Perplexity-Driven Contrastive Scoring for Unsupervised Detection of AI-Generated Texts in Polish

Damian Stachura
Evidence Prime, Kraków, Poland
damian.stachura1@gmail.com

Abstract

The ŚMIGIEL competition (Przybyła et al., 2025) at PolEval 2025 focuses on distinguishing Polish human-written text from AI-generated text. I participated in one of the subtasks that required a zero-shot detection method. My solution adapts the Binoculars detector by pairing language models and using calibrated thresholds. Specifically, I replaced the English language models from the original Binoculars method with models trained on Polish corpora. This approach achieved first place in the chosen competition track. Overall, my findings demonstrate that domain-specific language models and careful thresholding enable state-of-the-art zero-shot AI-text detection performance across new languages and domains. The code is publicly available at https://github.com/damian1996/2025-smigiel.

1 Introduction

Detecting AI-generated text is a pressing task as large language models (LLMs) become pervasive. In educational and content-moderation settings, distinguishing human and machine writing is crucial for ensuring authenticity and preventing misuse. The PolEval 2025 proposed a task caled ŚMIGIEL (Spotting Machine-Generated Text from Language Models for Polish). ŚMIGIEL provides a dataset of Polish texts, both human-authored and AI-generated, challenging participants to build effective detectors for this language. Importantly, the most robust detectors operate in a zero-shot or unsupervised setting, requiring no labeled examples of AI text from the target domain. Zero-shot methods are desirable because they generalize across domains and languages without costly retraining. For example, the Binoculars approach (Hans et al., 2024) contrasts two pretrained LLMs to score a document, achieving over 90% detection at extremely low false positive rates without any model-specific training. Similarly, DetectGPT (Mitchell et al., 2023) uses the log-probability curvature of a single LLM's output to flag generation, also without additional training. These works show that clever use of existing language models can separate human and machine text in a fully unsupervised way.

Perplexity-based methods are a classic example of zero-shot detection. In essence, perplexity measures how surprising or unpredictable a text is according to a language model. Prior studies have found that machine-generated text often has lower perplexity. It means that it is more predictable to an language model than human text. For instance, Liu and Kong (2024) used a sliding window of GPT-2 perplexity to capture patterns of AI text, demonstrating that such a metric can effectively distinguish AI-written samples. A popular AI detector called GPTZero [1] explicitly leverages sentence-level perplexity and a related burstiness metric to score text. The burstiness reflects variability in the writing. Humans tend to vary sentence complexity and word choice more than AI, which writes at a consistently uniform level. Indeed, recent work on stylometric detection shows that machine texts cluster tightly by model, whereas human writing exhibits greater stylistic diversity. My approach builds on these insights. I used Polish LLMs to compute contrastive log-probabilities and note that human Polish writers, like English writers, naturally introduce more variety and idiosyncrasy in their text. By calibrating a threshold on these scores for the ŚMIGIEL data, I effectively adapted the zero-shot Binoculars (Hans et al., 2024) method to Polish. In the end, my unsupervised detector using Polish language models and the Binoculars scoring rule, outperformed all other entries to win first place on the ŚMIGIEL leaderboard.

[1] https://gptzero.me/

Proceedings of the PolEval 2025 Workshop, pages 21–25
November 21, 2025 ©2025 Association for Computational Linguistics

2 Method

2.1 Perplexity Score as AI-Generated Text Detector

Perplexity is classical method for detection of AI-generatex texts. The perplexity of text can be computed under a language model M, defined as the exponentiation of the average negative log-likelihood, reported as log-perplexity, of the tokens in the text. Intuitively, perplexity measures how surprising a string is to a model. In practice, modern LLMs score human-written text as higher-perplexity than text they generate, because the model is trained to assign high probability to its own outputs. Thus, perplexity-based detection flags a text as machine-generated if its perplexity under the model is unusually like presented in Gutiérrez Megías et al. (2024). In formula terms, for a token sequence X of length T and model probability for ith token contitioned on the sequence of tokens $p_\theta(x_i \mid x_{<i})$, the log-perplexity is defined as

$$\log \text{PPL}(M, X) = -\frac{1}{T}\sum_{i=1}^{T} \log p_\theta(x_i|x_{<i}) \quad (1)$$

and the actual perplexity is $\exp(\ell_{\text{ppl}})$. Lower perplexity indicates text closely matches the model's distribution. Since human text tends to have higher perplexity under the model, simple detectors have used a perplexity threshold to distinguish between human and AI text.

2.2 Cross Perplexity Score

Cross-Perplexity, noted as $\text{XPPL}(M_p, M_t; X)$, is a metric derived from cross-entropy and is utilized to assess the difference in the probability distributions predicted by two distinct language models M_p and M_t over a given sequence of text X. This metric is particularly relevant in the field of AI-generated text detection as it can quantify how surprising a text is to the candidate model M_p when its probability distribution is evaluated based on the expected distribution of the reference model M_t. In more details, cross-perplexity is the average per-token cross-entropy when the next-token probabilities of M_t are evaluated under M_p. The final score is computed as a logarithm of XPPL score and stated as log XPPL. Formally,

$$logXPPL = -\frac{1}{T}\sum_{i=1}^{T} M_p(s_i) \cdot \log M_t(s_i) \quad (2)$$

, where log XPPL = log $\text{XPPL}(M_p, M_t, X)$, and $M_p(s_i)$ and $M_t(s_i)$ denote the probability distributions over the vocabulary for the next token at position i. At each step i, the dot product between $M_p(s_i)$ and $logM_t(s_i)$ is computed.

2.3 Binoculars Score

The Binoculars method uses two LLMs to overcome pitfalls of naive perplexity detection (Hans et al., 2024). One of the significant challenges is associated with model prompting due to its potential to influence the linguistic structure of the generated text, resulting in outputs with elevated perplexity scores that may evade detection by perplexity-based AI-generated text classifiers. Furthermore, the typical lack of access to the original prompts makes it impossible to verify the output's adherence to the specified instructions, introducing difficulties in assessing model fidelity.

Let M_p be an observer model and M_t a performer model. Both models are chosen under the assumption that they need to have the same tokenizer. I compute the model's self-perplexity $\text{PPL}(M_t, X)$ and the cross-perplexity $\text{XPPL}(M_p, M_t; X)$, as defined in the earlier sections. The core Binoculars score is then defined as the ratio of computed self-perplexity to cross-perplexity:

$$\text{Binoculars}(X) = \frac{\log \text{PPL}(M_t, X)}{\log \text{XPPL}(M_p, M_t; X)} \quad (3)$$

Intuitively, this ratio re-scales the perplexity by the baseline perplexity that model M_p assigns to the next-token predictions of the model M_t. In typical use, M_p and M_t are two similar models so that their predictions diverge less on machine text than on human text. In the original paper authors used a pair of base model and its instruction-tuned variant. A key advantage is that if a text is high-perplexity only because it is unlikely overall. The cross-perplexity normalizes this: humans tend to choose even more surprising continuations than the model does, so human text yields a larger PPL relative to XPPL. In practice, Binoculars vastly outperforms raw perplexity at low false-positive rates.

2.4 Threshold selection

The Binoculars scores are converted into a binary decision by thresholding. Specifically, the decision threshold are set as the median of binocular score computed over multiple texts. Texts with score below threshold are classified as AI-generated while higher scores are considered human-written. This choice centers the decision boundary on the observed distribution of scores.

2.5 Models

As mentioned earlier, the Binoculars score requires two LLMs that share the same tokenizer. I use three pairs of Polish models proposed by SpeakLeash. Each pair comprising two versions of the same base architecture. I decided to use two instruction tuned models based on the same base model or combination instrunction-tuned and base models. Each of these pairs uses the same tokenizer, satisfying the Binoculars requirement.

- Bielik-11B-v2 (Ociepa et al., 2025b) and Bielik-11B-v2.3-Instruct [2]
- Bielik-11B-v2.5-Instruct [3] and Bielik-11B-v2.6-Instruct [4]
- Bielik-4.5B-v3 (Ociepa et al., 2025a) and Bielik-4.5B-v3.0-Instruct [5]

2.6 My Approach

The final submissions used the proposed Binoculars method with two different thresholding strategies and two backbone models, Bielik-4.5B-v3 and Bielik-4.5B-v3.0-Instruct, for computing the logPPL and logXPPL scores. In the first submission, reported in Table 1, the threshold was determined using the validation dataset. In the second submission, the threshold was set to the median of the Binoculars scores across the entire test set. The resulting performance differences between the two submissions were negligible. The corresponding threshold values were 0.936 and 0.931, respectively.

3 Results

3.1 Task Description

The dataset for ŚMIGIEL competition (Przybyła et al., 2025) consists of Polish texts from the four sources:

- customer reviews
- literature
- social media
- wikipedia

Model 1	Model 2	Score
Bielik-4.5Bv3	Bielik-4.5B-v3.0-Instruct	79.6
Bielik-11B-v2	Bielik-11B-v2.3-Instruct	79.4
Bielik-11B-v2.5-Instruct	Bielik-11Bv2.6-Instruct	72.4

Table 1: Average Binoculars scores for various pairs of Polish LLMs on the validation dataset.

Place	Accuracy
1	**79.77**
2	**79.70**
3	75.74
4	74.79
5	70.23

Table 2: Final ŚMIGIEL Leaderboard. Two first systems were submitted by me.

The human-written and AI-generated texts are evenly distributed across the dataset. Machine-generated texts were produced by multiple open-weight LLMs like LLama 3.1 8B (Grattafiori et al., 2024), Bielik 7B (Ociepa et al., 2024), Mistral 7B (Jiang et al., 2023), Bielik 11B (Ociepa et al., 2025b), PLLuM 12B (Kocoń et al., 2025), and Gemma 3 27B (Team et al., 2025).

3.2 Comparing Model Pairs

I compared three pairs of models, all utilizing the same tokenizer as mentioned in the previous section. The key observation is that a pair consisting of the base model and its respective instruction-tuned model outperforms a pair of two similar instruction-tuned LLMs. This observation is further supported by the pairs of models mentioned in the original Binoculars paper. The results are presented in Table 2.

3.3 Competition Leaderboard

The performance of all participating systems is summarized on the competition leaderboard in Table 1. Among these, my two submissions obtained the highest accuracy scores. The strongest results overall were achieved by using the Binoculars score with Bielik-4.5Bv3 and Bielik-4.5B-v3.0-Instruct models as backbones for it.

3.4 Analysis

I conducted two additional measures to analyze the results achieved by Binoculars. I decided to visualize:

[2] huggingface.co/speakleash/Bielik-11B-v2.3-Instruct
[3] huggingface.co/speakleash/Bielik-11B-v2.5-Instruct
[4] huggingface.co/speakleash/Bielik-11B-v2.6-Instruct
[5] huggingface.co/speakleash/Bielik-4.5B-v3.0-Instruct

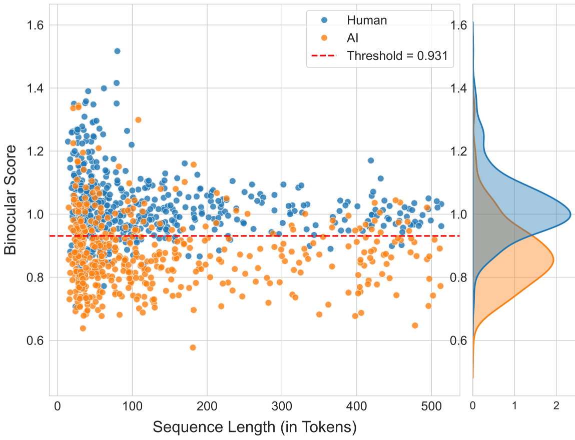

Figure 1: The distribution of the Binoculars scores was visualized as a function of text length (measured in tokens). This analysis used 1000 randomly selected samples from the provided validation dataset, with both classes (AI-generated and human-written) being evenly distributed.

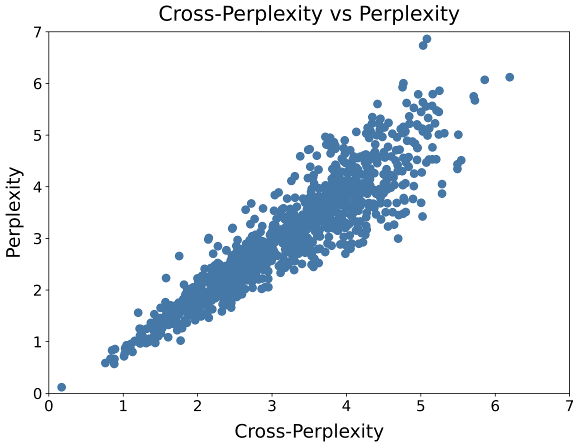

Figure 2: The distribution of the corresponding perplexity and cross-perplexity scores. This analysis used 1000 randomly selected samples from provided validation dataset with both classes being evenly distributed. Models Bielik-4.5Bv3 and Bielik-4.5B-v3.0-Instruct were used for this visualization.

- The distribution of the Binoculars score as a function of text length (in tokens)
- The distribution of the corresponding perplexity and cross-perplexity scores

In the first case, it can be observed that the Binoculars scores scale to similar ranges regardless of the number of tokens in the provided texts. This is a very important property as it demonstrates that the method can be used effectively, even for long texts. I presented it on the Figure 1

I also tested the distribution of the corresponding perplexity and cross-perplexity scores. The key insight from this analysis is that these distributions are very close to one another, suggesting that the specific models chosen as backbones for the Binoculars method may not be critical to its performance. The results for the pair consisting of Bielik-4.5Bv3 and Bielik-4.5B-v3.0-Instruct models are presented in Figure 2. Similarly, the distribution for the pair Bielik-11B-v2 and Bielik-11B-v2.3-Instruct is visible in Figure 3.

4 Discussion

Two key challenges in developing the proposed solution were the selection of models for score computation and the determination of an appropriate decision threshold.

In my experiments, only a narrow set of models was considered due to the requirement that both models share the same tokenizer. One potential direction for improving performance would be to evaluate additional Polish and multilingual large language models.

The second major challenge was selecting an optimal threshold. My two thresholds were computed separately for the validation and test datasets, yielding nearly identical values of 0.936 and 0.931, respectively. This similarity indicates that the data distributions of the validation and test sets are highly consistent. For comparison, the original Binoculars paper reported a threshold of 0.896 for an English dataset, suggesting that the distribution of Binoculars scores is broadly similar across languages. Consequently, further refinement of the threshold selection procedure is unlikely to result in substantial performance gains.

An alternative approach could involve the use of a hybrid method for samples with Binoculars scores close to the decision threshold. Such a strategy may help strengthen classification decisions for the most challenging texts.

5 Conclusions

My experiments showed that perplexity-based methods are too naive for AI-generated text detection. The emergence of powerful prompts, which significantly influence model behavior, presents a key pitfall for such detectors. Multiple teams have presented more sophisticated methods in recent years. In my submissions, I decided to measure the performance of Binoculars scores for Polish texts.

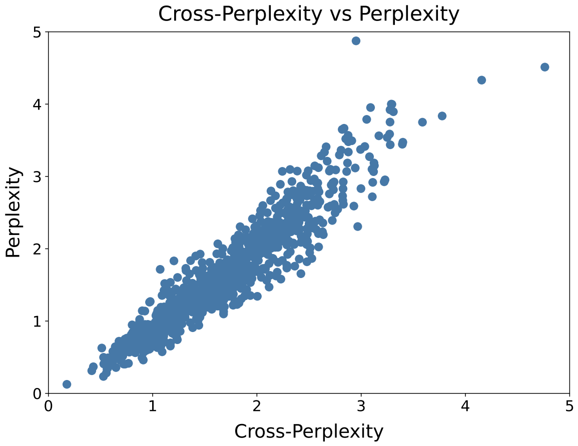

Figure 3: The distribution of the corresponding perplexity and cross-perplexity scores. This analysis used 1000 randomly selected samples from provided validation dataset with both classes being evenly distributed. Models Bielik-11B-v2 and Bielik-11B-v2.3-Instruct ere used for this visualization.

This method performed strongly compared to other submissions and allowed me to achieve first place in the unsupervised subtask of ŚMIGIEL. The best results were achieved using relatively small LLMs with 4.5 B parameters, which is a particularly interesting property as it makes the method relatively efficient and fast.

References

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Alberto José Gutiérrez Megías, L. Alfonso Ureña-López, and Eugenio Martínez Cámara. 2024. The influence of the perplexity score in the detection of machine-generated texts. In *Proceedings of the First International Conference on Natural Language Processing and Artificial Intelligence for Cyber Security*, pages 80–85, Lancaster, UK. International Conference on Natural Language Processing and Artificial Intelligence for Cyber Security.

Abhimanyu Hans, Avi Schwarzschild, Valeriia Cherepanova, Hamid Kazemi, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2024. Spotting llms with binoculars: zero-shot detection of machine-generated text. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Jan Kocoń, Maciej Piasecki, Arkadiusz Janz, Teddy Ferdinan, Łukasz Radliński, Bartłomiej Koptyra, Marcin Oleksy, Stanisław Woźniak, Paweł Walkowiak, Konrad Wojtasik, Julia Moska, Tomasz Naskręt, Bartosz Walkowiak, Mateusz Gniewkowski, Kamil Szyc, Dawid Motyka, Dawid Banach, Jonatan Dalasiński, Ewa Rudnicka, and 80 others. 2025. Pllum: A family of polish large language models. *arXiv preprint arXiv:2511.03823*.

Xurong Liu and Leilei Kong. 2024. Ai text detection method based on perplexity features with strided sliding window. In *Working Notes of CLEF 2024 — Conference and Labs of the Evaluation Forum (CLEF 2024)*, number 3740 in CEUR Workshop Proceedings, pages 2755–2760, Aachen.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: zero-shot machine-generated text detection using probability curvature. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Krzysztof Ociepa, Łukasz Flis, Remigiusz Kinas, Krzysztof Wróbel, and Adrian Gwoździej. 2025a. Bielik v3 small: Technical report. *Preprint*, arXiv:2505.02550.

Krzysztof Ociepa, Łukasz Flis, Krzysztof Wróbel, Adrian Gwoździej, and Remigiusz Kinas. 2024. Bielik 7b v0.1: A polish language model – development, insights, and evaluation. *Preprint*, arXiv:2410.18565.

Krzysztof Ociepa, Łukasz Flis, Krzysztof Wróbel, Adrian Gwoździej, and Remigiusz Kinas. 2025b. Bielik 11b v2 technical report. *Preprint*, arXiv:2505.02410.

Piotr Przybyła, Jakub Strebeyko, and Alina Wróblewska. 2025. PolEval 2025 Task 1 Śmigiel: Spotting Machine-Generated Text from LLMs for Polish. In *Proceedings of the PolEval 2025 Workshop*.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.