentation in the submitted experiments, e.g., by inserting explicit human variation and departure from norm in the generated texts.

The human authored texts were on average misclassified as machine generated only 15% of the time and this is entirely due to two texts which both caused a majority of the classification systems to assign them a non-zero likelihood to be machine-generated: the excerpt from the Maastricht treaty (038) and the Intro to Large Language Models (045). All other human-authored texts were correctly assumed to be human-authored by every participating classifier system. The best generative models fooled the classifiers with many, but never with all of their generated texts. Some of the test items appear to have been easier to generate convincing human-like output for: Session Moderator Instructions (036), What is Free Software? (042), and the Wikipedia text on Saffron (052). It is unclear if this is an effect of the qualities of the summaries or if the language model training data are better equipped for academic text than for other genres.

As a very general finding, it is clear that generative language models still have recognizable traits in their output and that classifiers are able to pick up on them quite effectively. Generating truly human-like text remains an open challenge for developers and operators of generative language models.

## 3. PAN Subtask 1: Voight-Kampff AI Detection Sensitivity

At PAN 2024 [15], we offered, for the first time, the *"Voight-Kampff" Generative AI Authorship Verification* task [6], which attracted a large number of submissions. To start with, we formalized different task variants and ordered them from easiest to hardest (Figure 2). To establish a baseline, we decided to start

with the easiest variant, in which participants were given a pair of texts, of which exactly one was of human and the other of machine origin.

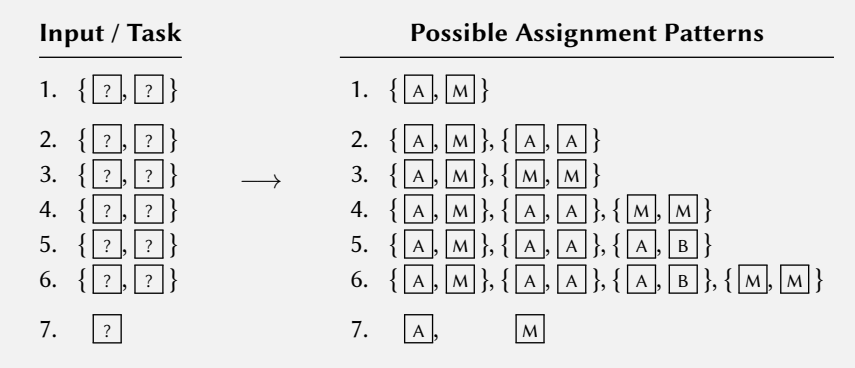


**Figure 2:** Hierarchy of authorship verification problems from *easiest* (1) to *hardest* (7), involving LLM-generated text. Ignoring mixed human and machine authorship, the difficulty arises from the pairing constraints imposed by the possible assignment patterns. M denotes LLM-generated text, while A and B denote human-authored text (same letter meaning same human author).

For PAN 2025 [16], we move on to the harder variant, in which participants are given only one text. This variant reflects a more realistic scenario of authorship verification "in the wild," and it also aligns with the settings commonly addressed in other LLM detection shared tasks. The PAN 2025 subtask 1 is again co-organized with the ELOQUENT [17] in a builder-breaker style.

The subtask 1 is in essence the classic binary detection task known also from other LLM detection shared tasks. However, we are testing the limits of the detectors by crafting a test set with text "obfuscations" that try to evade detection. Apart from drastic text length restrictions, the obfuscations we tested or received from ELOQUENT participants in the previous year had turned out to be mostly ineffective. So this year, we tested what happens when the human writers obfuscate their style and whether machines can replicate this.

### 3.1. Dataset

The training, validation, and test datasets were built from a selection of 19$^{\text{th}}$-century English fiction from Project Gutenberg,[1] the Extended Brennan-Greenstadt [18] corpus, and the Riddell-Juola [19] corpus. We also included a sample of the PAN'24 training and test sets (containing U.S. news articles from 2021) in this year's training and test sets (the PAN'24 test data, which were never openly shared). Participants were free to use any other sources for augmenting their training data, including the PAN'24 training set.

The (Extended) Brennan-Greenstadt corpus has been used a lot in the fields of authorship verification and authorship obfuscation. For its construction, volunteers on Amazon Mechanical Turk were asked to submit existing writing samples (about 500 words) of their own. They were then asked to write another text describing their neighborhood, but in a way that hides their own writing style. No particular instructions were given for how this style obfuscation was to be achieved. They were also asked to write a third text in which they should try to imitate a particular style from a given sample of a novel. For PAN, we used the original and obfuscated neighborhood essays only. We chose this corpus in particular, as the obfuscated nature of the texts adds an interesting aspect to the classification task.

The Riddell-Juola corpus was created for a replication study of the original experiments by Brennan et al., but on a larger scale and with an additional control group. Volunteers were also asked to submit existing writing samples and a new piece about their neighborhood. However, the control group was not instructed to obfuscate their style.

[1] https://www.gutenberg.org/

You are an essay summarizer and a forensic writing style analyst. Given an essay, you summarize its key points in ten bullet points, extract the main topic in just a few words, and some other details.

As additional details, classify the point of view from which the essay is written ("first-person", "third-person", "second-person") and the tense ("past-tense" or "present-tense"). If the essay is argumentative, classify the author's stance on the main topic as "pro" or "con" or "neutral" if the article is not argumentative.

Point out any traits that make the author's writing style unique, but phrase them as instructions for another writer who wants to imitate the style. For example: "Use very short sentences," "Use passive voice a lot," "Write in a nominal style with few adjectives," "Use very poetic and descriptive language," "Add spelling mistakes," "Start multiple sentences with the same word," "Use the word X more often than usual," "Write in long and nested sentences," "Use a lot of technical terms," "Use very simple language," "Use metaphors/similes/anaphoras/alliterations/...," etc.

Answer in structured JSON format (without Markdown formatting) like so:

```
{
   "key_points": ["key point 1", "key point 2", ...],
   "main_topic": "essay topic",
   "pov": "narrative point of view, either first-person, third-person, or
second-person",
   "tense": "essay tense, either past-tense or present-tense",
   "stance": "stance on topic, either pro, con, or neutral",
   "style_instructions": ["style instruction 1", "style instruction 2", ...]
}
```

**Figure 3:** Prompt used for summarizing essays from the Brennan-Greenstadt and Riddell-Juola corpora.

From Project Gutenberg, we sampled 927 English novels tagged as "19th-century" and "Fiction," trying not to duplicate individual titles. The texts were cleaned of headers and footers and split into no more than 10 chunks of 500–700 words, resulting in a total of 9,185 original human texts.

#### 3.1.1. Summary Generation

From the essays, we generated JSON-structured bullet-point summaries with GPT-4o using the prompt displayed in Figure 3. The model was instructed to extract a list of key points from the text, but also stylistic markers such as the narrative point of view, the tense, stance, and any other traits that seem unique to that text (such as the use of many technical terms, very poetic language, etc.). The neighborhood texts were summarized with a similar prompt, but with the additional information that the text describes a neighborhood and that the author tried to hide their writing style. For the $19^{\text{th}}$-century fiction texts, a much simpler prompt was used that asked only for a bullet-point summary given a section of a novel.

#### 3.1.2. Machine Text Generation

From the summaries, we generated the final machine texts using 14 different LLMs: *GPT-3.5 Turbo, GPT-4o, GPT-4o-mini, GPT-4.5-preview, OpenAI o1, OpenAI o1-mini, OpenAI o3-mini, Gemini 1.5 Pro, Gemini 2.0 Flash, DeepSeek-R1-Qwen-32b, Falcon3-10b, Llama3.1-8b, Llama3.3-70b, Ministral-8b-2410.*

Using the summaries extracted earlier, we asked the models to write (1) argumentative essays for each of the original and obfuscated neighborhood essays from the Brennan-Greenstadt and Riddell-Juola corpora; and (2) continuations of the novel chunks as faithful pastiches in honor of the original authors. In both cases, the models were instructed to follow the given style instructions. For the neighborhood essays, the models were explicitly told to follow these instructions to mask their own style. For the novels, we asked the models not to plagiarize the original texts, since we were almost sure that Project

**Table 3**
Training, validation, and test splits for PAN subtask 1 by data source. The test set contains unobfuscated human and LLM texts, obfuscated human texts (Brennan-Greenstadt / Riddell-Juola neighborhood writing prompt), obfuscated LLM texts, held-back OpenAI o1 and DeepSeek R1 texts, a sample of the PAN'24 test set, and submissions from ELOQUENT participants. The individual test data subsets (Section 3.1.4) are built by combining human and LLM texts for each source. Obfuscated human / LLM texts and OpenAI o1 texts form their own subsets.

| | **Training** | | **Validation** | | **Test** | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Data Source** | Human | LLM | Human | LLM | Human | LLM | H. Obfs. | LLM Obfs. | LLM o1 |
| *All* | 9,101 | 14,606 | 1,277 | 2,312 | 1,497 | 2,216 | 72 | 1,285 | 300 |
| Brennan-Greenstadt | 490 | 1,958 | 69 | 280 | 140 | 559 | 45 | 300 | – |
| Riddell-Juola | 441 | 1,772 | 63 | 253 | 128 | 504 | 27 | 600 | 100 |
| Gutenberg Fiction | 7,300 | 7,397 | 928 | 909 | 957 | 880 | – | 100 | 200 |
| PAN'24 Train | 870 | 3,479 | 217 | 870 | – | – | – | – | – |
| PAN'24 Test | – | – | – | – | 272 | 273 | – | – | – |
| ELOQUENT | – | – | – | – | 22 | – | – | 285 | – |

Gutenberg was part of the training data of most models. Despite this instruction, some of the models still reproduced parts of the original novels, but to a degree we deemed acceptable (e.g., OpenAI o1-mini reproduced two paragraphs of Charlotte Brontë's "The Professor" before continuing with new text).

### 3.1.3. Machine Text Obfuscation

In addition to the style-informed prompts described above, we also prompted GPT-4o and OpenAI o1 with five variations of the essay prompts, in which we instructed the models (1) to write like a 7-year-old, (2) write like a 7-year-old, but only given the main topic and no key points, (3) start every sentence with an alliteration, (4) read but ignore a list of 200 random words we appended to the real prompt, (5) write in Japanese subject-object-verb word order ("Yoda grammar").

As a sixth obfuscation, we increased the temperature of DeepSeek R1 to a value of 1.1, the highest that still produced sensible text.

Finally, we also paraphrased already-generated texts from GPT-4o, GPT-4.5, and DeepSeek R1 using the adversarial DIPPER paraphraser [20]. Unfortunately, we were unable to get a satisfactory text quality from this. We used the implementation and pre-trained models provided by the authors, yet the paraphraser often generated long strings of repeated white space and punctuation marks and sometimes degenerated entirely into seemingly random tokens. We got this result with both the context-aware and the context-unaware model and with all sampling parameters we tested. To improve the quality, we tried to clean up at least some of the repeated characters using a regular expression.

### 3.1.4. Dataset Sampling and Splits

From all generated texts, we randomly sampled a subset with a varying balance of human to LLM texts. We sampled from the Gutenberg set with a human:LLM ratio of 1:1, from Brennan-Greenstadt, Riddell-Juola, and PAN'24 training with a 1:4 ratio, and PAN'24 test with a 1:1 ratio.

The sampled texts were split into training, validation, and test sets. Texts created by OpenAI o1 and o1-mini texts were held back and included only in the test set. This was inspired by our earlier findings [5] that OpenAI o1 was much harder to detect when detectors were trained only on other LLMs. For this reason, we also created new o1 texts from the PAN'24 news article summaries. The texts provided by ELOQUENT participants were later added to the test set without balancing them. We kept track of the generating models and the individual source datasets, so we could later distinguish between different subsets of the full test set during the evaluation phase. For this reason, we distinguish between the terms *dataset*, which refers to all data from all sources, and *data subset*, which refers to texts from an individual source. The subsets were chosen so that each subset contains both classes

(with the exception of the OpenAI o1 and DIPPER-obfuscated Gutenberg subsets), which is crucial for measures such as ROC-AUC and $F_1$. A detailed listing of all subsets, splits, and class balances is given in Table 3.

### 3.2. Baselines

We provide three baselines: Two zero-shot baselines and one supervised. Baseline (1) is an implementation of a Binoculars [21] model using Llama-3.1-8b and Llama-3.1-8b-instruct. The decision threshold was optimized for high accuracy (not for a low false-positive rate as in the original paper) on the validation set that was handed out to participants. Baseline (2) is a simple PPMd-based compression model using the compression-based cosine measure [22, 23]. The operating point of this detector was also tuned on the validation set. Lastly, as a supervised baseline, (3) we provide a linear SVM trained on the top-1000 TF-IDF 1–4-grams from the validation set. The TF-IDF detector and Binoculars can be considered state of the art; the compression model represents a more conservative lower baseline.

### 3.3. Submissions

We received submissions from 24 teams, 21 of which also submitted a work notebook describing their approach. The following section gives an overview of them with a short description for each. Table 4 summarizes the use of certain feature families and methods employed by the submitted systems.

**Basani and Chen [36]** use GPT-2 to estimate mean, variance, skewness, and kurtosis of the negative per-token log likelihood (token "surprisal") distributions of a text. In addition to these stationary statistical moments, they also calculate their first- and second-order rates of change. An XGBoost classifier is learned on these features for distinguishing between human and LLM texts.

**Huang et al. [40]** fine-tune a RoBERTa model for binary classification, but does extensive data augmentation work. The existing training data is modified via synonym replacement and sentence reorganization, and noise is added via random repetitions or stop and fill word insertions. In addition, new generated documents are added using models not included in the training data, where different prompts are used to generate text across different genres to roughly double the number of generated training texts.

**Jimeno-Gonzalez et al. [32]** use stylometric and word-frequency features with a stacking ensemble of Random Forest, XGBoost, and LinearSVC, where a logistic regression classifier casts the ensemble vote. Specifically, the system uses the function-word ratio, average tf-idf scores, mean sentence length, mean word length, number of sentences, POS-tag distribution, and word frequency features over the vocabulary.

**Kumar et al. [42]** fine-tune a DistilBERT for binary classification on the provided training data. As features, the system uses the base model's CLS embedding concatenated with five stylometric features: average word length, average sentence length, punctuation frequency, type-token ratio, and character entropy.

**Larson [39]** aim to be especially light-weight and use a Support Vector Machine with RBF kernel and class weighting for binary classification. As features, the system uses the 40 most frequent 1- and 2-grams, along with the 15 most frequent punctuation features, selected based on performance on the validation set.

**Liang et al. [43]** fine-tune a ModernBERT for binary classification and use a custom loss function weighted by an example's difficulty.

**Liu et al. [25]** use an ensemble of a fine-tuned Qwen and a fine-tuned ModernBERT with contrastive loss to distinguish between human and LLM-generated text. The dataset was augmented by generating LLM paraphrases of the human texts.

**Table 4**
Overview of the kind of features and methods used in submitted systems. **Features:** Contextualized embeddings (Embed.), LLM-based text perplexity estimates (PPL), term frequency features (TF), other stylometric / linguistic features (Style). **Methods:** Ensembling methods (Ensemble), training / validation data augmentation (Data Aug.), specialized training routines or loss functions (Loss).

| | Features | | | | Methods | | |
|---|---|---|---|---|---|---|---|
| **Team** | Embed. | PPL | TF | Style | Ensemble | Data Aug. | Loss |
| Macko [24] | × | | | | | × | |
| Valdez-Valenzuela | × | | | × | | × | |
| Liu [25] | × | | | | × | × | × |
| Seeliger [26] | | | × | | | | |
| Voznyuk [27] | × | | | | | | × |
| Yang [28] | × | | | | | | × |
| Zaidi [29] | × | | | | | | |
| *Baseline TF-IDF SVM* | | | | | | | |
| Marchitan [30] | × | | | | × | | |
| Teja [31] | × | | | | × | | |
| Jimeno-Gonzalez [32] | | | × | × | × | | |
| Völpel [33] | | | | × | | | |
| Ochab [34] | | | | × | | × | |
| Ostrower [35] | × | | | | | | × |
| Basani [36] | | × | | | | | |
| Pudasaini [37] | × | | | | × | | |
| Sun [38] | | × | | × | × | | |
| Larson [39] | | | × | × | | | |
| Huang [40] | × | | | | | × | |
| Titze [41] | | × | | | | | |
| *Baseline Binoculars Llama3.1 [21]* | | | | | | | |
| Kumar [42] | × | | | × | | | |
| *Baseline PPMd Compression-based Cosine [23, 22]* | | | | | | | |
| Liang [43] | × | | | | | | × |
| Sum of Systems 2024 | 20 | 11 | 1 | 5 | 5 | 6 | – |
| Sum of Systems 2025 | 13 | 3 | 3 | 7 | 6 | 5 | 5 |
| Fraction of Systems 2024 | **0.67** | **0.37** | 0.03 | 0.17 | 0.17 | 0.20 | – |
| Fraction of Systems 2025 | 0.62 | 0.14 | **0.14** | **0.33** | **0.29** | **0.24** | **0.24** |

**Macko [24]** fine-tune a Qwen3-14B model via QLoRA for binary classification. What makes the approach work is the obfuscation via a homoglyph (replacement of characters with similar-looking ones Unicode characters) attack of parts of the training data (a variant of training data enhancement) and model selection on external training data to select the best model based on out-of-domain performance. The external training data is a collection of 2,000 examples across seven languages, sampled from 18 different AI-detection datasets of different genres and domains [44].

**Marchitan et al. [30]** describe two system: First, a voting-ensemble of LightGBM, XGBoost, Logistic Regression, and an SVM using embeddings from Qwen3-0.6B, and, second, fine-tuning a classification head on top of an LLM, where Qwen2.5-0.5B, Qwen3-0.6B, Mistral7B-v0.1, and Llama-3.1-8B were tested.

**Ochab et al. [34]** use an LGBM classifier on four types of stylometric features extracted using spaCy and inspired by the "stylo" R package. The dataset was augmented by adding about half a million texts from other AI detection datasets. 10-fold cross-validation was used to select the best hyperparameters for the model.

**Ostrower et al. [35]** propose three systems. The first system is a XGBoost ensemble using the Binoculars score, tf-idf scores, and BERT embeddings as features. The second, not submitted system uses a Maximum Likelihood Estimation based on various features: cohesiveness, type-token ratio, word count, stop-word frequency, non-sentiment word frequency, POS distribution. Cohesiveness is computed via the average BARTScore (between a text and multiple noisy, obfuscated copies) and multiple zero-shot LLM detectors (LRR, FastDetectGPT). The third approach uses adversarial training following Hu et al. [45].

**Pudasaini et al. [37]** use various ensembling strategies (voting, stacking, and boosting) with a number of fine-tuned pre-trained language models (such as DeBERTa, Longformer, RoBERTa, etc.). The results were tested on the PAN'25 dataset and the COLING'25 dataset.

**Teja et al. [31]** fine-tune several pre-trained language models (such as DeBERTa, DistilBERT, XLNet, and others) with a mixture-of-experts gating mechanism. They tested a hard gating mechanism which selects only a single experts and a soft gating mechanism which uses a linear layer with softmax to select a weighted sum of expert outputs. A DeBERTa model with hard gating performed best on the validation set and was submitted to Tira.

**Seeliger et al. [26]** generate a matrix of cumulative binary correlation coefficients between terms and documents. Three different versions were submitted using word unigrams, bigrams, and trigrams as terms. The authors also supplied two baselines, one of which is a fine-tuned RoBERTa model. The approach allows for stationary analysis of the whole document, but also temporal analysis of the cumulative sum of correlation coefficients per word.

**Sun et al. [38]** use a combination of 25 stylometric and 25 entropy-based features and a voting ensemble of five different classifiers (Gaussian Naive Bayes, AdaBoost, LightGBM, CatBoost, Random Forest). The stylometric features selected via univariate feature selection from 101 features suggested by the Claude LLM. The entropy-based features are selected via univariate feature selection from 72 statistical features describing the distribution of per-token forward and backward cross-entropy losses, following Guo et al. [46]. These losses are computed by "regenerating" a training text via teacher forcing, given a generated summary as prompt, and measuring the loss of each predicted output logits to the target (forward) and to the last token of the prefix (backward), according to a Llama2-7b.

**Titze and Halvani [41]** use off-the-shelf pre-trained LLMs to extract negative log likelihood "surprisal", mean Shannon entropy, log rank, Jensen-Shannon divergence from the token representation of the text. A logistic regression classifier is learnt to combine the features into a final score.

**Valdez-Valenzuela** generate a syntactic dependency graph representation of the text. Sentence-level graphs are merged into document-level graphs, which are embedded using a graph neural network for use in a dense neural network for classification. The dataset is augmented with three different kinds of obfuscations (shortening, Unicode replacement, paraphrasing) to make the system more robust.

**Völpel and Halvani [33]** use linearized constituent trees n-grams as features for a feed-forward neural network classifier. The trees are parsed via the Constituent Treelib library [47] and segmented by traversing from every node to all possible leaves and counting all encountered paths of length 1-7.

**Voznyuk et al. [27]** use multi-task learning with DeBERTa-v3 base. The system learns three tasks, each utilizing a different head: the binary Voight-Kampff task, a 3-class genre prediction task, and a 4-class model family prediction task. The model is then trained for all tasks, albeit only some losses are propagated to the base model to prevent overfitting. The results of the Voight-Kampff classification head are reported for the competition.

**Yang and Yan [28]** fine-tune a BERT model using genre-dependent contrastive loss. For this, the CLS output token is concatenated with a learned genre vector, which is then fed into an MLP with contrastive loss.

**Zaidi et al. [29]** fine-tune an uncased BERT-base for binary classification of human and LLM texts.

**Table 5**
Final leaderboard for subtask 1. The systems are ranked by the macro-average mean of all metrics over individual datasets in the main test collection. The mean score itself is a macro average over the datasets as well, so the columns do not necessarily sum up to that value. Teams that submitted multiple systems are listed only once with their best-performing system. False positive / negative rates are listed for reference, but are not part of the overall mean score and thus do not contribute to the ranking.

| | Team | ROC-AUC | Brier | C@1 | $F_1$ | $F_{0.5u}$ | Mean ↓ | FPR | FNR |
|---|---|---|---|---|---|---|---|---|---|
| 1. | Macko [24] | 0.995 | 0.984 | 0.982 | 0.989 | 0.993 | **0.989** | 0.006 | 0.018 |
| 2. | Valdez-Valenzuela* | 0.939 | 0.902 | 0.897 | 0.926 | 0.960 | **0.929** | 0.020 | 0.107 |
| 3. | Liu [25] | 0.962 | 0.891 | 0.889 | 0.923 | 0.963 | **0.928** | 0.005 | 0.120 |
| 4. | Seeliger [26] | 0.912 | 0.898 | 0.896 | 0.930 | 0.959 | **0.925** | 0.082 | 0.103 |
| 5. | Voznyuk [27] | 0.899 | 0.898 | 0.898 | 0.929 | 0.962 | **0.924** | 0.035 | 0.107 |
| 6. | Yang [28] | 0.930 | 0.893 | 0.886 | 0.920 | 0.960 | **0.923** | 0.018 | 0.122 |
| 7. | Zaidi [29] | 0.931 | 0.891 | 0.887 | 0.924 | 0.958 | **0.922** | 0.062 | 0.115 |
| 8. | *hello-world** | 0.963 | 0.900 | 0.897 | 0.904 | 0.946 | **0.922** | 0.106 | 0.093 |
| | *Baseline TF-IDF SVM* | 0.963 | 0.900 | 0.897 | 0.904 | 0.946 | **0.922** | 0.106 | 0.093 |
| 9. | *bohan-li** | 0.951 | 0.888 | 0.884 | 0.919 | 0.952 | **0.922** | 0.052 | 0.115 |
| 10. | Marchitan [30] | 0.945 | 0.890 | 0.869 | 0.905 | 0.952 | **0.916** | 0.011 | 0.142 |
| 11. | Teja [31] | 0.897 | 0.881 | 0.881 | 0.916 | 0.958 | **0.914** | 0.005 | 0.129 |
| 12. | *xlbniu** | 0.883 | 0.875 | 0.875 | 0.910 | 0.950 | **0.907** | 0.032 | 0.132 |
| 13. | Jimeno-Gonzalez [32] | 0.941 | 0.873 | 0.849 | 0.892 | 0.943 | **0.901** | 0.029 | 0.162 |
| 14. | Völpel [33] | 0.922 | 0.881 | 0.849 | 0.892 | 0.936 | **0.899** | 0.084 | 0.151 |
| 15. | Ochab [34] | 0.904 | 0.886 | 0.846 | 0.891 | 0.933 | **0.897** | 0.124 | 0.150 |
| 16. | Ostrower [35] | 0.872 | 0.854 | 0.854 | 0.896 | 0.943 | **0.891** | 0.041 | 0.151 |
| 17. | Basani [36] | 0.904 | 0.864 | 0.843 | 0.894 | 0.943 | **0.891** | 0.084 | 0.160 |
| 18. | Pudasaini [37] | 0.900 | 0.858 | 0.844 | 0.890 | 0.937 | **0.891** | 0.077 | 0.159 |
| 19. | Sun [38] | 0.903 | 0.877 | 0.843 | 0.883 | 0.933 | **0.890** | 0.087 | 0.152 |
| 20. | Larson [39] | 0.830 | 0.863 | 0.863 | 0.910 | 0.935 | **0.885** | 0.234 | 0.116 |
| 21. | Huang [40] | 0.864 | 0.827 | 0.848 | 0.906 | 0.869 | **0.870** | 1.000 | 0.000 |
| 22. | Titze [41] | 0.854 | 0.853 | 0.808 | 0.869 | 0.926 | **0.865** | 0.131 | 0.196 |
| | *Baseline Binoculars Llama3.1 [21]* | 0.827 | 0.856 | 0.818 | 0.866 | 0.907 | **0.863** | 0.263 | 0.173 |
| 23. | Kumar [42] | 0.591 | 0.826 | 0.826 | 0.888 | 0.873 | **0.831** | 0.729 | 0.057 |
| | *Baseline PPMd CBC [23, 22]* | 0.644 | 0.802 | 0.759 | 0.817 | 0.839 | **0.790** | 0.797 | 0.137 |
| 24. | Liang [43] | 0.734 | 0.694 | 0.694 | 0.752 | 0.827 | **0.751** | 0.157 | 0.298 |

* No notebook submitted.

## 3.4. Evaluation

The final system ranking based on the PAN and ELOQUENT test sets is listed in Table 5. Participating teams that submitted more than one system are ranked only with their best-performing system.

All systems were submitted and evaluated on Tira [7]. At test time, the participants had to calculate a score between 0 and 1 for each text, indicating the likelihood that the text was LLM-generated. A score of exactly 0.5 could be given to signal a non-decision.

### 3.4.1. Score Calculation

For each participant, we computed a confusion matrix and the following scores, which we used in previous authorship verification shared tasks as well:

- ROC-AUC: The area under the Receiver Operating Characteristic curve.
- Brier: The complement of the Brier score (mean squared loss)
- C@1: A modified accuracy score that assigns non-answers (score = 0.5) the average accuracy of the remaining cases [48].
- $F_1$: The harmonic mean of precision and recall.

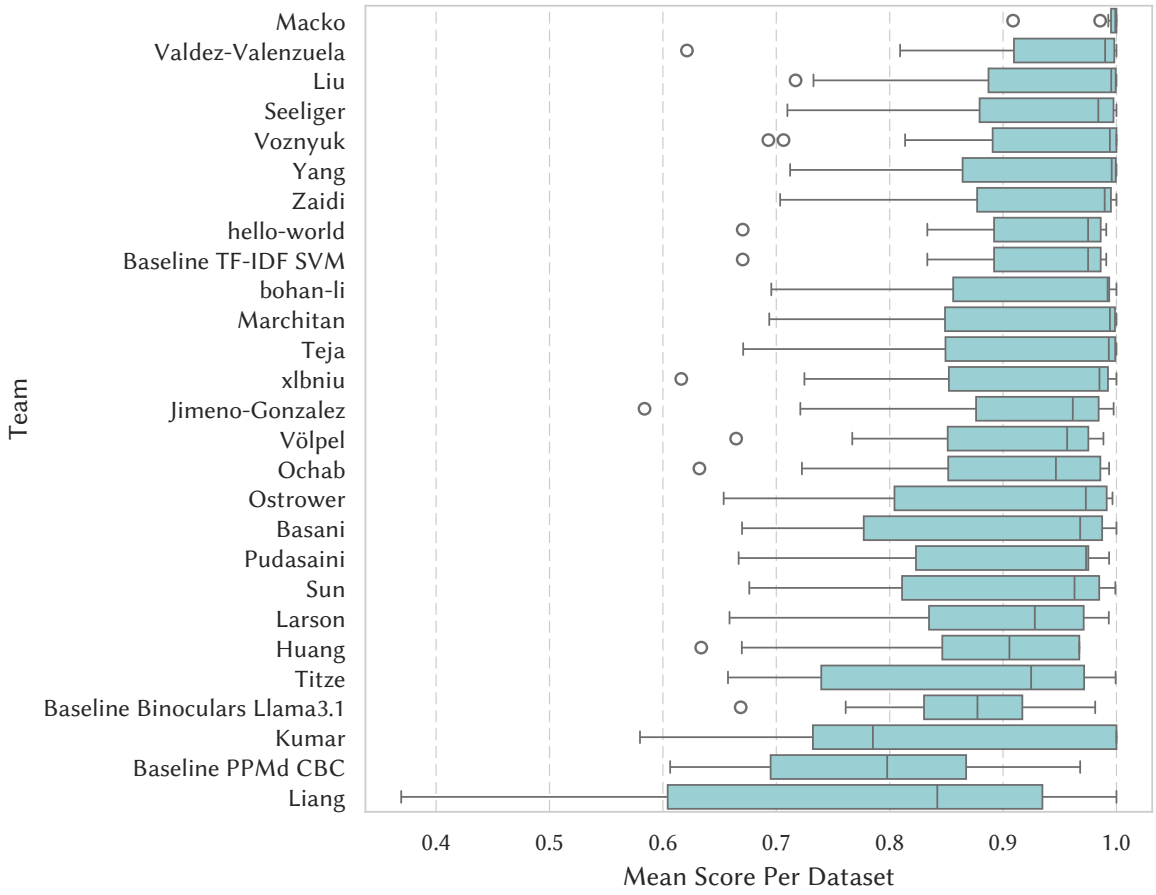


**Figure 4:** Distribution of mean scores over the test data subsets from which the final macro mean score in Table 5 is calculated. The best-performing system shows remarkable consistency across all data subsets, whereas most other systems have significantly larger variance.

- $F_{0.5u}$: A modified $F_{0.5}$ measure (precision-weighted F measure) that treats non-answers (score = 0.5) as false negatives [49].
- Mean: The arithmetic mean of all previous measures

The six measures are calculated for each system on each of the test data subsets described in Section 3.1.4. A final system score is calculated per measure as a macro average over all subsets. The resulting macro mean score is used to determine the system's ranking in the final leaderboard. The mean scores for data subsets with only positive examples (i.e., the held-back OpenAI o1 and DIPPER-obfuscated Gutenberg subsets) is calculated without the contribution of ROC-AUC, as the value would be undefined. The precision for the $F_1$ and $F_{0.5u}$ calculations is assumed to be 1.0 in those cases.

### 3.4.2. Final Ranking and Discussion

Table 5 shows the final macro scores per team and system.[2] If teams submitted multiple systems, only the best-performing system is considered in the ranking. Of the 24 submitted systems, 8 beat the strongest baseline (TF-IDF SVM) and 14 more beat the second-strongest baseline (Binoculars). Macko [24] lead the ranking with Valdez-Valenzuela (no notebook) and Liu et al. [25] being the runners-up.

Figure 4 shows the distribution of mean scores for each system over test data subsets from which the macro means are calculated. Most systems range between 0.7 and 1.0 for most subsets, apart

[2] *Note: Bevendorff et al. [16] and an earlier draft of this paper described significantly worse scores for all systems. This was due to a score calculation error on the ELOQUENT data subset, which has been corrected in this version.*

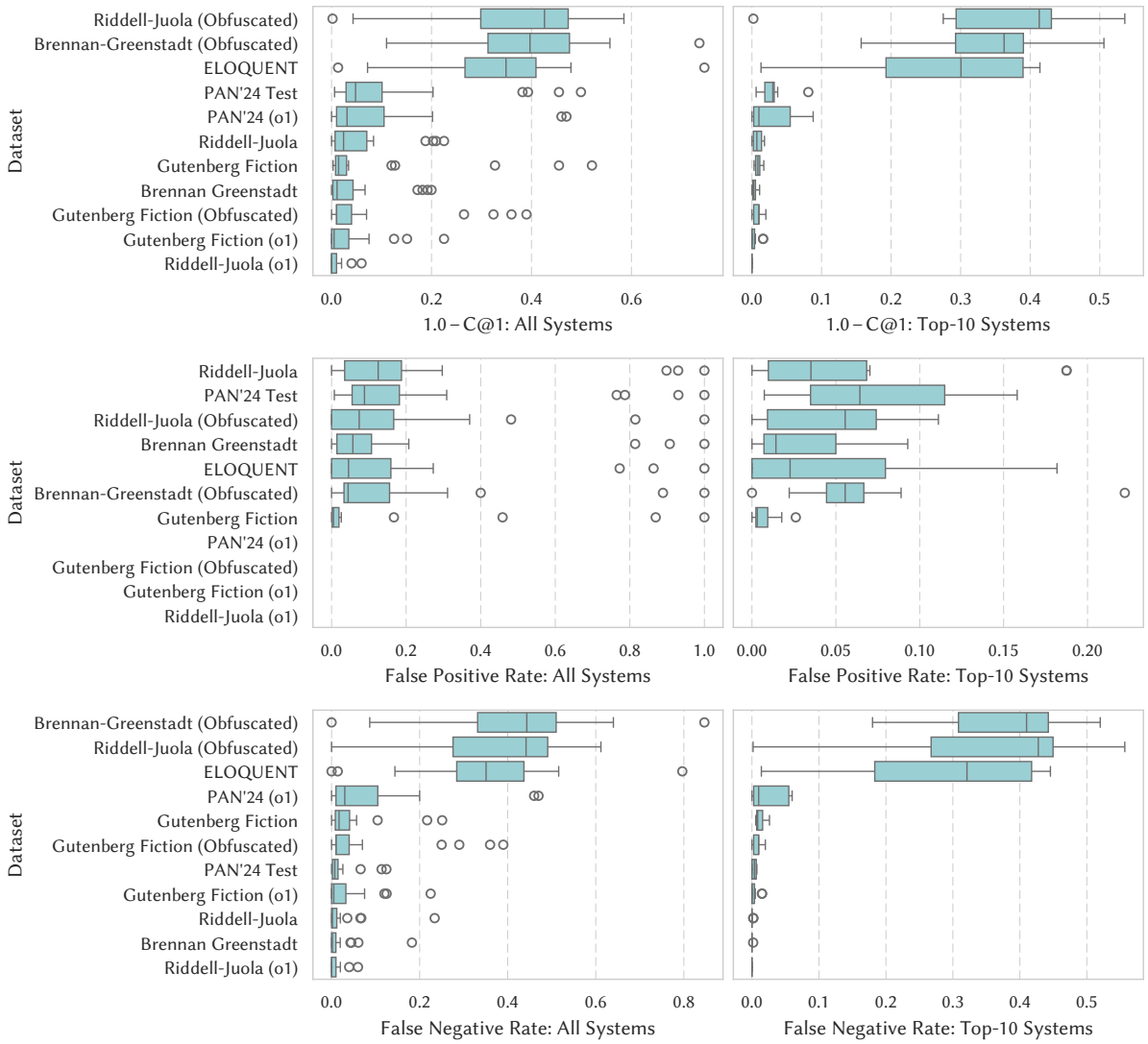


**Figure 5:** Difficulty of individual test data subsets as the inverse of C@1 scores and the corresponding false positive / false negative rates (positive being LLM-generated). No false positive rate is shown for the DIPPER-obfuscated Gutenberg and the OpenAI o1 subsets, since these contain only positive examples.

from certain outliers. On the ELOQUENT subset, the majority of systems achieve scores between 0.7 and 0.8, dragging down their final scores. The best system by Macko [24] is an exception to this rule by displaying more than solid performance also on ELOQUENT. The mean false positive rate of all systems on ELOQUENT's human texts is 0.15, which is probably best explained by the topic domain shift from the PAN training data. Cross-domain classification has been a longstanding problem in authorship identification [50], but also in LLM detection in particular [51].

If we analyze the distribution C@1 scores per subset (Figure 5), we can see the obfuscated Brennan-Greenstadt and Riddell-Juola and ELOQUENT subsets seem to be the most difficult. However, if we look at false positive and false negative rates separately, we can find this effect only in the latter. That means LLMs were more successful than humans in obfuscating their distinct "LLM style," whereas the obfuscated human texts still look human enough. This is not entirely unexpected, since the human authors were only asked to mask their personal style, not to "write like a machine" (and LLMs did not exist back then, anyway). The LLMs, on the other hand, apparently managed to deviate sufficiently from their usual style that makes them otherwise stand out. This effect, however, must also be attributed in large part to the other obfuscations we applied. Due to the way the test set was sampled, only parts of these subsets are generated from the plain neighborhood prompt (57 % for Brennan-Greenstadt, 18 % for Riddell-Juola). Hence, further analysis is needed to distinguish between these two effects. Since

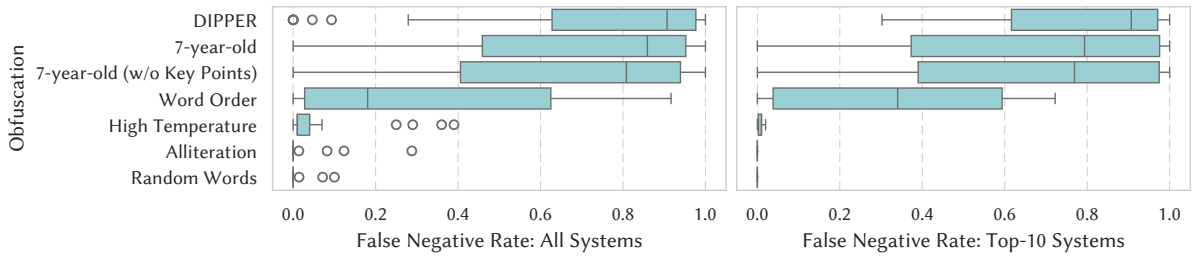


**Figure 6:** Effectiveness of text obfuscations (see Section 3.1.3). DIPPER paraphrasing and prompting the model to write like a 7-year-old seem to be the most effective. In fact, they seem to be even more effective in otherwise more accurate models, likely due to domain overfitting. Note that the boxes may not be fully comparable, since the underlying source texts or summaries are not always the same.

these obfuscations are in part also purely prompt-based, we can assume, though, that LLMs did manage to change their style sufficiently. Considering that the human texts were classified correctly for the most part, the topic / domain shift alone cannot serve as an explanation.

The Gutenberg data, on the other hand, seems to be the easiest, which may indeed be a result of the text homogeneity between training and test. The held-back OpenAI o1 texts also posed no particular challenge. This is in contrast to our previous findings [5]. This is even true for the TF-IDF SVM baseline, which is identical to the detector we used in those earlier experiments. This peculiarity definitely warrants more research. Perhaps the inclusion of GPT-4.5 texts in the training data played a role, but perhaps the included sample of only 300 texts was too small.

Figure 6 shows the effectiveness of individual obfuscations (not including the plain neighborhood prompt) described in Section 3.1.3. DIPPER paraphrasing appears to be highly effective in avoiding detection, but this comes at a heavy price with regard to text quality. Hence, this result should be taken with a grain of salt. The second-most effective obfuscation appears to be the prompt to write like a 7-year-old. This obfuscation uses only the prompt to change the model output, which appears to be sufficient to successfully avoid detection in most cases. Prompting the model to change the word order to subject-object-verb shows a medium effectiveness. All three are even more effective if tested against only the top-10 systems, hinting at overfitting. Increasing the temperature, prompting for alliterations, or inserting random words into the prompt did not have any significant impact.

The overall best performing system by Macko [24] had a false-negative rate of 42 % on DIPPER-paraphrased texts, but was entirely unaffected by all other obfuscations.

## 4. PAN Subtask 2: Human-AI Collaboration

The rise of AI-assisted writing is transforming the traditional notion of authorship and steadily blurring the boundary between human and machine contributions. As AI involvement now spans the full spectrum—from none to complete [52], new questions of ethical and intellectual accountability arise. Subtask 2 seeks to address these challenges by asking the participants to classify a collaboratively authored human-AI document into one of six categories, defined from an ethical and intellectual accountability perspective:

i. fully human-written;
ii. human-written, then machine-polished;
iii. machine-written, then machine-humanized (obfuscated);
iv. human-initiated, then machine-continued;
v. deeply mixed text, where some parts are written by a human and some are generated by a machine;
vi. machine-written, then human-edited.

**Table 6**
Subtask 2 training, development and test set distribution across six categories.

| Label | Text Category | Train | Dev | Test |
|---|---|---|---|---|
| 0 | Fully human-written | 75,270 | 12,330 | 34,509 |
| 1 | Human-written, then machine-polished | 95,398 | 12,289 | 43,154 |
| 2 | Machine-written, then machine-humanized | 91,232 | 10,137 | 25,234 |
| 3 | Human-initiated, then machine-continued | 10,740 | 37,170 | 22,802 |
| 4 | Deeply-mixed text | 14,910 | 225 | 12,500 |
| 5 | Machine-written, then human-edited | 1,368 | 510 | 2,557 |
| **Total** | | 288,918 | 72,661 | 140,756 |

## 4.1. Dataset

The training and validation sets for Subtask 2 were constructed from existing datasets focused on fine-grained machine-generated text detection. We provided 288,918 examples for training and 72,661 for validation. For testing, we collected four new datasets, with an emphasis on student essays and peer reviews. We additionally incorporated five recently released datasets to comprehensively evaluate the generalization of detection systems across unseen generators and domains. The test set consists of 140,756 instances. The dataset statistics, including the distribution across six categories, are shown in Table 6. All subsets underwent the cleaning procedure: we removed duplicates and filtered out texts shorter than 30 characters. Below, we describe each component dataset in detail, with an overview provided in Table 7.

### 4.1.1. Training and Development Sets

**MixSet [53]** includes machine-polished human-written texts and human-edited machine-generated texts. We sampled 3,491 out of 3,600 texts, ensuring a minimum text length of 30 characters.

**LLM-DetectAIve [54]** is a large-scale benchmark designed to improve machine-generated text detection across different domains and text variations. The dataset is based on M4GT [55], which includes a mix of human-written and machine-generated texts from sources such as arXiv, WikiHow, Wikipedia, Reddit, student essays (OUTFOX), and peer reviews (PeerRead). LLM-DetectAIve extends the M4GT dataset by adding new machine-generated texts from more advanced models (e.g., GPT-4o), machine-humanized MGTs, and human-written texts that were polished using LLMs. The resulting dataset comprises 91,358 fully machine-generated texts, 103,852 machine-humanized texts, and 107,900 human-polished texts. LLM-DetectAIve includes outputs from a diverse set of LLMs, including Llama3-8B/70B, Mixtral-8x7B, Gemma-7B, Gemma-2-9B, GPT-4o, Gemini-1.5-Pro, and Mistral-7B. For Subtask 2, we excluded the fully machine-generated texts and used 286,169 examples from the remaining two categories.

**RoFT [56]** is a collection of over 21k human annotations paired with error classifications to investigate how various variables such as model size, decoding strategy, and fine-tuning affect human detection performance. The dataset is in English and contains domains like recipes, presidential speeches, short stories, and New York Times. Each instance is initiated by a human and then continued by either a human or an LLM including GPT-2 [68], GPT-3 [69] and CTRL [70]. We preprocess the data by performing basic cleaning, removing duplicates and sanity entries. This results in a total of 9,148 instances that are either fully human-written or human-initiated and machine-continued.

**RoFT-ChatGPT [57]** is an augmented version of the Real or Fake Text (RoFT) dataset. The augmented version includes GPT-3.5-turbo generations. The dataset is entirely in English and covers domains such as short stories, recipes, New York Times news articles, and presidential speeches. It consists of 6,940 samples of text that are human-initiated and machine-continued, offering a rich resource for evaluating the interaction between human authorship and AI-generated text.

**Table 7**
**Subtask 2 Component Datasets** used for training, development and test sets. Size refers to the number of examples we sampled from a given dataset, rather than the full set of original dataset. In addition to leveraging existing datasets, we collected three datasets based on ICNALE, NLPeer and Peersum, and we initiated a dataset MBZUAI-CLEF based on our real-life usage.

| Dataset | Data License | Original Task | Types | Domains | LLMs | Size |
|---|---|---|---|---|---|---|
| **Training and Development Sets** | | | | | | |
| MixSet [53] | CC BY 4.0 | Multiway | ii, iv, v, vi | email, news, game reviews, paper abstracts, speech, blog | GPT-4, Llama2, Dolly | 3,491 |
| DetectiAIve [54] | MIT License | 4-class | i, ii, iii | arXiv research paper abstracts, Reddit posts, Wikihow and Wikipedia articles, OUTFOX essays, peer reviews | GPT-4o, mistral-7b, Llama3-8b, Llama3-70b, Gemini, Cohere | 286,169 |
| M4GT [55] | CC BY 4.0 | Boundary Detection | i, iv | peer review, OUTFOX essays | Llama2, GPT-4, GPT-3.5 | 31,928 |
| Real or Fake [56] | MIT License | Boundary Detection | i, iv | Recipes, Presidential Speeches, Short Stories, New York Times News articles | GPT-2, GPT-2-XL, CTRL | 9,148 |
| RoFT-ChatGPT [57] | CC BY 4.0 | Boundary Detection | iv | Recipes, Presidential Speeches, Short Stories, New York Times News articles | GPT-3.5-turbo | 6,940 |
| Co-author [58] | CC BY 4.0 | Boundary Detection | v | creative writing, argumentative writing (New York Times) | GPT-3 | 1,447 |
| TriBERT [59] | CC BY 4.0 | Boundary Detection | i, iv, v | essay in education | GPT-3.5-turbo | 21,178 |
| LAMP [60] | BSD 3-Clause License | Multi-Span Categorical Extraction | vi | creative writing | GPT-4o, Llaude3.5-sonnet, Llama3-70B | 1,282 |
| **Test Set** | | | | | | |
| Beemo [61] | MIT License | Multiway | i, iii, vi | generation, rewrite, open QA, summarization, closed QA | Llama2-70B, Llama3.1, GPT-4o, Zephyr, Mixtral, Tulu, Gemma | 17,331 |
| APT-Eval [62] | MIT License | Binary Classification (AI-Polished Text) | ii | email, news, game reviews, paper abstracts, speech, blog (300 human text in MixSet) | GPT-4o, Llama3.1-70B, Llama3-8B, Llama2-7B | 11,389 |
| HART [63] | MIT License | Fine-grained MGT | i, ii, iii, vi | news, arXiv, essay, writing | GPT-3.5-turbo, GPT-4o, Claude-3.5-Sonnet, Gemini-1.5-pro, Llama-3.3-70B-instruct, Qwen-2.5-72B-instruct | 23,999 |
| LLMDetect [64] | MIT License | Fine-grained MGT | i, iii, iv | N24News, Guardian news articles | Deepseek-v2, Llama3-70B-Instruct, Claude-3.5-Sonnet, GPT-4o | 48,216 |
| ALTA-2024 MGT [65] | MIT License | Fine-grained MGT | iv, v | essay in education, news | LlaMA 3.1-8B-Instruct | 16,574 |
| ICNALE | CC BY-NC 4.0 | Fine-grained MGT | i, ii, iii, iv | essays written by authors with different English proficiency | GPT-4.1, Qwen3-30B-a3b, Llama-3.3-70B, Deepseek-chat-v3, Gemini-2.5-pro-preview | 10,123 |
| Peersum [66] | CC BY 4.0 | Fine-grained MGT | i, ii, iii, iv | academic peer review | GPT-4o, GPT-4.1, GPT-4.1-mini, Deepseek-R1 | 5,270 |
| NLPeer [67] | Apache License 2.0 | Fine-grained MGT | ii, iii | academic peer review | GPT-4.1, o4-mini | 5,993 |
| MBZUAI-CLEF | Our collection | Fine-grained MGT | i, ii, vi | NLP paper, student report, rebuttal, peer review, admin letters, email, daily chat/message, slides summary, oral presentation, github readme | GPT-4o | 1,115 |

**Co-author [58]** detected sentence-level boundaries of human-AI collaborative mixed texts. They identified challenges in detecting AI-generated sentences in mixed texts, such as human writers' selecting and even editing AI-generated sentences based on personal preferences; the frequent change of authorship between neighboring sentences within the mixed text; and the short length of text segments within mixed texts, which provides on limited stylistic cues for reliable authorship determination.

Coauthor comprises 1,447 writing sessions produced through the collaboration of human writers, recruited from Amazon Mechanical Turk, and a GPT-3 assistant [69]. These sessions encompass both creative and argumentative writing. Writers were provided with a prompt as a starting point and could either compose independently or request sentence suggestions from the assistant. The suggested sentences could be accepted or rejected, and this iterative process continued until the essay was completed. This setup results in a highly mixed text, where some parts are authored by humans and others are generated by the machine.

**TriBERT [59]** is a dataset of human-AI collaborative writing. First, the authors collected the human-written essays from U.S.junior high school students (grades 7-10). They then designed 8 text modification

tasks, each involving the removal of specific segments from the original human-written texts, followed by LLM-generated fill-in. This approach resulted in 8 distinct forms of human-AI hybrid writing. In our six-label space, texts that start with a human-written introduction and then were completed by LLMs are mapped as *human initiated, machine continued*, and all other cases are classified as *deeply mixed*. This transformation doubles, resulting in 34,272 texts, and then we sampled 21,178.

**LAMP [60]** consists of 1,057 paragraphs generated by LLMs and subsequently edited by professional writers. The original paragraphs were sourced from reputable publication venues, including The New Yorker, The New York Times, and Dear Sugar. These paragraphs cover various domains, such as fiction, food writing, and internet advice. In total, approximately 1,200 paragraphs were selected, with the Literary Fiction genre accounting for 80% and the remaining 20% categorized as Creative Non-Fiction.

The instructions for generating these paragraphs were created using instruction back-translation [71]. In this process, GPT-4o was prompted to summarize each paragraph into writing instructions. After manual verification, ill-formed or overly specific instructions were filtered out, resulting in a total of 1,057 high-quality instructions. These instructions were then used to generate additional writings using GPT-4o, Claude-3.5-Sonnet, and Llama 3.1-70b. Finally, the LLM-generated responses were further refined and edited by a team of 18 professional writers, following a comprehensive edit taxonomy informed by expert writing practices.

#### 4.1.2. Test Set

**Beemo** [61] Benchmark of Expert-edited Machine-generated Outputs (Beemo) is a dataset designed to support fine-grained detection of machine-generated text (MGT), particularly in multi-author scenarios where LLM outputs are refined by either human experts or other LLMs. It comprises a total of 19.6k English texts spanning five tasks: open-ended generation, rewriting, summarization, and open/closed QA. We used 17,331 examples, removing the purely machine-generated texts. Specifically, this includes 2,184 human-written texts, 2,183 machine-written, then human-edited texts, with the rest as machine-written, then machine-humanized.

**AI-Polished-Text Evaluation (APT-Eval)** Based on the 300 purely human-written texts sampled from MixSet [53], Saha and Feizi [62] collected 11.7K machine-polished data using four different models: GPT-4o, Llama3.1-70B, Llama3-8B, and Llama2-7B with two polishing strategies. For degree-based polishing, the LLM is prompted to refine the text in four varying degrees of modification (1) extremely-minor, (2) minor, (3) slightly-major, and (4) major. For percentage-based polishing, the LLM is instructed to modify a fixed percentage (p%) of words in a given text. The percentage is systematically varied across the following values: p% = 1, 5, 10, 20, 35, 50, 75%. They find that detectors have a higher misclassification rate over smaller and older generators.

**HART [63]** is a dataset for fine-grained machine-generated text detection, including categories of purely human-written, human-written and then machine-polished, machine-generated and then machine-humanized, purely machine-generated text. There are 16K English examples, and 4K examples for the other four languages including Chinese, Arabic, French, and Spanish. Six LLM generators were involved in data collection, i.e., GPT-3.5-turbo, GPT-4o, Claude-3.5-Sonnet, Gemini-1.5-pro, Llama-3.3-70B-instruct, and Qwen-2.5-72B-instruct. They additionally gathered 250 machine-generated human-edited texts. This dataset can be used as part of our test set.

**LLMDetect [64]** is designed to distinguish between four categories of text origin: Human-Author (fully human-written), LLM-Creator (entirely machine-generated), LLM-Polisher (human-written text subsequently refined by an LLM), and LLM-Extender (human-written text extended or expanded by an LLM). LLMDetect includes the Hybrid News Detection Corpus (HNDC) for training detectors, as well as DetectEval, a comprehensive evaluation suite that considers five distinct cross-context variations and two multi-intensity variations within the same LLM role. This allows for a thorough assessment of detectors' generalization and robustness across diverse contexts.

**Table 8**
Domain / Task categories of MBZUAI-CLEF.

| Category | Representative Tasks |
|---|---|
| Academic Writing | academic paper/abstract writing, paper summary, peer review, rebuttal, |
| Professional & Academic Correspondence | Email writing, administrative letters, recommendation requests, ethical response letters, commitment letters |
| Application Materials & Career Development | PhD SOP review, teaching statements, job interview preparation, oral interview explanations, volunteer applications |
| Creative & Informal Writing | Story writing, daily chatting, continued writing |
| Technical & Support Writing | GitHub issues, README files, report writing, NLP slides summary |
| Prompt Engineering | Prompt refining |

**ALTA 2024 Shared Task Mollá et al. [65]** employ four distinct construction patterns to organize human and machine-generated sentences: (1) human-written sentences followed by machine-generated sentences; (2) machine-generated sentences followed by human-written sentences; (3) human-written sentences, then machine-generated sentences, followed again by human-written sentences; and (4) machine-generated sentences, then human-written sentences, followed by machine-generated sentences.

#### 4.1.3. Our Datasets

We collected three datasets based on ICNALE[3], Peersum, and NLPeer, and we further gathered a dataset from scratch based on authors' daily usage of LLMs.

**International Corpus Network of Asian Learners of English (ICNALE)** is an international learner corpus. We sampled 5,843 topic-controlled essays produced by more than 5,500 college students (incl. grad students) in ten countries/regions in Asia (China, Hong Kong, Taiwan, Indonesia, Japan, Korea, Pakistan, Philippines, Singapore, Malaysia, and Thailand), as well as English native speakers. Based on these human-written essays, we further generated 2,624 human-written, then machine-polished, 1,000 machine-written, then machine-humanized, and 656 human-initiated, then machine-continued text, using the latest SOTA LLM including GPT-4.1, Qwen3-30B-a3b, Llama-3.3-70B, Deepseek-chat-v3, Gemini-2.5-pro-preview, resulting in a total of 10,123 examples.

**Peersum [66]** is a dataset designed for generating meta-reviews of scientific papers based on reviews. The meta-reviews can be interpreted as abstractive summaries of reviews, multi-turn discussions among reviewers and the paper author, and the paper abstract. We sampled 1,000 human-written meta-reviews and the corresponding reviews. We used GPT-4.1-2025-04-14, GPT-4.1-mini-2025-04-16, and Deepseek-R1 to further produce 2,000 human-written, then machine-polished, 2,000 human-initiated, then machine-continued, and 270 machine-written, then machine-humanized cases, in total of 5,270.

**NLPeer [67]** is intended for the study of peer review and approaches to NLP-based assistance to peer review. Based on human-written reviews, we used GPT-4.1 and GPT-4.1-mini for human-written, then machine-polished (2,993). Based on either paper abstracts or the full papers, we first applied GPT-4o to generate reviews, and used GPT-4.1 and GPT-4.1-mini to humanize (3,000).

**MBZUAI-CLEF** is a dataset we collected from scratch. The human-written texts cover six categories including academic writing, professional correspondence, application materials, creative writing, technical and support writing, and prompting engineering, with 378 examples in total, as shown in Table 8. These human-written drafts were polished using GPT-4o, after which the authors further edited the polished outputs to adapt them to real-world applications.

[3] https://language.sakura.ne.jp/icnale/

## 4.2. Task Organization

The Subtask 2 was conducted in two phases:

**Development Phase.** Only training and development data were provided to participants, with gold labels available for the development set. Although they were not allowed to use external training and validation data, data augmentation strategies such as back-translation, synonym replacement, random word deletion, and replacement were allowed.

Participants competed against each other to achieve the best performance on the development set. A live leaderboard on CodaLab was made available to track all submissions.[4] Teams could make an unlimited number of submissions, and the best score for each team, regardless of the submission time, was displayed in real time on CodaLab.

**Test Phase.** Participants were given approximately one week to prepare their predictions. Then, participant teams submitted their results to CodaLab.[5] They could submit multiple runs, but they would not receive feedback on their performance. Only the latest submission from each team was considered "official," and was used for the final team ranking. In total, 22 teams submitted results, of which 16 submitted system description papers. After the competition concluded, we released the gold labels for the test set. Furthermore, we kept the submission system open for the test dataset for post-shared task evaluations and to monitor the state of the art.

## 4.3. Evaluation

Predictions of all systems were submitted and evaluated in CodaLab. At the test time, participants assigned the predicted label among [0, 1, 2, 3, 4, 5] for each text, indicating its category. Participants in the leaderboard were ranked by *macro-recall*. Macro-recall is selected as the primary evaluation metric for two reasons: (1) it gives equal importance to each class, preventing performance for majority classes from dominating the overall score on an unbalanced test set; and (2) macro-recall provides a more focused view on the model's ability to capture all positive instances for every class, compared with macro-F1 balancing precision and recall for each class. As additional evaluation metrics, we also computed accuracy and macro-F1.

## 4.4. Baseline

To establish a baseline, we fine-tuned a pre-trained transformer-based model RoBERTa on the training set. Fine-tuning was performed using the Hugging Face `Trainer` API with the following configuration: learning rate of $2 \times 10^{-5}$, batch size of 16 for both training and evaluation, weight decay of 0.1, and a total of 3 training epochs. Checkpoints were evaluated at the end of each epoch, and the best-performing model on the development set was retained for subsequent testing. The baseline achieved a macro-recall of 68.67 % on the development set, with corresponding macro-F1 and accuracy scores of 61.26 % and 56.71 %, respectively.

## 4.5. Submissions

22 teams submitted their predictions to CodaLab, of which 16 submitted notebook papers [24, 26, 30, 29, 27, 31, 72, 76, 77, 74, 79, 81, 78, 75, 73, 80]. The performance of 14 teams is above the baseline, and 8 teams are below fine-tuned RoBERTa-base, as shown in Table 9. One team submitted their prediction file after the deadline, which we marked in gray. Additionally, four teams submitted files with IDs misaligned with the test set. The test set contains IDs ranging from 0 to 141,410, with 655 entries filtered out, rather than forming a continuous range from 0 to 140,755. For fairness, we ranked all submissions based on their original versions. However, we also corrected the misaligned IDs and re-evaluated the

[4] https://codalab.lisn.upsaclay.fr/competitions/22620
[5] https://codalab.lisn.upsaclay.fr/competitions/22934

**Table 9**
Subtask 2 evaluation results of 22 submissions, ranking by macro-recall, along with macro-F1 and accuracy, with one delayed submission in gray. Teams ranked 17 to 21 submitted files with misaligned test set ids. The test set ids range from 0 to 141,410 with 655 filtered out, rather than being continuous from 0 to 140,755. This misalignment led to an underestimation of their system performances. We present their corrected scores in green in the updated ranking.

| | Team | Rec. | F1 | Acc. | System Description |
|---|---|---|---|---|---|
| 1. | mdok [24] | 64.46 | 65.06 | 74.09 | QLoRa PEFT fine-tuned Qwen3-4B-Base. |
| 2. | Bohan Li [72] | 61.72 | 61.73 | 69.28 | Under-sample high-frequency classes and adopt data augmentation for underrepresented classes, along with R-Drop regularization for DeBERTa-v3-base fine-tuning. |
| – | *lza* | 60.61 | 61.43 | 70.15 | – |
| 3. | Advacheck [27] | 60.16 | 60.85 | 69.04 | Shared Transformer Encoder between several classification heads trained to distinguish the domains. |
| 4. | StarBERT [73] | 57.46 | 56.31 | 66.81 | Combine the deep language understanding of DeBERTa-v3-large and the high-dimensional mapping ability of StarBlock2d. |
| 5. | Atu [74] | 56.87 | 56.45 | 66.30 | DeBERTa enhanced by contextual and geometric attention |
| 6. | TaoLi [75] | 56.74 | 55.39 | 66.27 | Use DeBERTa-v3-Large |
| 7. | ReText.Ai [76] | 56.11 | 55.25 | 64.79 | Fine-tune Gemma-2 2B for sequence classification with multiple classification heads. |
| – | *hkkk* | 56.01 | 55.79 | 66.48 | – |
| 8. | DetectTeam [77] | 54.49 | 54.40 | 62.89 | Fine-tune DeBERTa-V3-Large and combining multi-scale features. |
| – | *NanMu* | 54.39 | 53.63 | 64.62 | – |
| 9. | WeiDongWu [78] | 54.09 | 53.57 | 63.01 | Combine the contextual strength of BERT with the sequence modeling capabilities of Transformer layers. |
| 10. | zhangzhiliang [79] | 54.06 | 52.81 | 61.65 | Fine-tune DeBERTa-V3-Large and combine it with BiLSTM and attention mechanism. |
| 11. | CNLP-NITS-PP [31] | 54.05 | 53.49 | 62.23 | Soft and Hard Mixture of Experts (MoE) architectures with DeBERTa-V3-Large |
| 12. | a.dusuki | 52.83 | 51.44 | 60.45 | – |
| 13. | Steely [26] | 52.14 | 51.81 | 59.88 | Cumulative sum of token-Level correlation signals |
| 14. | a.elnenaey | 49.56 | 50.10 | 58.96 | – |
| | *Baseline* | 48.32 | 47.82 | 57.09 | Fine-tune RoBERTa |
| 15. | VerbaNex AI [80] | 47.15 | 47.15 | 56.24 | Fine-tune RoBERTa with class balancing, data augmentation, and calculation of specific weights for each unbalanced class. |
| – | *johanjthomas* | 45.30 | 43.94 | 53.42 | – |
| 16. | Unibuc-NLP [30] | 44.33 | 42.76 | 51.42 | Combine features at different layers extracted using Transformers with layer-wise projection and attentive pooling. |
| | *Nexus Interrogators [29]* | 33.86 | 31.86 | 35.45 | Fine-tune transformer models with data augmentation strategies on underrepresented classes *(late submission).* |
| 17. | johanjthomas | 33.71 | 31.63 | 37.85 | – |
| 18. | lza | 32.90 | 31.98 | 33.20 | – |
| 19. | NanMu | 32.87 | 31.79 | 34.52 | – |
| 20. | hkkk | 32.79 | 31.95 | 34.21 | – |
| 21. | YoussefAhmed21 | 16.48 | 14.98 | 21.22 | – |

affected submissions to better reflect the actual performance of their detection systems. The following analysis is based on these corrected results.

Many teams fine-tuned DeBERTa-v3-large and achieved better results than RoBERTa. Larger language models such as Qwen-3 4B and Gemma-2 2B were superior to DeBERTa and RoBERTa, see more in Table 10. The performance drop observed on the test set compared to the development set highlights the need for further improvement in fine-grained human-AI collaborative text detection.

**Table 10**
**Subtask 2 Participant System Overview.** MoE: Mixture of Experts, Geom. Attn: Geometric Attention, Multi-aug: Multiple augmentation techniques (back-translation, synonym/antonym replacement, random deletion). Features Comb.: Combination of different features.

| Team Name | Ranking | Small PLM | Large PLM | Fine-tuning | Data Aug. | Features Comb. | Ensemble | Mult. Heads | Model Details |
|---|---|---|---|---|---|---|---|---|---|
| mdok | 1 | | ✓ | ✓ | | | | | Qwen3-4B (QLoRA PEFT) |
| Bohan Li | 2 | ✓ | | ✓ | ✓ | | | | DeBERTa-v3-base (R-Drop) |
| Advacheck | 3 | ✓ | | ✓ | | ✓ | | ✓ | DeBERTa-base (Multi-task) |
| StarBERT | 4 | | ✓ | ✓ | | | | | DeBERTa-V3-Large (Hybrid) |
| Atu | 5 | | ✓ | ✓ | | | | | DeBERTa-V3-Large (Geom. Attn) |
| TaoLi | 6 | | ✓ | ✓ | | | | | DeBERTa-V3-Large |
| ReText.Ai | 7 | | ✓ | ✓ | | | | ✓ | Gemma-2 2B (Multi-head) |
| DetectTeam | 8 | | ✓ | ✓ | | ✓ | | | DeBERTa-V3-Large (Multi-scale) |
| WeiDongWu | 9 | ✓ | | ✓ | | | | | BERT_T |
| zhangzhiliang | 10 | | ✓ | ✓ | | | | | DeBERTa-V3-Large + BiLSTM |
| CNLP-NITS-PP | 11 | | ✓ | ✓ | | | ✓ | | DeBERTa-V3-Large (MoE) |
| Steely | 13 | | | ✓ | | ✓ | | | Token-level correlation signals |
| VerbaNex AI | 15 | ✓ | | ✓ | ✓ | | | | RoBERTa-base (Class weight) |
| Unibuc-NLP | 16 | ✓ | | ✓ | | | | | RoBERTa-base (Layer proj.) |
| Nexus Interrogators | - | | ✓ | ✓ | ✓ | | | | RoBERTa-large |
| Sentence-BERT [81] | - | ✓ | | ✓ | | | | | BERT-base (Sent. segment) |

### 4.5.1. System Overview

The highest-ranking system, mdok [24], fine-tuned the Qwen3-4B-Base model using QLoRA for efficient parameter adaptation. In comparison, Bohan Li [72] used DeBERTa-v3-base enhanced with data augmentation, balancing, and R-Drop regularization, showing strong performance despite a smaller backbone.

Several teams built upon DeBERTa-V3-Large for human-AI collaborative writing detection. StarBERT [73], Atu [74], TaoLi [75], and Zhangzhiliang [79] all fine-tuned this model, each introducing unique extensions: StarBERT proposed a hybrid classification framework, Atu incorporated contextual and geometric attention, TaoLi focused on fine-grained categorization, and Zhangzhiliang added a BiLSTM layer with attention mechanisms. Advacheck [27] also used DeBERTa-base within a multi-task learning framework to distinguish between human, machine, and hybrid authorship styles.

DetectTeam [77] enhanced DeBERTa-V3-Large with a Feature Pyramid Network to capture multi-scale features, while CNLP-NITS-PP [31] employed the Mixture-of-Experts (MoE) architectures with SoftMoE and HardMoE for dynamic model routing.

Other teams opted for RoBERTa-based architectures. VerbaNex [80] fine-tuned RoBERTa-base and addressed data imbalance using augmentation, oversampling, undersampling, and loss weighting. Nexus [29] fine-tuned RoBERTa-Large, applying targeted data augmentation techniques including back-translation, synonym/antonym substitution, and random deletion.

ReText.Ai [76] adopted a multi-head classification strategy over the Gemma-2 2B model, while WeiDongWu [78] introduced BertT, a hybrid model combining BERT-base with an extra Transformer encoder and dropout layers for six-class collaborative classification.

Unibuc-NLP [30] merged transformer layers using projection and attentive pooling. Steely [26] converted text into interpretable word-level correlation signals. Fuchuan [81] applied a BERT-based model for sentence-level classification on segmented inputs.

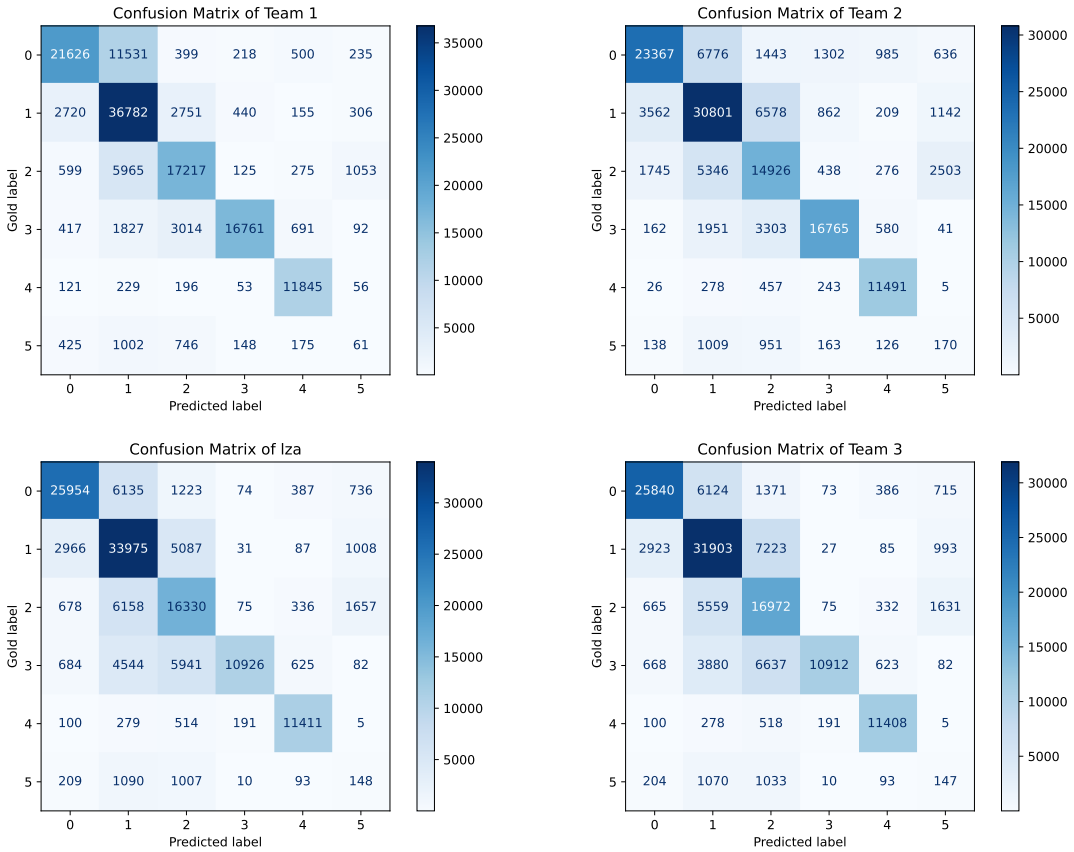


**Figure 7:** Confusion Matrix of Top3 Team and Team lza.

### 4.5.2. Recall Across Labels

We show the macro-averaged recall and label-wise recall scores of 22 submissions for Subtask 2 in Table 11. A clear trend emerges in terms of label difficulty. Label 4 (deeply-mixed text) consistently receives the highest recall across systems, indicating it is the easiest class to detect. This is followed by label 1 (human-written, then machine-polished) and label 0 (purely human-written), which also achieve high recall scores for most systems. In contrast, label 5 (machine-written, then human-edited) proves to be the most challenging, with most systems performing poorly on this category, and only a few submissions (e.g., team 7) achieving a recall above 9.0 %.

This trend partially aligns with the label distribution in the training data. The easiest labels 1 (human-written, then machine-polished) and 0 (purely human-written) are among those with the highest number of training examples. In contrast, labels 5 (machine-written, then human-edited) and 3 (human-initiated, then machine-continued) are the most challenging to detect, which also corresponds to their small training sizes. While label 2 (machine-written, then machine-humanized) has a comparable amount of training data to label 1 (91k vs. 95k), its performance is significantly worse across most systems.

The confusion matrices in Figure 7 further confirm this pattern. For all top-performing systems (Teams 1–3 and lza), label 2 is frequently misclassified as 1. This suggests that machine-humanized texts (label 2) share stylistic cues with machine-polished texts (label 1) likely due to the dominating influence of machine-generated text in both. Conversely, label 5 exhibits low recall and high confusion with label 2, indicating that human edits often obscure the origin of machine-generated text, making it particularly difficult to detect.

Taken together, these findings underscore the nuanced difficulty of distinguishing between varying degrees of human-AI collaboration. While abundant data helps, the boundary between polished and humanized content remains inherently fuzzy, calling for more fine-grained modeling approaches beyond token-level patterns or surface style cues.

**Table 11**
Subtask 2 evaluation results of 22 submissions: recall across six labels, along with the macro-avg recall.

| Rank | Macro-Avg | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|---|
| 1 | **64.46** | 62.67 | 85.23 | 68.23 | 73.51 | **94.76** | 2.39 |
| 2 | 61.72 | 67.71 | 71.37 | 59.15 | **73.52** | 91.93 | 6.65 |
| lza | 60.61 | 75.21 | 78.73 | 64.71 | 47.92 | 91.29 | 5.79 |
| 3 | 60.16 | 74.88 | 73.93 | 67.26 | 47.86 | 91.26 | 5.75 |
| 4 | 57.46 | 77.00 | 73.58 | **72.74** | 25.96 | 90.58 | 4.89 |
| 5 | 56.87 | 75.22 | 74.85 | 69.17 | 26.48 | 91.74 | 3.75 |
| 6 | 56.74 | 75.46 | 72.83 | 70.52 | 29.25 | 90.30 | 2.07 |
| 7 | 56.11 | **87.74** | 68.92 | 58.88 | 20.85 | 90.45 | **9.82** |
| hkkk | 56.01 | 79.56 | 79.00 | 57.16 | 27.14 | 90.77 | 2.42 |
| 8 | 54.49 | 61.42 | 73.95 | 67.58 | 30.40 | 90.82 | 2.78 |
| NanMu | 54.39 | 78.16 | 77.30 | 55.98 | 22.60 | 90.43 | 1.84 |
| 9 | 54.09 | 62.84 | 75.73 | 71.45 | 21.34 | 90.98 | 2.23 |
| 10 | 54.06 | 58.44 | 72.58 | 72.60 | 24.23 | 90.22 | 6.30 |
| 11 | 54.05 | 67.54 | 69.99 | 64.26 | 27.60 | 91.97 | 2.93 |
| 12 | 52.83 | 56.29 | 76.34 | 64.16 | 22.28 | 90.02 | 7.90 |
| 13 | 52.14 | 51.16 | 72.74 | 68.91 | 28.76 | 90.12 | 1.17 |
| 14 | 49.56 | 62.89 | 81.21 | 48.89 | 14.57 | 83.38 | 6.41 |
| *Baseline* | 48.32 | 42.76 | **85.90** | 56.03 | 14.74 | 87.62 | 2.85 |
| 15 | 47.15 | 49.18 | 83.31 | 49.47 | 13.48 | 84.89 | 2.58 |
| johanjthomas | 45.30 | 40.99 | 81.97 | 50.99 | 8.40 | 86.46 | 2.97 |
| 16 | 44.33 | 30.81 | 79.83 | 52.21 | 15.36 | 84.19 | 3.56 |
| 21 | 16.48 | 17.23 | 27.45 | 40.00 | 3.55 | 9.02 | 1.60 |

**Table 12**
Subtask 2 evaluation of 22 submissions: accuracy across nine component datasets in the testbed.

| Rank | Beemo | APT-Eval | HART | LLMDetect | ALTA | ICNALE | Peersum | NLPeer | MBZUAI-CLEF |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 46.12 | 75.90 | **81.95** | 73.15 | **90.97** | **80.99** | 53.43 | 96.90 | 39.64 |
| 2 | 44.64 | 62.83 | 53.67 | **80.09** | 90.03 | 70.17 | 53.24 | 92.27 | 35.61 |
| lza | 47.87 | 64.67 | 66.98 | 73.52 | 88.22 | 74.48 | 52.20 | 94.98 | 40.99 |
| 3 | 50.10 | 64.07 | 64.27 | 73.37 | 88.19 | 74.10 | 39.28 | 87.85 | 40.81 |
| 4 | **56.50** | 64.96 | 60.97 | 63.44 | 86.80 | 75.09 | 54.10 | 96.30 | **47.71** |
| 5 | 51.03 | 64.42 | 64.73 | 61.69 | 88.19 | 73.58 | **56.51** | **97.23** | 44.66 |
| 6 | 52.15 | 60.21 | 65.87 | 62.09 | 86.68 | 76.83 | 54.52 | 94.91 | 44.04 |
| 7 | 43.17 | 53.00 | 63.27 | 64.75 | 87.55 | 72.09 | 53.53 | 92.01 | 46.46 |
| hkkk | 40.75 | 66.44 | 66.76 | 65.13 | 85.77 | 77.42 | 53.19 | 94.11 | 44.13 |
| 8 | 50.45 | 68.61 | 59.16 | 54.72 | 87.86 | 72.08 | 55.07 | 96.18 | 44.93 |
| NanMu | 39.32 | 66.12 | 64.05 | 62.48 | 84.43 | 76.47 | 52.28 | 93.86 | 43.23 |
| 9 | 53.54 | 65.98 | 67.07 | 50.35 | 86.76 | 74.02 | 55.39 | 97.05 | 44.57 |
| 10 | 54.38 | 68.35 | 59.58 | 50.12 | 86.26 | 72.45 | 54.88 | 96.85 | 41.70 |
| 11 | 50.23 | 66.76 | 60.38 | 53.15 | 88.87 | 71.97 | 49.22 | 94.58 | 40.54 |
| 12 | 45.58 | 65.48 | 61.08 | 53.48 | 85.34 | 60.84 | 50.27 | 93.43 | 36.32 |
| 13 | 56.20 | 73.11 | 65.64 | 43.41 | 85.34 | 63.88 | 49.41 | 93.66 | 31.21 |
| 14 | 39.17 | 66.06 | 59.13 | 52.81 | 80.77 | 63.31 | 53.74 | 90.41 | 42.51 |
| baseline | 42.27 | 83.90 | 57.44 | 43.97 | 84.40 | 63.70 | 43.49 | 87.19 | 33.27 |
| 15 | 41.47 | 73.11 | 62.35 | 44.73 | 81.73 | 54.30 | 43.87 | 83.25 | 36.41 |
| johanjthomas | 41.58 | **84.30** | 59.62 | 37.56 | 76.45 | 50.20 | 48.14 | 83.53 | 39.91 |
| 16 | 43.64 | 77.97 | 51.88 | 37.42 | 81.43 | 37.55 | 41.02 | 90.67 | 35.07 |
| 21 | 32.32 | 27.51 | 28.24 | 15.91 | 7.74 | 21.11 | 17.76 | 33.79 | 15.34 |

#### 4.5.3. Accuracy Across Datasets

The system performance across the nine component datasets in the testbed is shown in Table 12. The top-ranked systems achieve the highest accuracies on datasets such as NLPeer (peer review), ALTA (essay), and ICNALE (essay), which consist primarily of academic and well-structured text genres. These datasets likely provide clearer linguistic signals and stylistic features for distinguishing different human-AI collaborative texts. For instance, the best-performing system achieves 90.97 % accuracy on ALTA and 96.90 % on NLPeer. These results suggest that model generalization is strongest in formal writing settings, particularly in educational and peer-review contexts.

In contrast, performance drops significantly on datasets such as MBZUAI-CLEF, Peersum, and Beemo, which contain more diverse genres and informal domains. MBZUAI-CLEF, for example, covers a wide range of real-world writing scenarios, including rebuttals, admin letters, emails, oral presentations, and GitHub README files, but consists of only 1,115 examples. All systems perform poorly on this dataset, with accuracies below 48 %, highlighting the difficulty of detecting fine-grained signals in low-resource, noisy, or mixed-genre settings. Similarly, Peersum and Beemo include peer review and open-domain generation tasks, both of which challenge the systems due to variability in style, structure, and intent.

We also observe that datasets with larger size and consistent formatting, such as LLMDetect (48k examples), HART (24k), and ALTA (16.5k), tend to support more stable performance across submissions. This indicates that both data volume and stylistic consistency facilitate more effective fine-tuning and generalization. The wide variance in performance across component datasets underscores the importance of robustness and adaptability in collaborative text detection. Future work should prioritize developing detection systems capable of generalizing across low-resource, diverse, and informal domains, where style-based cues may be weaker or inconsistent.

## 5. Conclusion

The PAN Subtask 1 received submissions from 24 teams, of which 21 also submitted a work notebook. Most of the submitted systems were quite strong, with the best system being almost perfect on the PAN test data. However, unlike in 2024, the texts submitted by ELOQUENT participants posed a difficult challenge for PAN systems. In part, this can be explained by the topic shift between the ELOQUENT dataset and the training data provided by PAN, but (almost) all systems also struggled with certain obfuscations applied to the texts, including ones that relied solely on prompting techniques without changing the actual topic.

The PAN Subtask 2 introduced a nuanced, six-category classification of human-AI collaboration, reflecting the complex reality of modern text production and arising ethical and intellectual accountability challenges. While participants' systems could distinguish certain categories like "deeply-mixed text" with high recall, they struggled significantly to identify "machine-written, then human-edited" texts—a crucial and challenging real-world scenario. Performance was heavily influenced by the distribution of training data and the domain of the text, with systems performing better on structured academic writing than on more diverse, informal datasets. The difficulty of distinguishing between machine-polished and machine-humanized text further underscores the fuzzy boundaries in collaborative writing. Together, these findings indicate that while progress has been made, the reliable detection of AI authorship, especially in its more subtle and varied forms, remains an open and pressing challenge.

## Acknowledgments

The work of Janek Bevendorff, Matti Wiegmann, Maik Fröbe, Martin Potthast, and Benno Stein on PAN subtask 1 has been funded as part of the OpenWebSearch project by the European Commission (OpenWebSearch.eu, GA 101070014).

## Declaration on Generative AI

Text, datasets, experiments, and analyses in this paper were created and conducted by the authors themselves. Generative AI was used by some authors for assistance in the writing process, but no substantial parts were generated by it.

## References


[1] J. Bevendorff, I. Borrego-Obrador, M. Chinea-Ríos, M. Franco-Salvador, M. Fröbe, A. Heini, K. Kredens, M. Mayerl, P. Pęzik, 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, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. 14th International Conference of the CLEF Association (CLEF 2023), volume 14163 of *Lecture Notes in Computer Science*, Springer, Berlin Heidelberg New York, 2023, pp. 459–481. doi:10.1007/978-3-031-42448-9_29.

[2] J. Bevendorff, B. Chulvi, E. Fersini, A. Heini, M. Kestemont, K. Kredens, M. Mayerl, R. Ortega-Bueno, P. Pezik, 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: Experimental IR Meets Multilinguality, Multimodality, and Interaction. 13th International Conference of the CLEF Association (CLEF 2022), volume 13186 of *Lecture Notes in Computer Science*, Springer, Berlin Heidelberg New York, 2022. doi:10.1007/978-3-031-13643-6.

[3] J. Bevendorff, B. Chulvi, G. L. D. L. P. Sarracén, M. Kestemont, E. Manjavacas, I. Markov, M. Mayerl, M. Potthast, F. Rangel, P. Rosso, E. Stamatatos, B. Stein, M. Wiegmann, M. Wolska, E. Zangerle, Overview of PAN 2021: Authorship Verification, Profiling Hate Speech Spreaders on Twitter,and Style Change Detection, in: 12th International Conference of the CLEF Association (CLEF 2021), Springer, 2021. URL: https://doi.org/10.1007/978-3-030-85251-1_26.

[4] J. Bevendorff, B. Ghanem, A. Giachanou, M. Kestemont, E. Manjavacas, I. Markov, M. Mayerl, M. Potthast, F. Rangel, P. Rosso, G. Specht, E. Stamatatos, B. Stein, M. Wiegmann, E. Zangerle, Overview of PAN 2020: Authorship Verification, Celebrity Profiling, Profiling Fake News Spreaders on Twitter, and Style Change Detection, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. 11th International Conference of the CLEF Initiative (CLEF 2020), volume 12260 of *Lecture Notes in Computer Science*, Springer, Berlin Heidelberg New York, 2020, pp. 372–383. doi:10.1007/978-3-030-58219-7_25.

[5] J. Bevendorff, M. Wiegmann, E. Richter, M. Potthast, B. Stein, The Two Paradigms of LLM Detection: Authorship Attribution vs. Authorship Verification, in: The 63rd Annual Meeting of the Association for Computational Linguistics (ACL 2025) (Findings), Association for Computational Linguistics, 2025.

[6] 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. García Seco de Herrera (Eds.), Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2024, pp. 2486–2506. URL: http://ceur-ws.org/Vol-3740/paper-225.pdf.

[7] 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. URL: https://link.springer.com/chapter/10.1007/978-3-031-28241-6_20. doi:10.1007/978-3-031-28241-6_20.

[8] Y. Wang, J. Mansurov, P. Ivanov, J. Su, A. Shelmanov, A. Tsvigun, O. M. Afzal, T. Mahmoud,

G. Puccetti, T. Arnold, SemEval-2024 Task 8: Multidomain, Multimodel and Multilingual Machine-Generated Text Detection, in: Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), 2024, pp. 2057–2079.

[9] Y. Wang, A. Shelmanov, J. Mansurov, A. Tsvigun, V. Mikhailov, R. Xing, Z. Xie, J. Geng, G. Puccetti, E. Artemova, J. Su, M. N. Ta, M. Abassy, K. A. Elozeiri, S. E. D. A. El Etter, M. Goloburda, T. Mahmoud, R. V. Tomar, N. Laiyk, O. M. Afzal, R. Koike, M. Kaneko, A. F. Aji, N. Habash, I. Gurevych, P. Nakov, GenAI Content Detection Task 1: English and Multilingual Machine-Generated Text Detection: AI vs. Human, in: F. Alam, P. Nakov, N. Habash, I. Gurevych, S. Chowdhury, A. Shelmanov, Y. Wang, E. Artemova, M. Kutlu, G. Mikros (Eds.), Proceedings of the 1stWorkshop on GenAI Content Detection (GenAIDetect), International Conference on Computational Linguistics, Abu Dhabi, UAE, 2025, pp. 244–261. URL: https://aclanthology.org/2025.genaidetect-1.27/.

[10] S. Saha, R. Das, D. Das, JUNLP_SS at ELOQUENT Lab 2025: Humanizing AI - Enhancing the Realism of Machine Generated Text, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), 26th Working Notes of the Conference and Labs of the Evaluation Forum, CLEF 2025, CEUR-WS, 2025.

[11] A. Creo, M. Hormazábal-Lagos, H. Cerezo-Costas, P. Alonso-Doval, HumanAIzers in Voight-Kampff at ELOQUENT 2025, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), 26th Working Notes of the Conference and Labs of the Evaluation Forum, CLEF 2025, CEUR-WS, 2025.

[12] P. Vachharajani, Literal Re-translation as a Method for AI Text Disguise and Detection Evasion, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), 26th Working Notes of the Conference and Labs of the Evaluation Forum, CLEF 2025, CEUR-WS, 2025.

[13] R. R. Gunti, The Data-Centric Approach for the Voight Kampff Task, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), 26th Working Notes of the Conference and Labs of the Evaluation Forum, CLEF 2025, CEUR-WS, 2025.

[14] M. Hoveyda, Bypassing Human/Machine Classifiers by Prompting LLMs for Naturally Imperfect Text, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), 26th Working Notes of the Conference and Labs of the Evaluation Forum, CLEF 2025, CEUR-WS, 2025.

[15] 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: Experimental IR Meets Multilinguality, Multimodality, and Interaction. 15th International Conference of the CLEF Association (CLEF 2024), volume 14959 of *Lecture Notes in Computer Science*, Springer, Berlin Heidelberg New York, 2024, pp. 231–259. doi:10.1007/978-3-031-71908-0_11.

[16] J. Bevendorff, D. Dementieva, M. Fröbe, B. Gipp, A. Greiner-Petter, J. Karlgren, M. Mayerl, P. Nakov, A. Panchenko, M. Potthast, A. Shelmanov, E. Stamatatos, B. Stein, Y. Wang, M. Wiegmann, E. Zangerle, Overview of PAN 2025: Voight-Kampff Generative AI Detection, Multilingual Text Detoxification, Multi-Author Writing Style Analysis, and Generative Plagiarism Detection, in: J. C. de Albornoz, J. Gonzalo, L. Plaza, A. G. S. de Herrera, J. Mothe, F. Piroi, P. Rosso, D. Spina, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Sixteenth International Conference of the CLEF Association (CLEF 2025), Lecture Notes in Computer Science, Springer, Berlin Heidelberg New York, 2025.

[17] J. Karlgren, E. Artemova, O. Bojar, V. Mikhailov, M. Sahlgren, E. Velldal, L. Øvrelid, ELOQUENT CLEF shared tasks for evaluation of generative language model quality, 2025 edition, in: Advances in Information Retrieval: 47th European Conference on Information Retrieval, ECIR 2025, Lucca, Italy, April 6–10, 2025, Proceedings, Part V, Springer-Verlag, Berlin, Heidelberg, 2025, pp. 366–372. URL: https://doi.org/10.1007/978-3-031-88720-8_56. doi:10.1007/978-3-031-88720-8_56.

[18] M. Brennan, S. Afroz, R. Greenstadt, Adversarial stylometry: Circumventing authorship recognition to preserve privacy and anonymity, ACM Transactions on Information and System Security 15 (2012). URL: http://dx.doi.org/10.1145/2382448.2382450. doi:10.1145/2382448.2382450.

[19] H. Wang, P. Juola, A. Riddell, Reproduction and replication of an adversarial stylometry experiment,

arXiv [cs.CL] (2022). URL: http://arxiv.org/abs/2208.07395. arXiv:2208.07395.

[20] K. Krishna, Y. Song, M. Karpinska, J. Wieting, M. Iyyer, Paraphrasing evades detectors of AI-generated text, but retrieval is an effective defense, in: A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine (Eds.), Advances in Neural Information Processing Systems 36 (NeurIPS 2023), volume 36, 2023, pp. 27469–27500. URL: https://proceedings.neurips.cc/paper_files/paper/2023/hash/575c450013d0e99e4b0ecf82bd1afaa4-Abstract-Conference.html.

[21] 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, in: Forty-first International Conference on Machine Learning, {ICML} 2024, Vienna, Austria, July 21-27, 2024, volume abs/2401.12070, OpenReview.net, 2024, pp. 17519–17537. URL: https://dl.acm.org/doi/10.5555/3692070.3692768. doi:10.48550/arXiv.2401.12070.

[22] D. Sculley, C. E. Brodley, Compression and machine learning: A new perspective on feature space vectors, in: Data Compression Conference (DCC'06), IEEE, 2006, pp. 332–341. URL: https://ieeexplore.ieee.org/abstract/document/1607268?casa_token=EwORidpcgTwAAAAA:zONgvLu6aVgw-jrz0A-5JHXs-SdAljLqNXabhAQh6w685CRwAXXe7FxcD67SDkf6Ztfaj6AEwWA. doi:10.1109/dcc.2006.13.

[23] O. Halvani, C. Winter, L. Graner, On the usefulness of compression models for authorship verification, in: Proceedings of the 12th International Conference on Availability, Reliability and Security, volume Part F1305, ACM, New York, NY, USA, 2017, pp. 54:1–54:10. URL: http://dx.doi.org/10.1145/3098954.3104050. doi:10.1145/3098954.3104050.

[24] D. Macko, mdok of KInIT: Robustly Fine-tuned LLM for Binary and Multiclass AI-Generated Text Detection, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[25] J. Liu, L. Kong, Z. Peng, F. Chen, Generative AI Authorship Verification Based on Contrastive-Enhanced Dual-Model Decision System, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[26] M. Seeliger, P. Styll, M. Staudinger, A. Hanbury, Human or Not? Light-Weight and Interpretable Detection of AI-Generated Text, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[27] A. Voznyuk, G. Gritsai, A. Grabovoy, Team Advacheck at PAN: Multitasking Does All the Magic, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[28] J. Yang, K. Yan, Genre-Aware Contrastive Learning for AI Text Detection: A RoBERTa-Based Approach, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[29] S. A. Zaidi, H. T. Ahmed, S. A. Akbar, Z. Shakeel, F. Alvi, A. Samad, Team Nexus Interrogators at PAN: Voight-Kampff Generative AI Detection, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[30] T.-G. Marchitan, C. Creanga, L. P. Dinu, Unibuc - NLP at "Voight-Kampff" Generative AI Detection PAN 2025, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[31] L. Teja, A. Yadagiri, P. Pakray, Team CNLP-NITS-PP at PAN: Advancing Generative AI Detection: Mixture of Experts with Transformer Models, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[32] M. Jimeno-Gonzalez, E. Martínez-Cámara, P. G. Noelia Fernandez, L. A. U. na López, Team SINAI-INTA at PAN 2025: Uncovering machine generated text with linguistic features, in: Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[33] F. Völpel, O. Halvani, Adept: AI-Generated Text Detection Based on Phrasal Category N-Grams, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[34] J. K. Ochab, M. Matias, T. Boba, T. Walkowiak, StylOch at PAN: Gradient-boosted trees with frequency-based stylometric features, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working

Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[35] B. Ostrower, P. Doongare, M. T. Unnikrishnan, Binoculars, BART, and Adversaries: Multi-Faceted AI Text Detection for PAN 2025, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[36] A. R. Basani, P.-Y. Chen, DivEye at PAN 2025: Diversity Boosts AI-Generated Text Detection, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[37] S. Pudasaini, L. Miralles-Pechuán, D. Lillis, M. L. Salvador, Enhancing AI Text Detection with Frozen Pretrained Encoders and Ensemble Learning, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[38] Y. Sun, S. Afanaseva, K. Stowe, K. Patil, Bi-directional Cross-entropy loss and Stylometric Feature combined Classifier, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[39] J. Larson, Generative AI detection using simple Feature Selection and SVM, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[40] J. Huang, H. Cao, X. Lin, Z. Han, Application and Analysis of Roberta-base Model Fine Tuning Based on Data Enhancement in AI Text Detection, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[41] S. Titze, O. Halvani, LOG-AID: Logit-Based Statistical Features for AI Text Detection, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[42] R. Kumar, A. Trivedi, O. Varshney, Voight-Kampff AI Detection Sensitivity : IIITS@CLEF'25, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[43] Z. Liang, K. Sun, H. Cao, J. Luo, Z. Han, Research on Text Author Classification Based on ModernBERT and Gradient Loss Function, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[44] D. Macko, R. Moro, I. Srba, Increasing the robustness of the fine-tuned multilingual machine-generated text detectors, arXiv preprint arXiv:2503.15128 (2025).
[45] X. Hu, P.-Y. Chen, T.-Y. Ho, RADAR: Robust AI-text detection via adversarial learning, Neural Information Processing Systems abs/2307.03838 (2023) 15077–15095. URL: https://proceedings.neurips.cc/paper_files/paper/2023/hash/30e15e5941ae0cdab7ef58cc8d59a4ca-Abstract-Conference.html. doi:10.48550/arXiv.2307.03838. arXiv:2307.03838.
[46] H. Guo, S. Cheng, X. Jin, Z. Zhang, K. Zhang, G. Tao, G. Shen, X. Zhang, Biscope: Ai-generated text detection by checking memorization of preceding tokens, Advances in Neural Information Processing Systems 37 (2024) 104065–104090.
[47] O. Halvani, Constituent Treelib - A Lightweight Python Library for Constructing, Processing, and Visualizing Constituent Trees., 2024. URL: https://github.com/Halvani/constituent-treelib. doi:10.5281/zenodo.10951644.
[48] A. Peñas, Á. Rodrigo, A Simple Measure to Assess Non-response, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 2011, pp. 1415–1424. URL: https://aclanthology.org/P11-1142.pdf.
[49] J. Bevendorff, B. Stein, M. Hagen, M. Potthast, Generalizing unmasking for short texts, in: Proceedings of the 2019 Conference of the North, Association for Computational Linguistics, Stroudsburg, PA, USA, 2019, pp. 654–659. URL: https://aclanthology.org/N19-1068.pdf. doi:10.18653/v1/n19-1068.
[50] M. Kestemont, K. Luyckx, W. Daelemans, T. Crombez, Cross-Genre Authorship Verification Using Unmasking, English Studies 93 (2012) 340–356. URL: http://dx.doi.org/10.1080/0013838X.2012.668793. doi:10.1080/0013838X.2012.668793.

[51] Y. Wang, J. Mansurov, P. Ivanov, J. Su, A. Shelmanov, A. Tsvigun, C. Whitehouse, O. M. Afzal, T. Mahmoud, T. Sasaki, T. Arnold, A. F. Aji, N. Habash, I. Gurevych, P. Nakov, M4: Multi-generator, Multi-domain, and Multi-lingual Black-Box Machine-Generated Text Detection, in: Y. Graham, M. Purver (Eds.), Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, 2024, pp. 1369–1407. URL: https://aclanthology.org/2024.eacl-long.83.pdf.

[52] J. Hutson, Human-ai collaboration in writing: A multidimensional framework for creative and intellectual authorship, International Journal of Changes in Education (2025).

[53] Q. Zhang, C. Gao, D. Chen, Y. Huang, Y. Huang, Z. Sun, S. Zhang, W. Li, Z. Fu, Y. Wan, L. Sun, LLM-as-a-coauthor: Can mixed human-written and machine-generated text be detected?, in: K. Duh, H. Gomez, S. Bethard (Eds.), Findings of the Association for Computational Linguistics: NAACL 2024, Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 409–436. URL: https://aclanthology.org/2024.findings-naacl.29/. doi:10.18653/v1/2024.findings-naacl.29.

[54] M. Abassy, K. Elozeiri, A. Aziz, M. N. Ta, R. V. Tomar, B. Adhikari, S. E. D. Ahmed, Y. Wang, O. Mohammed Afzal, Z. Xie, J. Mansurov, E. Artemova, V. Mikhailov, R. Xing, J. Geng, H. Iqbal, Z. M. Mujahid, T. Mahmoud, A. Tsvigun, A. F. Aji, A. Shelmanov, N. Habash, I. Gurevych, P. Nakov, LLM-DetectAIve: a tool for fine-grained machine-generated text detection, in: D. I. Hernandez Farias, T. Hope, M. Li (Eds.), Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 336–343. URL: https://aclanthology.org/2024.emnlp-demo.35.

[55] Y. Wang, J. Mansurov, P. Ivanov, J. Su, A. Shelmanov, A. Tsvigun, O. Mohammed Afzal, T. Mahmoud, G. Puccetti, T. Arnold, A. Aji, N. Habash, I. Gurevych, P. Nakov, M4GT-bench: Evaluation benchmark for black-box machine-generated text detection, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 3964–3992. URL: https://aclanthology.org/2024.acl-long.218. doi:10.18653/v1/2024.acl-long.218.

[56] L. Dugan, D. Ippolito, A. Kirubarajan, S. Shi, C. Callison-Burch, Real or fake text?: Investigating human ability to detect boundaries between human-written and machine-generated text, 2022. URL: https://arxiv.org/abs/2212.12672. arXiv:2212.12672.

[57] L. Kushnareva, T. Gaintseva, G. Magai, S. Barannikov, D. Abulkhanov, K. Kuznetsov, E. Tulchinskii, I. Piontkovskaya, S. Nikolenko, Ai-generated text boundary detection with roft, 2024. URL: https://arxiv.org/abs/2311.08349. arXiv:2311.08349.

[58] Z. Zeng, S. Liu, L. Sha, Z. Li, K. Yang, S. Liu, D. Gašević, G. Chen, Detecting ai-generated sentences in human-ai collaborative hybrid texts: challenges, strategies, and insights, in: Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI '24, 2024. URL: https://doi.org/10.24963/ijcai.2024/835. doi:10.24963/ijcai.2024/835.

[59] Z. Zeng, L. Sha, Y. Li, K. Yang, D. Gašević, G. Chen, Towards automatic boundary detection for human-ai collaborative hybrid essay in education, 2023. URL: https://arxiv.org/abs/2307.12267. arXiv:2307.12267.

[60] T. Chakrabarty, P. Laban, C.-S. Wu, Can ai writing be salvaged? mitigating idiosyncrasies and improving human-ai alignment in the writing process through edits, 2025. URL: https://arxiv.org/abs/2409.14509. arXiv:2409.14509.

[61] E. Artemova, J. S. Lucas, S. Venkatraman, J. Lee, S. Tilga, A. Uchendu, V. Mikhailov, Beemo: Benchmark of expert-edited machine-generated outputs, in: L. Chiruzzo, A. Ritter, L. Wang (Eds.), Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), Association for Computational Linguistics, Albuquerque, New Mexico, 2025, pp. 6992–7018. URL: https://aclanthology.org/2025.naacl-long.357/.

[62] S. Saha, S. Feizi, Almost ai, almost human: The challenge of detecting ai-polished writing, 2025. URL: https://arxiv.org/abs/2502.15666. arXiv:2502.15666.

[63] G. Bao, L. Rong, Y. Zhao, Q. Zhou, Y. Zhang, Decoupling content and expression: Two-dimensional detection of ai-generated text, arXiv preprint arXiv:2503.00258 (2025).

[64] Z. Cheng, L. Zhou, F. Jiang, B. Wang, H. Li, Beyond binary: Towards fine-grained llm-generated text detection via role recognition and involvement measurement, arXiv preprint arXiv:2410.14259 (2024).

[65] D. Mollá, Q. Xu, Z. Zeng, Z. Li, Overview of the 2024 ALTA shared task: Detect automatic AI-generated sentences for human-AI hybrid articles, in: T. Baldwin, S. J. Rodríguez Méndez, N. Kuo (Eds.), Proceedings of the 22nd Annual Workshop of the Australasian Language Technology Association, Association for Computational Linguistics, Canberra, Australia, 2024, pp. 197–202. URL: https://aclanthology.org/2024.alta-1.17/.

[66] M. Li, E. Hovy, J. Lau, Summarizing multiple documents with conversational structure for meta-review generation, in: H. Bouamor, J. Pino, K. Bali (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2023, Association for Computational Linguistics, Singapore, 2023, pp. 7089–7112. URL: https://aclanthology.org/2023.findings-emnlp.472/. doi:10.18653/v1/2023.findings-emnlp.472.

[67] N. Dycke, I. Kuznetsov, I. Gurevych, NLPeer: A unified resource for the computational study of peer review, in: A. Rogers, J. Boyd-Graber, N. Okazaki (Eds.), Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Toronto, Canada, 2023, pp. 5049–5073. URL: https://aclanthology.org/2023.acl-long.277/. doi:10.18653/v1/2023.acl-long.277.

[68] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, Language models are unsupervised multitask learners, 2019. URL: https://api.semanticscholar.org/CorpusID:160025533.

[69] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, 2020. URL: https://arxiv.org/abs/2005.14165. arXiv:2005.14165.

[70] N. S. Keskar, B. McCann, L. R. Varshney, C. Xiong, R. Socher, CTRL: A conditional transformer language model for controllable generation, CoRR abs/1909.05858 (2019). URL: http://arxiv.org/abs/1909.05858. arXiv:1909.05858.

[71] X. Li, P. Yu, C. Zhou, T. Schick, O. Levy, L. Zettlemoyer, J. Weston, M. Lewis, Self-alignment with instruction backtranslation, arXiv preprint arXiv:2308.06259 (2023).

[72] B. Li, H. Qi, K. Yan, Team Bohan Li at PAN: DeBERTa-v3 with R-Drop regularization for Human-AI Collaborative Text Classification, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[73] M. Zheng, Y. Zhong, F. Liu, T. Xian, M. Xie, W. Wu, Z. Zhang, Q. Sun, StarBERT: A Hybrid Neural Network Model for Human-AI Collaborative Text Classification, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[74] T. Xian, Y. Zhong, F. Liu, M. Xie, Q. Sun, M. Zheng, W. Wu, Z. Zhang, DBG: Human-AI Collaborative Text Classification with DeBERTa-enhanced Contextual and Geometric Attention, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[75] T. Li, Fine-Grained Human-AI Collaborative Text Classification Using DeBERTa, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[76] D. Ignatenko, K. Zaitsev, O. Shkriaba, ReText.Ai Team at PAN 2025: Applying a Multiple Classification Heads to a Transformer Model for Human-AI Collaborative Text Classification, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[77] Q. Sun, L. Ma, W. Yang, T. Xian, M. Xie, W. Wu, Z. Zhang, M. Zheng, DeBERTa-FPN: Fusion Feature Pyramid Network for Human-AI Collaborative Text Classification, in: G. Faggioli, N. Ferro,

P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[78] W. Wu, W. Yang, Z. Zhang, M. Xie, M. Zheng, T. Xian, Q. Sun, Bert_T for Human-AI Collaborative Text Classification, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[79] Z. Zhang, W. Yang, W. Wu, M. Xie, M. Zheng, Q. Sun, T. Xian, DBA: A Hybrid Neural Network Model for Generative Human-AI Collaborative Text Classification, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[80] D. G. Sánchez, J. Jimenez, M. Ramírez, J. Martinez, RoBERT-IA: Human-AI Collaborative Text Classification, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[81] Y. Fuchuan, H. Cao, H. Zhongyuan, Sentence-Level AI-Generated Text Detection with Fine-Tuned BERT, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.