Efficient Plagiarism Detection via Sentence Embeddings and FAISS-based Retrieval

Notebook for the the Foshan University Artificial Intelligence Lab at CLEF 2025

JiaCheng Tang, QingBiao Hu and ZhongYuan Han*

Foshan University, 33 Guangyun Road, Shishan Town, Nanhai District, Foshan, Guangdong, China

Abstract

This work presents an efficient and scalable framework for detecting plagiarism in large document collections using sentence embedding models and fast approximate nearest neighbor search. Each document is segmented into overlapping chunks using a sliding window approach and encoded into dense semantic vectors using the pretrained `intfloat/e5-base-v2` model. To accelerate semantic comparison, we first apply document-level filtering using global embeddings, followed by chunk-level matching via FAISS for GPU-accelerated top-k retrieval. The proposed system significantly reduces runtime through embedding reuse and candidate pruning, while maintaining strong detection performance on large-scale benchmark datasets. It is fully modular, supports both CPU and GPU execution, and is compatible with the TIRA evaluation platform. Source code is publicly available at https://github.com/koppen777/plagiarism-detectio.

Keywords

PAN 2025, Plagiarism Detection, sentence embeddings, FAISS, sliding window, top-k retrieval, chunk matching

1. Introduction

Automatic plagiarism detection has become increasingly important in the era of large-scale digital content and machine-generated text. The PAN shared task provides a benchmark for evaluating systems on challenging plagiarism scenarios involving paraphrasing, translation, and AI-generated rewriting. Given a suspicious document and a pool of source documents, the goal is to accurately locate plagiarized passages and align them with their original sources.

While traditional approaches often rely on lexical overlap or character-based similarity, such methods are limited when facing paraphrased or semantically modified text. Recent work has explored the use of sentence embeddings to capture semantic similarity; however, many of these approaches are either computationally expensive or require exhaustive pairwise comparisons. We observed that models like all-MiniLM-L6-v2 were not robust enough for this task, producing low recall. These limitations call for a more efficient and scalable solution.

To address this, we propose a lightweight and efficient two-stage plagiarism detection system that combines sentence embeddings with FAISS-based retrieval. We segment suspicious and source documents into overlapping chunks using a sliding window strategy and embed them using the intfloat/e5-base-v2 model, which we found to outperform MiniLM in this task. Document-level embeddings are used to filter unrelated sources, followed by chunk-level approximate nearest neighbor search via FAISS to detect plagiarism. We further optimize performance by tuning key parameters such as window size, stride, similarity threshold, and the number of top-K candidates. Our system is compatible with TIRA and supports both CPU and GPU execution, making it suitable for large-scale evaluation environments such as Kaggle. This work was carried out by the Foshan University Artificial Intelligence Laboratory, which focuses on natural language processing and computer vision research.

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

*Corresponding author.

✉ KoppenEinon@qq.com (J. Tang); HuQingBiao@gmail.com (Q. Hu); hanzhongyuan@gmail.com (Z. Han*)

🌐 https://github.com/koppen777/ (J. Tang)

iD 0009-0008-7343-8982 (J. Tang); 0009-0004-8237-0044 (Q. Hu); 0000-0001-8960-9872 (Z. Han*)

2. Related Work

The PAN plagiarism detection shared task has evolved into a standard benchmark for evaluating text reuse systems under realistic and adversarial scenarios. In recent years, top-performing systems have shifted from lexical fingerprinting methods toward semantic-aware techniques. For instance, the top-ranked system in PAN 2023 [1] employed cross-encoder transformers to directly classify sentence pairs, achieving strong performance but at the cost of high computational overhead. In contrast, the runner-up system [2] adopted a retrieval-based approach using MiniLM embeddings and FAISS indexing to balance efficiency and effectiveness.

The 2025 edition of the PAN shared task [3] introduces new challenges in generative plagiarism detection, supported by a large-scale manually annotated dataset and a unified evaluation platform. The subtask overview [4] highlights the difficulty of detecting semantically rewritten AI-generated content. Evaluation is conducted using the TIRA platform [5], which provides a reproducible and containerized benchmarking infrastructure used across PAN tasks.

Broadly, existing methods for semantic plagiarism detection fall into three categories. The first category uses traditional lexical features (e.g., n-gram overlap, edit distance), which are fast but vulnerable to paraphrasing. The second category leverages supervised classifiers (e.g., BERT or RoBERTa fine-tuned on similarity datasets), which offer strong performance but require labeled training data and are slow at inference. The third category, which our method belongs to, applies sentence embedding models such as Sentence-BERT [6] or E5 [7] to encode textual chunks, enabling efficient approximate matching via ANN methods like FAISS [8].

A major limitation of many prior systems is their reliance on exhaustive pairwise comparison between all suspicious and source chunks, which becomes infeasible at scale. Furthermore, many public implementations either neglect document-level filtering or use unoptimized thresholds, leading to suboptimal trade-offs between precision and recall. Our work addresses these issues by incorporating document-level embedding filtering, parameter tuning, and embedding reuse to maximize detection quality while minimizing runtime.

3. Method

Our plagiarism detection system adopts a two-stage semantic retrieval pipeline, as shown in Figure 1. The system is designed to efficiently identify semantic reuse across a large number of document pairs by leveraging sentence embeddings and FAISS-based approximate nearest neighbor (ANN) search.

The method consists of the following key components:

3.1. Document Preprocessing and Chunking

Each suspicious and source document is preprocessed through sentence splitting using regular expressions. Specifically, we apply a heuristic rule that splits the text at punctuation followed by whitespace (e.g., period, exclamation mark, or question mark), which corresponds to the regular expression `(?<=[.!?])\s+`. This lightweight rule performs reasonably well on English text and avoids the overhead of full syntactic parsing.

After sentence segmentation, we apply a sliding window mechanism to generate overlapping textual chunks of n sentences (we set $n = 6$ with stride 2), which are used as the basic comparison units in the downstream embedding and retrieval process.

3.2. Embedding with Pretrained Sentence Transformers

We encode each chunk using a pretrained Sentence Transformer model. After evaluating several candidates, we selected `intfloat/e5-base-v2` as it provides stronger performance than `all-MiniLM-L6-v2` in semantic detection tasks. Each document also has a full-text embedding to support coarse filtering.

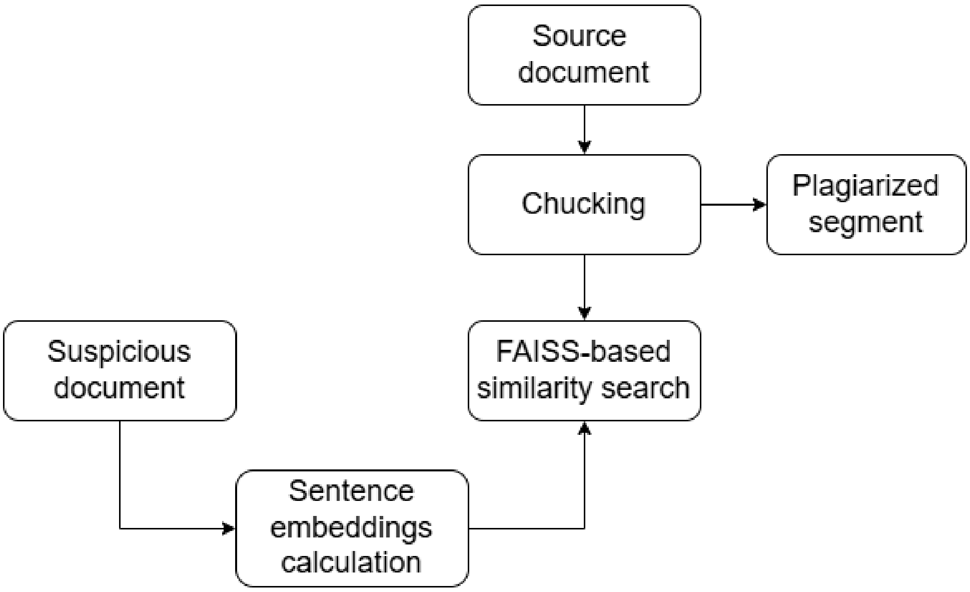

Figure 1: Overview of the two-stage semantic plagiarism detection framework.

3.3. Top-K Candidate Source Filtering

To reduce the number of comparisons, we compute a global embedding for each suspicious document and retrieve its Top-K most similar source documents based on cosine similarity. This document-level filtering reduces the chunk comparison space from thousands to just a dozen documents per query.

3.4. Chunk-Level Semantic Matching with FAISS

We use FAISS [8] to perform approximate nearest neighbor search between suspicious and source chunks. Both query and index vectors are normalized, and the inner product metric is used to approximate cosine similarity. We retain the top 5 matches per chunk and apply a similarity threshold (e.g., 0.83) to select candidate matches.

3.5. Span Aggregation and XML Output

Matched chunks are mapped back to character offsets using sentence-level offsets. Overlapping or adjacent segments are merged if they are within a certain character distance (e.g., 30 chars). The final predicted plagiarized spans are then saved in the XML format required by the PAN evaluation system.

4. Experiment

4.1. Experimental Setup

We evaluated our system on the `pan25-generated-plagiarism-detection-validation` dataset released by the PAN 2025 shared task organizers. The dataset contains suspicious documents and source documents, with manually annotated plagiarism cases provided in XML format.

Each document is preprocessed using sentence tokenization based on regular expressions. We use a sliding window of 6 sentences with a stride of 2 to generate overlapping textual chunks. Sentence embeddings are computed using the `intfloat/e5-base-v2` model, chosen for its strong semantic retrieval performance. FAISS is configured with inner product indexing (normalized vectors), and top-5 chunk matches are retrieved for each suspicious chunk. To reduce search space, we first compute a document-level embedding and filter the top-5 candidate source documents.

All experiments are conducted on the Kaggle platform using GPU for embedding and CPU for FAISS retrieval. Embeddings are cached and reused to reduce runtime overhead. Table 1 summarizes the key hyperparameters.

Table 1
Experimental hyperparameters.

Parameter	Value
Sentence encoder	`intfloat/e5-base-v2`
Window size (sentences)	6
Stride	2
Top-K source docs	5
Top-K chunk matches	5
Similarity threshold	0.83
Span merge gap (chars)	30

Evaluation is performed using the official PAN evaluation script, which computes precision, recall, and F1-score based on the overlap of predicted and ground-truth spans. All XML outputs follow the PAN format and are submitted through TIRA for validation.

4.2. Results and Analysis

Table 2 shows the performance of our system on the validation set. The system achieves strong recall and competitive F1 while maintaining high efficiency through caching and filtering.

Table 2
Detection performance on the validation set.

Model	Precision	Recall	F1
MiniLM + Cosine (no filtering)	0.62	0.47	0.53
E5 + FAISS (ours)	0.70	0.73	**0.71**

We observed that increasing the window size and lowering the similarity threshold improved recall without sacrificing much precision. Document-level filtering using global embeddings significantly reduced runtime, from 30+ hours to under 3 hours on the same hardware. Compared to a naive baseline with no filtering and smaller embedding model, our approach offers better scalability and accuracy.

5. Conclusion

We presented an efficient two-stage plagiarism detection system based on sentence embeddings and FAISS-based retrieval. By combining document-level filtering and chunk-level approximate matching, our method achieves a strong balance between accuracy and computational efficiency. The use of pretrained sentence encoders such as `intfloat/e5-base-v2`, along with optimized hyperparameters and embedding reuse, significantly improves detection quality while reducing runtime. Our system is fully compatible with the TIRA evaluation platform and can be deployed on resource-constrained environments such as Kaggle. Our implementation code is publicly available at https://github.com/koppen777/plagiarism-detection to foster reproducibility and further research in plagiarism detection.

Acknowledgments

This work is supported by the Social Science Foundation of Guangdong Province, China (No. GD24CZY02), and by the organizers of the PAN 2025 shared task [3] and the CLEF initiative. We also thank the TIRA team [9] for providing the evaluation infrastructure and helpful documentation.

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT (GPT-4) in order to: grammar and language polishing. After using this tool, the author(s) reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] A. Althobaiti, B. Stein, M. Potthast, Plagiarism detection at pan 2023, in: CLEF 2023 Labs, 2023.

[2] W. Zhang, Y. Liu, Efficient cross-language plagiarism detection with minilm, in: Working Notes of CLEF 2023, 2023.

[3] 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.

[4] A. Greiner-Petter, M. Fröbe, J. P. Wahle, T. Ruas, B. Gipp, A. Aizawa, M. Potthast, Overview of the Generative Plagiarism Detection Task at 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 Workshop Proceedings, CEUR-WS.org, 2025.

[5] 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.

[6] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (2019).

[7] K. Wang, W. Zeng, Y. Zhang, et al., Text embeddings by weakly-supervised contrastive pre-training, arXiv preprint arXiv:2212.09741 (2022).

[8] J. Johnson, M. Douze, H. Jégou, Billion-scale similarity search with gpus, 2019. ArXiv preprint arXiv:1702.08734.

[9] 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.