# Overview of the Text Watermarking Task at PAN 2026

Henry Plutz[1,*], Andreas Jakoby[1], Janek Bevendorff[1], Maik Fröbe[1,2], Martin Potthast[3,4,5] and Benno Stein[1]

[1]Bauhaus-Universität Weimar, Germany
[2]Friedrich-Schiller-Universität Jena, Germany
[3]University of Kassel, Germany
[4]hessian.AI, Germany
[5]ScaDS.AI, Germany

pan@webis.de https://pan.webis.de

**Abstract**

The new Text Watermarking Task at PAN 2026 aims to attribute texts to their source using invisible watermarks. Participants are asked to develop a post-hoc text watermarking system that can embed watermarks into existing texts. The resulting texts are then subjected to different systematic obfuscations designed to remove the watermark. Submitted watermarking systems should then attempt to detect the watermark, demonstrating the robustness of their approaches against such obfuscations. To evaluate the systems, a new metric called Text Watermarking Fidelity (TWF) is proposed, which jointly measures watermark detection performance and similarity between the watermarked and original text. Six teams submitted watermarking systems and five work notebooks were submitted describing their approach. The best-performing system achieved a TWF score of 0.818, outperforming the baseline by 18.5 percentage points. This paper provides an overview of the task, summarizes the approaches submitted by participants and discusses the results.

**Keywords**

Text Watermarking, LLM Watermarking, Watermark Robustness, Workshop, PAN

## 1. Introduction

Text watermarking is the task of embedding an imperceptible identifier, such as a statistical signal, into a text to enable text attribution. Although text watermarking has long been an active topic in research [1], the emergence of Large Language Models (LLMs) has renewed interest in text watermarking as a method of mitigating the potential misuse of LLM-generated content [2]. While LLM detection is already the focus of another PAN shared task, the Voight-Kampff Generative AI / LLM Detection task [3], watermarking offers a complementary approach by enabling direct influence on signals contained in generated texts instead of only relying on post-hoc detection. As LLM-generated texts are expected to become increasingly difficult to distinguish from human-written text in the future, robust watermarking may provide a reliable solution.

This year, PAN organizes the world's first shared task dedicated to text watermarking. In this task, participants receive texts in which they must embed a watermark. The resulting watermarked texts are subsequently subjected to different obfuscations, aiming to remove the watermark. The obfuscations are designed with several levels of severity, such that the watermark is tested in more extreme and in less severe scenarios. Finally, the watermark should be detected by the submitted watermarking system and the detection performance is evaluated. To penalize watermarking systems that substantially change the meaning of the original text, a new metric called Text Watermarking Fidelity (TWF) is proposed,

*CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany*
*Corresponding author.
henry.plutz@uni-weimar.de (H. Plutz); andreas.jakoby@uni-weimar.de (A. Jakoby); janek.bevendorff@uni-weimar.de (J. Bevendorff); maik.froebe@uni-weimar.de (M. Fröbe); martin.potthast@uni-kassel.de (M. Potthast); benno.stein@uni-weimar.de (B. Stein)
0009-0006-1917-6067 (H. Plutz); 0000-0002-3797-0559 (J. Bevendorff); 0000-0002-1003-981X (M. Fröbe); 0000-0003-2451-0665 (M. Potthast); 0000-0001-9033-2217 (B. Stein)

which balances detection accuracy and similarity to the original text. Thus, participants are encouraged to develop a watermarking system that is robust against text obfuscations and preserves the content of the original text. This task aims to establish a benchmark for evaluating the robustness of text watermarking methods under several obfuscation scenarios.

In Section 2, the task, including the dataset and obfuscations, is described in detail. Section 3 summarizes the submitted approaches by participants. Afterwards, Section 4 presents and discusses the results of this task. The paper is concluded in Section 5. The baseline and the code used for obfuscating the texts and evaluating the systems are available.[1]

## 2. Text Watermarking Task

This section outlines the text watermarking task, the dataset used, the obfuscations and the evaluation method.

### 2.1. Task Definition

The Text Watermarking task requires participants to develop a system that embeds a watermark into existing text and subsequently detects the embedded watermark. The task is structured into multiple steps.

1. The participants develop a watermarking system that can embed a watermark into existing text and perform watermark detection. Participants have access to a training dataset for developing and testing their system but they may use other data sources for this purpose.
2. The participants submit their watermarking system to TIRA [4]. This includes the embedding component and the detection component.
3. The submitted system is run on the test dataset to generate watermarked texts.
4. The watermarked texts are subjected to a set of predefined obfuscations. Each obfuscation separately results in a new obfuscated text. Additionally, each of the original texts are also subjected to the same set of obfuscations, such that a balanced set of watermarked and unwatermarked texts is created.
5. The watermark detection system is run on the obfuscated texts to evaluate the performance on detecting watermarks.

This task design enables the evaluation of the robustness of submitted watermarking systems under realistic text modifications. By also testing unwatermarked text on the detection system, both detection performance and false positive behaviour can be observed.

### 2.2. Dataset

The training and test datasets were constructed from the ParlLawSpeech dataset [5], which consists of a corpus of political speeches from the European Parliament. The training and test set contain 300 parliamentary speeches each in the English language of varying lengths. The training set is only used as a help for participants in developing their watermarking system. The test set was constructed such that it contains texts of different lengths. This is important because the effectiveness of watermark detection strongly depends on the text length and watermarks are more difficult to detect in short texts [6]. The test set comprises of 100 texts containing 50-150 words (short), another 100 texts containing 400-600 words (medium) and another 100 texts containing 1000-1200 words (long). The final test set removed meta comments and other textual artefacts from the selected texts, resulting in clean texts.

[1] Code is available on GitHub.

**Table 1**
Obfuscations on watermarked texts used in the PAN Text Watermarking task. Each obfuscation is applied at four severity levels. Level 1 affects every unit, level 2 every second unit, level 3 every fourth unit and level 4 every eighth unit.

| Obfuscation Type | Unit | Description |
|---|---|---|
| Typo Attack | Word | Randomly replace one letter in each word with adjacent letters on a QWERTY-keyboard. |
| Paraphrasing Attack | Sentence | Paraphrase each sentence using another LLM. |
| Copy-Paste Attack | Sentence | Replace each watermarked sentence with an unwatermarked sentence. |

## 2.3. Obfuscations

The robustness of the watermarking systems can be evaluated by subjecting watermarked texts to obfuscations aiming to remove the watermark. The Text Watermarking task independently applies three different types of obfuscations with varying levels of severity, which were kept secret from participants. These obfuscations are commonly used in text and LLM watermarking literature [7, 8] and were adapted to fit our evaluation scheme. An overview of the obfuscations is provided in Table 1. The first obfuscation introduces typos into the text, resembling errors when typing in text per hand. Four severity levels are considered by replacing one random letter in either every word, every second word, every fourth word or every eighth word with an adjacent letter on a QWERTY-keyboard. The second obfuscation paraphrases parts of the text. Depending on the severity, every sentence, every second sentence, every fourth sentence or every eighth sentence is paraphrased. For paraphrasing, the `Qwen2.5-1.5B-Instruct` [9] language model is used. The third obfuscation replaces watermarked sentences with their corresponding unwatermarked sentence from the original text, simulating a copy-paste attack. Every sentence, every second sentence, every fourth sentence or every eighth sentence is replaced depending on the severity level. Applying all obfuscations across four severity levels instead of only one provides a more comprehensive evaluation of the watermark robustness under this range of attack scenarios.

## 2.4. Evaluation

While the accuracy score can measure the detection performance of the watermarking system, one limitation of it in this scenario is that watermarking systems that completely change the original meaning of the text can still achieve high scores. For this reason, we propose a new metric named Text Watermarking Fidelity (TWF), which jointly captures watermark detection performance and the similarity of the watermarked text to the original text. It is defined as

$$\mathrm{TWF} = \max(\mathrm{BLEU}, \mathrm{BERTScore}) \times \mathrm{BA},$$

where BA denotes the balanced accuracy of the watermark detector on watermarked and unwatermarked texts. BLEU measures the syntactic similarity between the watermarked and the original text [10], while BERTScore measures the semantic similarity between the watermarked and the original text [11]. BLEU and BERTScore are computed at sentence level and are then averaged over the full text. While the balanced accuracy is computed after applying the obfuscations, BLEU and BERTScore are computed for the unmodified watermarked text as the obfuscations are not part of the watermarking system. All watermarking systems are evaluated using the TWF metric. As a baseline, an implementation of the KGW watermarking scheme [6] is used[2]. Since the KGW watermarking scheme is a generation-inherent LLM watermarking scheme, whereas this task focuses on post-hoc text watermarking, the baseline was implemented such that it inserts the watermark while paraphrasing the original text.

[2]Baseline is available on GitHub.

## 3. Submissions

We received six software submissions and five work notebooks that describe their approach. We describe each notebook submission briefly below.

**Nuovo et al. [12]** propose a watermarking system, that incorporates multiple channels. Two keyed channels perform lexical substitutions using WordNet and contraction manipulations based on assigned green and red bits. A key-independent channel applies syntactic transformations such as active-to-passive voice conversion and semicolon substitution. Detection is performed by aggregating evidence from all channels using a combined log-likelihood ratio test. The watermarking system is especially optimized to the domain of English political speeches present in the dataset. Besides the three-channel system (v1), they also submitted an extended system (v2) which uses a domain-specific vocabulary signal when embedding the watermark and by basing the synonym selection on the BERTScore instead of the Wu-Palmer similarity.

**Ibrahim [13]** extends on the idea of the KGW watermarking scheme [6] and PostMark [14]. An HMAC key is used to partition the vocabulary into green and red words. To address the repetitive wording common in political speeches contained in the dataset, only unique word types are considered instead of token occurences during watermark detection. All occurences of high-frequency red-list word types are replaced with synonyms, while green-list words are preferably selected. The watermarking system further introduces a dual-key system in which a second key is used to substitute other word types. They submitted both a system using only one key (Single-key) and a system using two keys (Dual-key).

**Elshamy et al. [15]** propose more suitable parameters for the KGW baseline. By choosing the self-referential hashing scheme and an increased logit bias, the robustness of the watermark is improved. Another approach by **Elshamy et al. [16]** splits a text into individual sentences and applies the KGW watermark to each sentence independently. The watermarked sentences are then concatenated to form the full watermarked text. Detection on the full concatenated text works as the z-score still accumulates across all tokens.

**Rudolph and Stadelmann [17]** developed a watermarking system that incorporates two channels, a sentence-level SimMark channel and a token-level green list channel. For each sentence, the SimMark channel generates paraphrases and selects the one that falls into a predefined cosine similarity bin relative to the preceding two sentences. The token-level green list channel replaces eligible tokens with green-list near-synonyms proposed by a masked language model. The partition into green and red words follows the KGW watermarking scheme [6]. For watermark detection, each channel independently produces a p-value, which are then combined using Fisher's method.

**Table 2**
Overall results for the Text Watermarking task ranked by TWF. Best results are marked in bold.

| Team | TWF | BA | BLEU | BERTScore |
|---|---|---|---|---|
| *yeyang (no notebook submitted)* | **0.818** | **0.824** | **0.948** | **0.992** |
| aimoment (Single-key) [13] | 0.775 | 0.785 | 0.904 | 0.988 |
| team [17] | 0.762 | 0.773 | 0.857 | 0.987 |
| JRC-GENESIS (v2) [12] | 0.761 | 0.785 | 0.642 | 0.969 |
| aimoment (Dual-key) [13] | 0.734 | 0.753 | 0.789 | 0.975 |
| *alex-csed-3n (no notebook submitted)* | 0.727 | 0.819 | 0.071 | 0.887 |
| JRC-GENESIS (v1) [12] | 0.686 | 0.720 | 0.572 | 0.952 |
| KGW Baseline [6] | 0.633 | 0.720 | 0.014 | 0.880 |

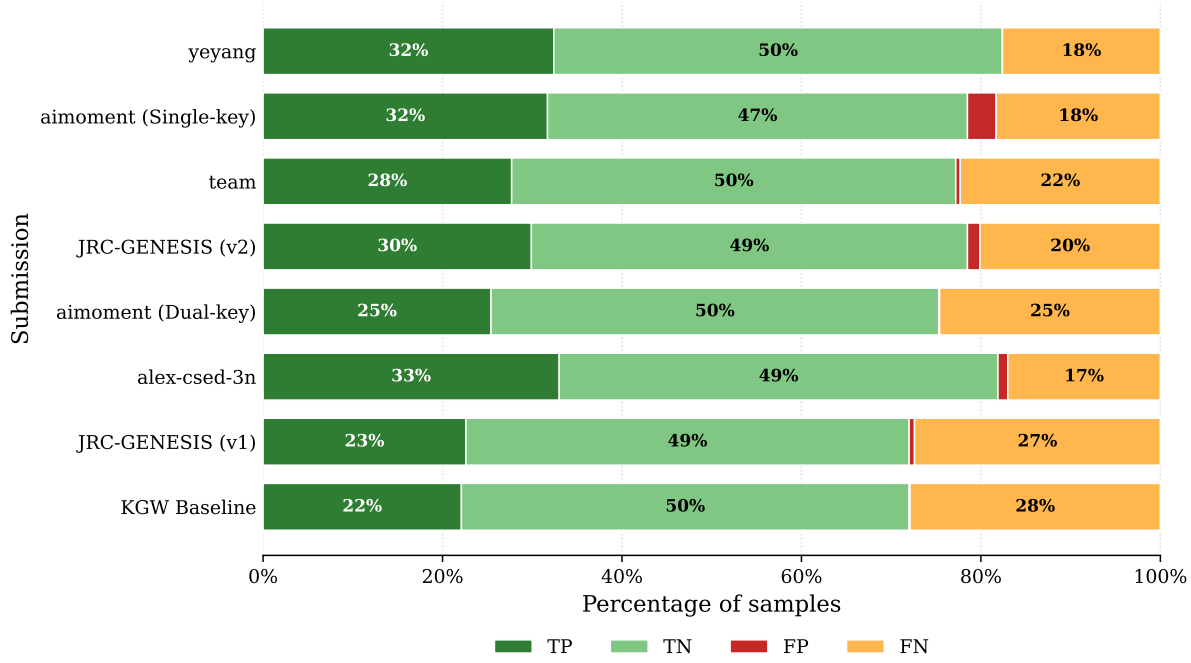


**Figure 1:** Breakdown of watermark detection outcomes for each submitted system sorted by rank in the overall TWF results. Each bar shows the proportion of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) on the test dataset.

## 4. Evaluation Results

Table 2 presents the results on the PAN 2026 text watermarking test dataset. The team ”AlexU” is missing from the ranking because the performance metrics from their submitted systems could not be obtained. All submitted systems outperform the KGW baseline method in terms of TWF. The highest TWF score of 0.818 is achieved by team yeyang. Unfortunately, no work notebook was submitted for this system, thus their approach cannot be analyzed in detail. Among the teams that submitted a notebook, the highest TWF score of 0.775 is achieved by the Single-key watermarking system submitted by team aimoment.

An interesting observation is that the TWF score is driven by the BERTScore rather than BLEU as every submitted system achieves a higher BERTScore than BLEU. Furthermore, different strategies can be observed with the BLEU scores. While some systems heavily paraphrase or alter the wording of the original text, resulting in lower BLEU scores, other approaches preserve the original wording more closely in the watermarked text, which results in higher BLEU scores. Both strategies are valid for embedding watermarks evaluated with the TWF metric, as only the maximum of BLEU and BERTScore contributes to the final score.

The detection accuracy is another key influencing the TWF score. In general, higher ranked systems also achieve higher balanced accuracy scores. The TWF score works as intended, as for example the system submitted by team alex-csed-3n achieves the second-highest balanced accuracy among all submitted systems, but is held back by a lower BLEU and BERTScore compared to the top-ranked systems, reducing its overall TWF score.

Figure 1 shows the breakdown of the confusion matrix outcomes during watermark detection for each system. It can be observed that all submitted systems can consistently identify the unwatermarked samples and the false positive rates are generally low. Better performing systems do not generally have lower false positive rates, as indicated by the higher number of false positives of the Single-key system by team aimoment especially compared to their Dual-key system. The TWF does not directly account for false positives besides the detection accuracy, which is why high false positive rates are not severely penalized in the ranking. Furthermore, better-performing systems achieve higher number

of true positives with the exception of the system by team alex-csed-3n, which detects the highest number of true positives among all submissions, but, as previously discussed, achieve a lower BLEU and BERTScore resulting in an overall lower ranking.

One limitation of the TWF score is that it does not penalize very simple and highly localized watermarking approaches, that only insert the watermark at one specific part of the original text. Obfuscations not attacking the specific part containing the watermark will result in high TWF scores as the BLEU and BERTScore will be near 1 because only a small part of the text is changed compared to the original. The detection accuracy will also be high as the probability of an obfuscation removing the watermark is low. Such kinds of watermarks are usually very simple to spot and easy to reverse-engineer. Future work should address this problem by adjusting the TWF metric or designing the obfuscations such that these kinds of watermarks are removed in most cases.

## 5. Conclusion

In the first edition of the PAN Text Watermarking task, participants developed post-hoc watermarking systems that were designed to be robust against text obfuscations. A total of six teams submitted software along with five work notebooks and all of the submitted systems outperformed the provided KGW baseline. The results demonstrate the feasibility of robust post-hoc text watermarking as well as the effectiveness of the proposed TWF metric for evaluation and highlight the value of benchmarking and comparing watermark robustness. We hope that future editions of the task will further advance research on robust text watermarking.

## Acknowledgments

Supported by the BMFTR Project LANTMARK: Potential and limitations of LLMs for applications in Watermarking of text and spoken language. Funding ID 16KIS2307K.

## Declaration on Generative AI

During the preparation of this work, the authors used DeepL and ChatGPT in order to: Sentence polishing. After using these services, the authors reviewed and edited the content as needed and takes full responsibility for the publication's content.

## References

[1] N. S. Kamaruddin, A. Kamsin, L. Y. Por, H. Rahman, A review of text watermarking: Theory, methods, and applications, IEEE Access 6 (2018) 8011–8028. URL: https://doi.org/10.1109/ACCESS.2018.2796585. doi:10.1109/ACCESS.2018.2796585.

[2] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, E. H. Chi, T. Hashimoto, O. Vinyals, P. Liang, J. Dean, W. Fedus, Emergent abilities of large language models, Trans. Mach. Learn. Res. 2022 (2022). URL: https://openreview.net/forum?id=yzkSU5zdwD.

[3] J. Bevendorff, J. Karlgren, M. Potthast, B. Stein, Overview of the third "Voight-Kampff" Generative AI / LLM Detection Task at PAN and ELOQUENT 2026, in: A. Barrón-Cedeño, A. G. S. de Herrera, E. S. Salido, S. MacAvaney, J. M. Struß (Eds.), Working Notes of CLEF 2026 – Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2026.

[4] 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: 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.

[5] J. Schwalbach, L. Hetzer, S.-O. Proksch, C. Rauh, M. Sebők, Parllawspeech, (Version 1.0.0) [Data set]. GESIS, Köln. https://doi.org/10.7802/2824, 2025.

[6] J. Kirchenbauer, J. Geiping, Y. Wen, J. Katz, I. Miers, T. Goldstein, A watermark for large language models, in: A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, J. Scarlett (Eds.), International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA, Proceedings of Machine Learning Research, PMLR, 2023, pp. 17061–17084. URL: https://proceedings.mlr.press/v202/kirchenbauer23a.html.

[7] J. Piet, C. Sitawarin, V. Fang, N. Mu, D. A. Wagner, Markmywords: Analyzing and evaluating language model watermarks, in: IEEE Conference on Secure and Trustworthy Machine Learning, SaTML 2025, Copenhagen, Denmark, April 9-11, 2025, IEEE, 2025, pp. 68–91. URL: https://doi.org/10.1109/SaTML64287.2025.00012. doi:10.1109/SATML64287.2025.00012.

[8] J. Kirchenbauer, J. Geiping, Y. Wen, M. Shu, K. Saifullah, K. Kong, K. Fernando, A. Saha, M. Goldblum, T. Goldstein, On the reliability of watermarks for large language models, in: The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024, OpenReview.net, 2024. URL: https://openreview.net/forum?id=DEJIDCmWOz.

[9] Q. Team, Qwen2.5: A party of foundation models, 2024. URL: https://qwenlm.github.io/blog/qwen2.5/.

[10] K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, Bleu: a method for automatic evaluation of machine translation, in: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, ACL '02, Association for Computational Linguistics, USA, 2002, p. 311–318. doi:10.3115/1073083.1073135.

[11] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, Bertscore: Evaluating text generation with BERT, in: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020, OpenReview.net, 2020. URL: https://openreview.net/forum?id=SkeHuCVFDr.

[12] E. D. Nuovo, D. Colla, Q. Luong, S. Ceppi, Team JRC-GENESIS at PAN 2026: SYN2 – Hybrid Watermarking via Keyed SYNonym Substitution and SYNtactic Signals, in: E. Sánchez Salido, A. Barrón-Cedeño, A. G. S. de Herrera, S. MacAvaney, J. M. Struß (Eds.), Working Notes of CLEF 2026 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2026.

[13] M. Ibrahim, Team aimoment at PAN 2026: Post-Hoc Synonym Watermarking with Dual-Key Type-Level Detection, in: E. Sánchez Salido, A. Barrón-Cedeño, A. G. S. de Herrera, S. MacAvaney, J. M. Struß (Eds.), Working Notes of CLEF 2026 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2026.

[14] Y. Chang, K. Krishna, A. Houmansadr, J. Wieting, M. Iyyer, Postmark: A robust blackbox watermark for large language models, in: Y. Al-Onaizan, M. Bansal, Y. Chen (Eds.), Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024, Association for Computational Linguistics, 2024, pp. 8969–8987. URL: https://doi.org/10.18653/v1/2024.emnlp-main.506. doi:10.18653/V1/2024.EMNLP-MAIN.506.

[15] H. Elshamy, M. Elsayed, M. Torki, N. ElMakky, Team AlexU at PAN 2026: Robust KGW Watermarking via Self-Referential Hashing and Amplified Bias, in: E. Sánchez Salido, A. Barrón-Cedeño, A. G. S. de Herrera, S. MacAvaney, J. M. Struß (Eds.), Working Notes of CLEF 2026 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2026.

[16] H. Elshamy, M. Elsayed, M. Torki, N. ElMakky, Team AlexU at PAN 2026: Improving Watermark Fidelity via Sentence-Level Generation Chunking, in: E. Sánchez Salido, A. Barrón-Cedeño, A. G. S. de Herrera, S. MacAvaney, J. M. Struß (Eds.), Working Notes of CLEF 2026 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2026.

[17] T. Rudolph, N. Stadelmann, A Two-Channel Watermark: Combining SimMark and Sparse green listing, in: E. Sánchez Salido, A. Barrón-Cedeño, A. G. S. de Herrera, S. MacAvaney, J. M. Struß (Eds.), Working Notes of CLEF 2026 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2026.