

# HALvest-Contrastive: Retrieval-like Authorship Attribution with Patch-level Late Interaction

**Francis Kulumba\***  
Inria Paris  
Sorbonne Université

**Wissam Antoun**  
Inria Paris  
Sorbonne Université

**Guillaume Vimont**  
IRIF

**Laurent Romary**  
Inria Paris

**Florian Cafiero**  
LRE, EPITA  
Ecole nationale des chartes – PSL

## Abstract

Authorship attribution is the process of identifying the writer of a disputed text by analyzing distinctive stylistic patterns. By comparing the disputed text against a baseline of known authorship, the task amounts to assessing how strongly the two texts are likely to share an author. However, topical confounding makes the task deceptively easy: two authors covering the same topic may look more alike than one author covering two topics. Scholarly prose offers a natural solution. Academic writers produce multiple papers on related but distinct topics while maintaining consistent stylistic habits. Thus, we introduce HALvest-Contrastive, an English benchmark that draws same-author passages from distinct papers within a disciplinary field to reduce topical overlap, while using different-author passages from the same field to discourage naive keyword matching. The benchmark’s design removes topical shortcuts: a strong lexical baseline fails under these constraints. As a by-product of our benchmark construction, we release the 17B-token multilingual corpus used to produce it. We then revisit how to score authorship: while standard systems compress each document into a single vector, we retain the sequence of token-level representations produced by a language model and compare them with late interaction. We also study patch-level late interaction, which groups neighboring tokens into patches before interaction. Matching at the token level increases Recall@20 by a factor of three to five. The optimal interaction granularity, however, depends on sequence length, with no single granularity performing best across all settings.

## 1 Introduction

Authorship Attribution (AA) asks whether a piece of text is stylistically consistent with a given author, typically by comparing it against text of known authorship. The field has a long statistical tradition

that treats style as a document-level property summarized by a frequency vector (Burrows, 2002; Schler et al., 2006; Treeratpituk and Giles, 2009). Neural approaches have largely inherited this view. A transformer encoder (Vaswani et al., 2017; Devlin et al., 2019) embeds a passage, the token representations are mean-pooled into a single vector, and training minimizes a triplet loss: a three-way comparison between an anchor, a same-author positive, and a different-author negative, with a fixed margin (Wegmann et al., 2022; Kantharuban et al., 2026). Information retrieval, a field that also contrasts documents against one another, has moved in a different direction. Dense retrieval has replaced triplet loss with InfoNCE (Oord et al., 2019; Gao et al., 2021), a contrastive objective that uses many in-batch negatives instead of one hard negative, and single-vector pooling has given way to late interaction (LI) (Khattab and Zaharia, 2020; Santhanam et al., 2022), where documents are represented by sets of token vectors and the score is the sum of maximum matches between said tokens (Figure 1). This paper asks what happens when the retrieval playbook is applied to AA, and where the two tasks diverge.

Topical confounding presents a primary obstacle. The AA field’s shift toward contrastive approaches aimed to minimize vocabulary overlap within each triplet (Wegmann and Nguyen, 2021) to mitigate it. Decreasing the discriminative signal from individual tokens may encourage models to learn subtler distributional cues such as stop-word frequency, sequence length, or overall tone. Academic papers’ structure supports this approach: authors write on related but distinct scholarly topics while maintaining consistent stylistic patterns. We construct HALvest, a 17-billion-token corpus from 778k open-access academic papers, and derive HALvest-Contrastive, a benchmark in English with controlled topic variation. HALvest-Contrastive maximizes topic decoupling. An anchor and posi-

\* Correspondence: francis.kulumba@inria.fr

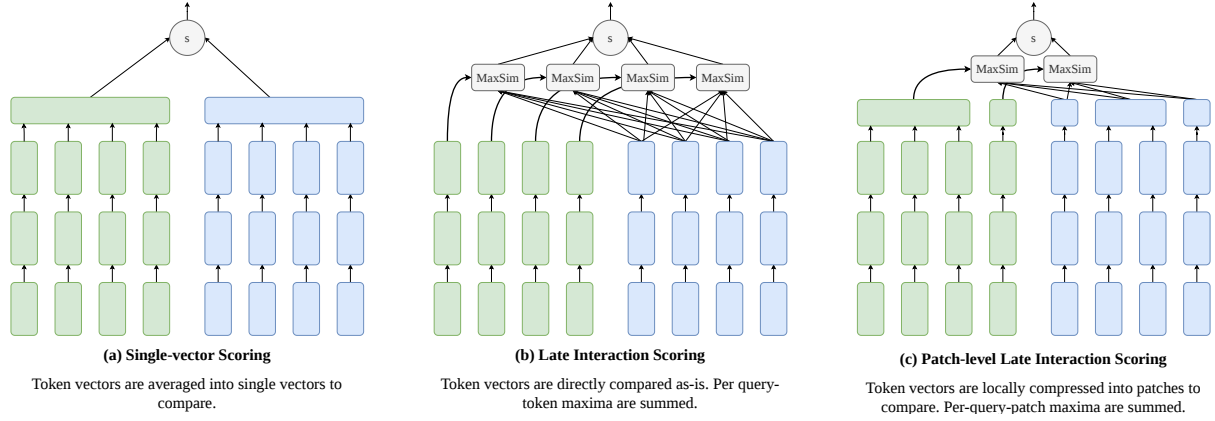


Figure 1: **Three scoring modes for authorship attribution.** The green and blue tiles represent token vectors from an anchor and a positive document, respectively. **(a) Single-vector pooling** averages token embeddings into one vector per document and scores pairs by cosine similarity. **(b) Late interaction** (Khattab and Zaharia, 2020) keeps the full sequence of token vectors and scores by MaxSim: for each query token, the maximum cosine similarity to any document token is computed, and the results are summed. **(c) Patch-level late interaction (PLI)** groups token vectors into contiguous patches, pools each patch into a single vector, and applies MaxSim at the patch level. We train and evaluate all three modes under a matched contrastive objective. The granularity comparison is the main focus of this paper.

tive are drawn from different papers written by the same author-set. The negative comes from within the same disciplinary field. A model cannot entirely succeed by learning topic vocabulary alone.

This design allows us to study how retrieval techniques transfer to authorship attribution, from the training objective to the interaction mechanism such as late interaction and its patch-level variant. The comparison is controlled: all models share the same encoder, the same data, and the same loss. Only the pooling and scoring granularity differ. Our contributions are as follows.

1. **HALvest**, a 17-billion-token multilingual corpus harvested from open-access academic papers. **HALvest-Contrastive**, the derived English benchmark for scholarly AA with a verified multi-author structure, controlled topic decoupling, and an evaluation structure by span length. HALvest-Contrastive targets authorship disambiguation in scholarly repositories. Researchers sharing a name within one field cannot be separated by metadata alone. The remaining signal must come from the text.
2. **Empirical validation of topic decoupling:** a strong lexical baseline succeeds on topic-rich data but fails on topic-decoupled data. The gap persists across span lengths.
3. **Granularity study:** mean pooling, token-level late interaction, and fixed-policy PLI variants are compared on matched data under a matched training objective, across five

span-length subsets of HALvest-Contrastive and PAN19 zero-shot.

4. **Empirically best fixed- $n$  patch size:** across our subsets and PAN19, the best patch size is well approximated by  $n_* \approx 0.18 \cdot \sqrt{S}$ , where  $S$  is the token count. This descriptive fit provides a practical starting point for practitioners.

HALvest is the upstream multilingual corpus from which HALvest-Contrastive is derived. The AA-specific contributions of this paper center on HALvest-Contrastive, topic-decoupling validation, and the granularity study. HALvest itself is released as a general-purpose scholarly resource. CamemBERTv2 (Antoun et al., 2024) has already adopted HALvest’s French split for continued pre-training.

The code<sup>1</sup> and the corpus itself<sup>2</sup> are released under an open license.

## 2 Related Work

Three literatures provide the closest precedents to our setup: stylometry, scholarly text corpora, and multi-vector retrieval.

### 2.1 Stylometry and authorship attribution

Stylometric AA has traditionally relied on frequency-based representations of writing style,

<sup>1</sup><https://github.com/Madjakul/DeepStylometry>

<sup>2</sup><https://huggingface.co/datasets/almanach/halvest-contrastive>

using features such as function words (Mosteller and Wallace, 1963; Burrows, 2002; Treeratpituk and Giles, 2009), part-of-speech n-grams (Argamon-Engelson et al., 1998), most frequent character n-grams (Kešelj et al., 2003), or combinations of those (Cafiero and Camps, 2019; Juola, 2015). Machine-learning treatments formalized the task in closed, verification, and open-set variants (Koppel et al., 2009; Seroussi et al., 2014). A recurring concern is topical confounding: Sapkota et al. (2015) showed that classifiers trained and tested on the same topic substantially overestimate generalization, and proposed cross-topic evaluation as a stricter protocol.

Large-scale neural AA systems have followed suit: contrastive learning has emerged as a strong paradigm for stylometry (Ai et al., 2022; Wegmann et al., 2022; Huertas-Tato et al., 2024). A prevailing strategy in this area has been to decorrelate an author’s style from literal text content (Altakrori et al., 2021). This includes developing edit-invariant loss functions to capture signals beyond surface-level text (Liu et al., 2022), or creating positive pairs with high semantic overlap to force models to learn distributional cues (Rivera-Soto et al., 2021; Wegmann and Nguyen, 2021; Alshomary et al., 2025). Most recently, Kantharuban et al. (2026) studied idiolect at scale.

We target scholarly prose, a low-entropy register with strong domain structure. Internet and literary text do not share these properties. We replace single-vector pooling and triplet loss with late interaction and InfoNCE (Oord et al., 2019). We ask whether the induced performance changes transfer (Figure 1). The InfoNCE objective has become a standard discriminative loss for representation learning: it learns embeddings by contrasting a positive pair against a pool of in-batch negatives. Chen et al. (2020) and Khosla et al. (2021) showed that large negative pools and strong augmentations suffice for self-supervised image representations. For our setting, they report that in-batch false negatives, pairs labeled as negative that in fact share a latent class, degrade performance only weakly.

## 2.2 Scholarly text corpora

Several corpora have been built around scientific writing. Prominent examples include domain-specific documents from ArXiv, DBLP, and PubMed (Sen et al., 2008; Doğan et al., 2014; Wahle et al., 2022). Other initiatives provide full-text access, such as the ACL Anthology for

computational linguistics (Bird et al., 2008), or S2ORC (Lo et al., 2020), notable for its scale and multi-domain coverage.

HALvest draws from a single open-access repository that spans multiple high-level domains. It supplies the author-set labels as they appear in the metadata used by real retrieval systems.

## 2.3 Late interaction and multi-vector retrieval

ColBERT (Khattab and Zaharia, 2020) introduced late interaction: documents are represented by a set of token embeddings and scored against a query via MaxSim, the sum over query tokens of the maximum cosine similarity to any document token. ColBERTv2 (Santhanam et al., 2022) added residual compression for storage. TRIAL (Kang et al., 2025) adds bigram-level token relation scores and per-query importance weights to MaxSim. These modifications target semantic retrieval where phrase coherence and content-word emphasis improve relevance estimation. Token pruning approaches (Zong and Piwowarski, 2025) learn to remove tokens before scoring. ColBERTer (Hofstätter et al., 2022) compresses tokens into coarser units, while ConstBERT (MacAvaney et al., 2025) pools token embeddings into a fixed-size set of learned vectors, functioning as a learned-patch reranker.

These methods are conceptually the closest precedents to our patch-level view. They target semantic retrieval or generative modeling. AA is not their focus.

## 3 HALvest and HALvest-Contrastive

A stylometry benchmark must specify the text’s source and define how to construct same-author and different-author pairs so that the resulting signal isolates style from topic. This section addresses both questions.

**Why scholarly prose?** Scholarly writing provides an under-exploited testbed for stylometric research. Academic prose is low-entropy: its vocabulary is tightly constrained, its syntactic templates are strongly standardized, and its structure recurs across papers within a field. This compresses surface variation and isolates the stylistic residual from topical noise. The resulting benchmark targets a practical disambiguation task: in repositories such as HAL, Semantic Scholar, or DBLP, researchers who share a name within the same field cannot be reliably distinguished by metadata alone.

In these cases, stylometric signal remains, and the method we study applies directly.

### 3.1 Source and processing pipeline.

HAL is an open-access repository for French and international scholarly output. We extract full-text PDFs for all papers with permissive licenses. We process each PDF with GROBID ([GROBID Repository](#)) to recover structured XML and serialize the result to plain text. HALvest covers 778k documents in 56 languages across 16 disciplinary domains.

**From HALvest to HALvest-Contrastive.** We construct the contrastive dataset by applying a sequence-level filter that restricts to English spans. This yields HALvest-Contrastive, the English-only triplet dataset used for all models in this paper. We further filter to the 13 domains with sufficient document counts for reliable hard-negative mining (Appendix B). We remove spans with excessive symbols, abnormal layout, code fragments or HTML markup, high uppercase ratios, or anomalous sentence-length variance. Mining begins with three configurations. We scale the best configuration to millions of rows.

The **unrestricted** configuration samples five random spans per query document. Positives come from all documents with the exact same author-set. If no other document exists for that set, we sample positive spans from the query document itself. Hard negatives come from documents that share no authors with the query but have the same HAL domain label. This configuration permits high topical overlap between query and positive and serves as a topical control.

The **base** (restricted) configuration enforces strict topic decoupling: positives come exclusively from different documents by the same author-set, never from the query document itself. We scale and release the base configuration as HALvest-Contrastive. Table 1 gives the total number of triplets. A third **inverse cloze task** (ICT) configuration (Lee et al., 2019) treats a random span from a document as the anchor and the surrounding context as the positive, with a non-overlapping passage from the same document as the negative. ICT supplies a style-aware retrieval signal that is not purely lexical and not fully topic-decoupled. We use ICT as an intermediate baseline in §5.1.

**Triplet construction.** Within each configuration, a triplet consists of three spans, each contain-

| # Sentences | Train | Valid | Test  |
|-------------|-------|-------|-------|
| 2           | 1.88M | 19.1k | 19.1k |
| 4           | 1.4M  | 14.3k | 14.3k |
| 6           | 1.11M | 11.3k | 11.3k |
| 8           | 892k  | 9.1k  | 9.1k  |

Table 1: **HALvest-Contrastive triplet counts.** Each row corresponds to a span-length subset ( $k$  contiguous sentences per span).

ing  $k$  contiguous sentences. The span length  $k$  parametrizes task difficulty: shorter spans provide less textual evidence, and AA accuracy increases monotonically with  $k$ . The dataset supports  $k \in \{2, 4, 6, 8, 10\}$ . Table 1 gives training counts for the four subsets used for training, with the base-10 counts included in the release. We report  $k = 4$  as the primary split and provide full per- $k$  results in Appendix D. Training uses base configuration triplets throughout. Evaluation reports both base and unrestricted configurations where relevant to the argument.

**Author-set as the label.** The label of a triplet is its set of co-authors. We use author-sets because they match the metadata exposed by scholarly repositories and avoid imposing an arbitrary single-author heuristic. This framing aligns with real-world settings. About 51% of our triplets are already single-author, so the label covers the common case while preserving collaborative papers without forcing a noisy reduction to one writer. Forensic linguistics supports this approach. In the general case, we cannot know which listed co-authors contributed as writers or provided ideas or feedback. Changing the author-set definition would alter the task itself. We hold the label constant across all experimental conditions.

**Cross-domain evaluation via PAN19.** We use PAN19 (Kestemont et al., 2019) as a zero-shot cross-domain evaluation. PAN19 labels documents at the individual-author level and draws texts from fan-fiction. Both corpora are English, but the registers differ. Academic prose is heavily normalized by journal style guides. Fan-fiction exhibits broader variation in sentence length, punctuation, and paragraph structure. A model that generalizes across both the label granularity gap (author-set to individual author) and the register gap (academic to fan-fiction) cannot be a topic classifier in disguise.

## 4 Contrastive Modelling

We separate two design choices that the AA literature has historically bundled together. We select InfoNCE as the training objective in place of triplet loss. We then define the three pooling and interaction families we compare in our experiments.

We adopt standard retrieval terminology throughout. A *triplet* is an ordered tuple of three passages  $(a, p, n)$ , where  $a$  (the *anchor*) and  $p$  (the *positive*) share authorship, and  $n$  (the *negative*) does not. A contrastive loss pulls  $a$  and  $p$  close in embedding space while pushing  $a$  away from  $n$ . We evaluate models with the standard ranking metrics Recall@ $k$  (the fraction of anchors whose correct positive is in the top- $k$  retrieved candidates) and nDCG@ $k$  (a graded, position-aware variant). For zero-shot cross-domain evaluation, we use the *triplet accuracy* (the fraction of triplets for which  $s(a, p) > s(a, n)$ ).

### 4.1 A simple look on InfoNCE

Given an anchor  $a$ , a positive  $p$  from the same author-set, and a pool of in-batch negatives  $\mathcal{N}$  from other author-sets, we train to minimize the InfoNCE loss.

$$\mathcal{L} = -\log \frac{\exp(s(a, p)/\tau)}{\exp(s(a, p)/\tau) + \sum_{n \in \mathcal{N}} \exp(s(a, n)/\tau)} \quad (1)$$

The pooling-specific similarity score  $s(\cdot, \cdot)$  is cosine for single-vector pooling and MaxSim for late interaction and PLI (Figure 1). The temperature parameter  $\tau$  is set to 0.5 throughout. We use full-gather across GPUs. The candidate pool at each anchor grows with the total batch size. Each triplet yields a positive and a mined negative, and both are gathered. Four GPUs at a batch size of 32 per device give a pool of 256 candidates per anchor. We further scale the batch size to 64. Larger candidate pools yield measurable gains, as we show in §5. This finding is consistent with the retrieval literature (Karpukhin et al., 2020; Chen et al., 2020).<sup>3</sup>

**InfoNCE gradient** naturally concentrates on hard negatives, which aligns with suppressing topical shortcuts in authorship attribution. Writing  $s_n = s(a, n)$  for a generic negative, the partial derivative of Equation 1 with respect to  $s_n$  is

$$\frac{\partial \mathcal{L}_{\text{InfoNCE}}}{\partial s_n} \propto \frac{\exp(s_n/\tau)}{\exp(s_p/\tau) + \sum_{n' \in \mathcal{N}} \exp(s_{n'}/\tau)} \quad (2)$$

<sup>3</sup>We chose InfoNCE over triplet loss early in development. Preliminary hyperparameter sweeps showed that triplet loss lagged 10–20 percentage points behind InfoNCE on retrieval metrics, consistent with the gradient analysis below. We adopted InfoNCE for all reported experiments.

This quantity is the softmax weight of  $n$  in the contrastive denominator. The gradient concentrates on highly similar negatives according to the model’s current state, without an explicit mining step. Early in training, the hardest negatives for a pre-trained encoder are the topically similar ones provided in the triplet. InfoNCE allocates gradient to decorrelating style from topic. This direction aligns with AA’s requirements. Appendix A gives a full derivation of Equation 2.

### Multi-positive evaluation under single-positive training.

Our training mines one positive per anchor, but at evaluation time some anchors have multiple positives, because author-sets recur across documents. In-batch “false negatives” (other valid positives that happen to appear in the batch) can therefore arise. We treat these cases as benign: Chen et al. (2020) report that in-batch false negatives only weakly degrade performance, and Khosla et al. (2021) find that aggregating multiple positives improves representation quality in supervised contrastive learning.

### 4.2 Pooling and interaction

We compare three families of similarity functions, each determining how the representations produced by the encoder are combined into a document score. The first two are not our constructions. Mean pooling under a contrastive objective is the readout used by the contrastive attribution systems that define the current state of the art (Rivera-Soto et al., 2021; Huertas-Tato et al., 2024; Wegmann et al., 2022), and layerwise attention pooling is the readout of IDIOLEX (Kantharuban et al., 2026). We reimplement both under the same encoder, corpus, and loss, so that the comparison isolates the readout.

**Mean pooling.** The mean of its token embeddings represents the document, and  $s(a, p)$  is the cosine similarity between the two mean vectors. This is also the default readout for general-purpose sentence encoders. We additionally trained the layerwise attention pooling variant, with and without centering. Validation-set performance only marginally outperformed mean pooling, and we discarded it before test-set evaluation.

**Late interaction.** The document is represented by its full sequence of token embeddings, and  $s(a, p) = \sum_i \max_j \cos(\mathbf{h}_i^a, \mathbf{h}_j^p)$  is the MaxSim score of Khattab and Zaharia (2020), with punctuation and padding positions masked out. This

representation is used at both train and test time; the scoring function is identical to the training objective.

**Patch-level late interaction (PLI).** We group the token sequence into contiguous patches. Each group is mean-pooled into a single patch vector. We compute MaxSim between the two resulting patch sequences. PLI reduces the per-document vector count by the average patch length. The granularity of interaction becomes an explicit design choice. Both the memory footprint and the MaxSim scoring cost drop roughly in proportion to the patch size. We study two fixed-policy families.

**$n$ -gram PLI.** Patches are fixed-length, non-overlapping windows of  $n$  tokens. We evaluate  $n \in \{2, 3, 4, 5\}$ . This policy is parameter-free but ignores whitespace and word boundaries so that a single linguistic word can be split across patches.

**Whole-word PLI.** Patch boundaries align with word starts. Each patch corresponds to one linguistic word, so the representation is invariant to how the word is broken into sub-word units.

**What about learned patches?** We make the patch boundaries trainable by adding a small head that emits a per-token cut probability. The boundaries are sampled with a Gumbel-Softmax straight-through estimator (Jang et al., 2017; Maddison et al., 2017). This setup allows the patching policy to be trained jointly with the encoder under InfoNCE. We implemented this approach and ran it under two single-term regularizers. Both failed in characterizable ways. The resulting model never reached the fixed- $n$  baselines. Appendix G gives the mathematical analysis, the empirical failure-mode characterization, and the requirements for a working multi-term regularizer. This negative result delineates the boundary of what fixed-policy PLI can achieve.

## 5 Experiments

We evaluate each trained model with two complementary views. The first is retrieval on HALvest-Contrastive, reporting Recall@20, Recall@100, nDCG@20, and nDCG@100, with the scorer matching the training objective: cosine for mean pooling, token-level MaxSim for late interaction, and patch-level MaxSim for PLI. The second is triplet accuracy on PAN19 zero-shot.

### 5.1 Topic decoupling

The base configuration of HALvest-Contrastive prevents models from learning topical shortcuts. We verify empirically that the design achieves this goal. Two pieces of evidence support the claim. We measure the Jaccard overlap between the word

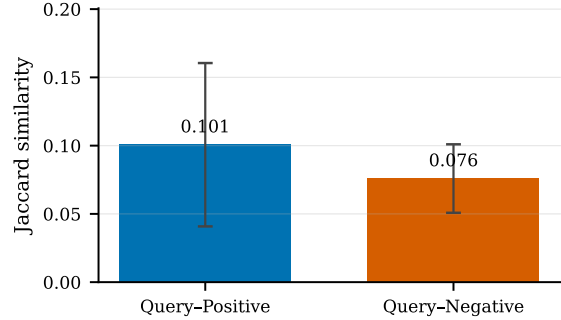


Figure 2: **Vocabulary overlap in HALvest-Contrastive triplets.** Mean Jaccard similarity between the word sets of query-positive and query-negative pairs, computed on 10,000 sampled base-configuration triplets.

sets of triplet components on a sample of 10,000 triplets. Figure 2 reports the results.

A purely lexical retriever gives us a direct test. If the base configuration has successfully decoupled topic, BM25 (Robertson et al., 1994) should succeed on the unrestricted data (where the positive can share words with the query) and fail on the base data. Table 2 reports BM25 accuracy on both configurations across span lengths, alongside the ICT baseline and a RoBERTa (Liu et al., 2019) contrastive model trained on each configuration. None of the configurations at this stage is scaled to its full size. The comparison justifies the design choice for HALvest-Contrastive. On four-sentence spans, BM25 achieves 79.26% accuracy on unrestricted and 68.36% on base, a 10.9-point drop. The gap holds between 10.9 and 11.3 points at every span length beyond two sentences. The base configuration remains harder even where BM25 has twice as many content words. The ICT model consistently outperforms BM25, especially on the base set. Its retrieval objective learns more abstract representations. The RoBERTa contrastive model trained with triplets where the anchor and positive originate from different documents achieves top performance across all conditions. We scaled the base data afterward and made it HALvest-Contrastive.

| Model               | $k = 2$  |              | $k = 4$  |              | $k = 6$  |              | $k = 8$  |              |
|---------------------|----------|--------------|----------|--------------|----------|--------------|----------|--------------|
|                     | Unrestr. | Base         | Unrestr. | Base         | Unrestr. | Base         | Unrestr. | Base         |
| BM25                | 68.15    | 60.24        | 79.26    | 68.36        | 85.03    | 73.72        | 89.09    | 78.21        |
| ICT                 | 86.61    | 79.72        | 91.60    | 85.11        | 93.77    | 88.51        | 94.41    | 88.07        |
| RoBERTa-UnrestrData | —        | 80.73        | —        | 87.03        | —        | 88.44        | —        | 89.91        |
| RoBERTa-BaseData    | —        | <b>82.33</b> | —        | <b>88.07</b> | —        | <b>90.15</b> | —        | <b>90.83</b> |

Table 2: **Topic-decoupling validation.** Accuracy (%) on the unrestricted and base test configurations of HALvest-Contrastive across span lengths  $k$ . All models are compared at 3.1k training steps for fairness.

## 5.2 Effect of span length

Accuracy rises monotonically with the span length  $k$  for every model in Table 2. The slopes differ: BM25 gains 18 points from  $k=2$  to  $k=8$  on the base configuration, while RoBERTa-BaseData gains only 8.5 points and plateaus by  $k=6$ .

Neural models thus capture most of the authorial signal already at short spans. The lexical signal keeps growing with span length (Appendix I). All subsequent experiments use the base configuration at  $k = 4$  as the primary reporting split, with full per- $k$  tables in Appendix D.

## 5.3 Granularity across span lengths

We trained ModernBERT (Warner et al., 2025) with InfoNCE for one epoch under each of the granularities introduced in §4.2: mean pooling, late interaction, whole-word PLI, and fixed- $n$  PLI for  $n \in \{2, 3, 4, 5\}$ . We ran each model on every subset of HALvest-Contrastive ( $k \in \{2, 4, 6, 8, 10\}$ ). Figure 3 reports the four ranking metrics across the five subsets for the multi-vector configurations. The tables with all values are in Appendix D. We exclude mean pooling from the figure to keep the y-axis readable. Its values sit far below the multi-vector cluster at every subset.

Under a matched training objective, moving from single-vector pooling to late interaction multiplies Recall@20 by four on base-4, from 0.121 to 0.485 (Table 9), a proportionally larger gain than the one reported for passage retrieval under the same change of scorer (Khattab and Zaharia, 2020).

We can also apply MaxSim at inference to an encoder fine-tuned under mean pooling, without retraining it. The “Mean pooling (LI)” row of Table 9 gains roughly 20% in relative Recall@20 over plain mean pooling, but in absolute terms this closes 0.026 of the 0.364 gap, about 7%. The scoring operation applied to a fixed encoder accounts

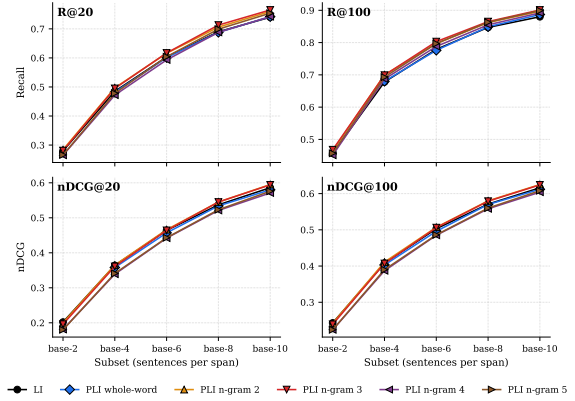


Figure 3: **Retrieval metrics across HALvest-Contrastive subsets.** Each panel reports one of four ranking metrics (R@20, R@100, nDCG@20, nDCG@100); each marker family corresponds to one model. The x-axis labels denote the subset (base- $k = k$  contiguous sentences per anchor, positive, and negative span). Mean pooling is omitted for readability (its values lie far below, see Appendix D).

for a small part of the advantage. Most of the gain comes from the encoder that training under MaxSim produces. The same swap applied to E5, an off-the-shelf encoder with an XLM-RoBERTa backbone (Conneau et al., 2020), raises Recall@20 from 0.097 to 0.179 on PAN19 (Appendix H), a factor of 1.85. The direction of the effect at inference is not specific to ModernBERT. Within the multi-vector family, no fixed policy is uniformly best across subsets. Token-level LI is competitive but not dominant, whole-word PLI sits in the middle of the cluster, and the n-gram variants reorder as  $k$  grows.

We also report alignment and uniformity (Wang and Isola, 2020) as a representation-quality diagnostic (Appendix E). Within the fixed- $n$  PLI family, the two quantities trade off along a clean monotonic axis as  $n$  grows. Across families the diagnostic does not track retrieval. Mean pooling attains both a lower alignment loss and a lower uniformity loss

than PLI  $n=2$ , and still retrieves four times worse on base-4. The separation between the two readouts is not visible in the geometry of the pooled representation space. We report the pair as a diagnostic, not as an explanation.

#### 5.4 AA is not semantic retrieval

To test whether the gains come from semantic retrieval alone, we compute the cosine similarities between anchors and positives and between anchors and negatives using both E5 (Wang et al., 2024) and our fine-tuned mean-pooled model on a sample of 4,000 triplets (Kantharuban et al., 2026). If the two signals were equivalent, they would similarly separate same-author and different-author pairs. Figure 4 shows that neither is the case. On

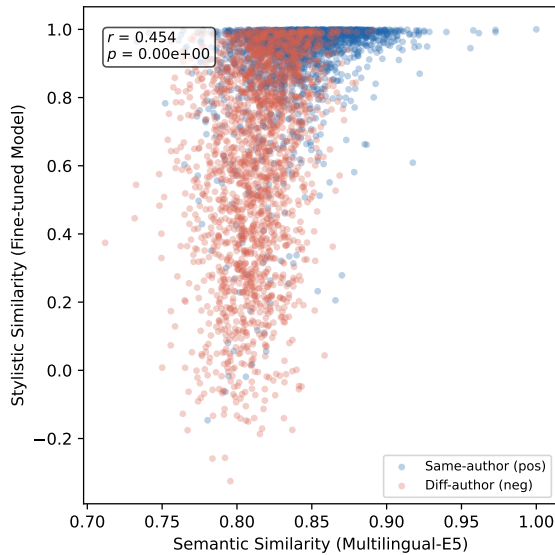


Figure 4: **Stylistic vs. semantic similarity.** Each point is one anchor-candidate pair from HALvest-Contrastive base-4 ( $N=4,000$ ). The x-axis measures the cosine similarity with a multilingual-E5 encoder (semantic), while the y-axis uses our fine-tuned stylistic model (mean pooling). Same-author pairs are in blue and different-author pairs in red.

the stylistic axis, our fine-tuned model separates same-author pairs from different-author pairs by a margin of 0.30 in mean cosine similarity. On the semantic axis, E5 separates the same populations by only 0.029. Semantic similarity under E5 is almost invariant to the authorship relation between two passages. The stylistic representation responds to it strongly. The pooled Pearson correlation across all 4,000 pairs is  $r = 0.454$ , which may suggest partial agreement between the two signals. This coefficient is dominated by the between-group shift

we just described. The within-group structure in the scatter shows stylistic similarity varying across the full range while semantic similarity stays confined in a narrow vertical band.

An error analysis of the strongest PLI baseline (Appendix F) reveals that highly ranked distractors share domain labels and partial author overlap with the query but not lexical similarity, a pattern structurally different from a topic-retrieval failure mode.

#### 5.5 Cross-domain and patch-size scaling

| Model           | Triplet accuracy |
|-----------------|------------------|
| Mean pooling    | 0.520            |
| LI (full token) | 0.719            |
| PLI $n$ -gram 2 | <b>0.740</b>     |
| PLI $n$ -gram 3 | 0.724            |
| PLI $n$ -gram 4 | <b>0.740</b>     |
| PLI $n$ -gram 5 | 0.704            |
| PLI whole-word  | 0.715            |
| E5 zero-shot    | 0.628            |

Table 3: **Zero-shot transfer to PAN19.** Triplet accuracy on PAN19 for all trained models, evaluated without any further fine-tuning.

Table 3 reports triplet accuracy on PAN19 for every trained model, evaluated zero-shot. The scorer effect replicates on a benchmark drawn from a different genre, with a disjoint author population and individual-author labels. The pattern within the fixed-policy family matches what we saw on HALvest-Contrastive’s longer subsets, where token-level LI is competitive but not best and whole-word sits in the middle.

**An empirical patch-size fit.** The shift in optimal  $n$  across subsets is well approximated by a one-parameter square-root fit. Let  $S$  denote the token count of an anchor, which varies by subset because longer- $k$  subsets fill more of the ModernBERT context (512 tokens). Across six subsets with span length ranging from 69.5 tokens to 512.0 tokens, the best fixed  $n$  is fitted by

$$n_{\star} \approx 0.18 \cdot \sqrt{S}. \quad (3)$$

The relationship is empirical, not derived. It also explains the apparent disagreement between rankings in Figure 3 for a given subset. Each subset samples a different point on the same curve. We

| Subset            | $S$   | $0.18\sqrt{S}$ | $n_*$ |
|-------------------|-------|----------------|-------|
| HALvest-C base-2  | 69.5  | 1.50           | 1 / 2 |
| HALvest-C base-4  | 133.2 | 2.08           | 2     |
| HALvest-C base-6  | 201.4 | 2.55           | 3     |
| HALvest-C base-8  | 266.2 | 2.94           | 3     |
| HALvest-C base-10 | 327.6 | 3.26           | 3     |
| PAN19 (zero-shot) | 512.0 | 4.07           | 4     |

Table 4: **Empirical fit of  $n_*$ .** For each subset,  $S$  is the mean token count and  $n_*$  is the fixed patch size with the highest Recall@20, where  $n=1$  denotes token-level LI. Two entries are tied within 0.004 Recall@20. The limited resolution of our patch sizes means ties of this kind are expected.

treat Equation (3) as a descriptive fit over the range we tested. It tells a practitioner who has chosen a context length what fixed patch size to start at, but does not say why  $\sqrt{\cdot}$  is the right exponent. Under that empirical fit, the best-performing patch size grows sublinearly with sequence length. The number of stored patch vectors also grows sublinearly, approximately proportional to  $\sqrt{S}$  over the range we tested. Patch-level interaction gives a cheaper alternative to full token-level LI while preserving most of the gain on AA.

Scaling the candidate pool from 256 to 512 yields small but consistent gains across all metrics, matching the standard retrieval scaling pattern (Karpukhin et al., 2020).

## 6 Conclusion

Scholarly prose forms a strong testbed for stylometry because its low-entropy register compresses topical variation and makes stylistic cues easier to isolate. HALvest-Contrastive exploits this structure by pairing same-author passages across different papers within a disciplinary field. The resulting benchmark reduces lexical shortcuts and sets a practical task: authorship disambiguation in scholarly repositories where metadata alone is insufficient.

The transfer from retrieval to authorship attribution varies across design choices. We find that moving from single-vector pooling to multi-vector interaction yields large gains. Full token-level matching does not always improve performance. Grouping neighboring tokens into patches maintains most of the retrieval improvement and reduces the number of stored vectors and pairwise MaxSim comparisons. Across both HALvest-Contrastive and PAN19, the best-performing fixed patch size

increases gradually with sequence length and is well approximated, over the range we tested, by  $n_* \approx 0.18 \cdot \sqrt{S}$ .

More broadly, the scoring function sets the effective depth of the retrieval pipeline. Under a matched encoder, corpus, and objective, it moves Recall@20 by a factor of four on base-4, and by between three and five across our five splits, more than the margins usually reported for changes of architecture or pre-training data. Authorship attribution departs from semantic retrieval in its interaction scale. Semantic matching benefits from fine-grained lexical alignment. Stylometric similarity emerges at an intermediate interaction scale.

## Limitations

### Author-set rather than individual attribution.

The triplet label is the set of co-authors rather than a single writer. We argue in §3 that this is the correct target for scholarly repositories and that the forensic-linguistics literature supports it as a principled design choice rather than a concession (Dauber et al., 2019). Author-set labels represent a reliable level of supervision already available in scholarly metadata. Individual attribution within multi-author documents is a different task, and we leave it to future work. The concern about noise is empirically bounded: 51% of our triplets are already single-author, so multi-author effects apply to at most half the data, and the failure-mode analysis in Appendix F confirms that partial author overlap is exactly where the model struggles, consistent with a genuine stylistic signal rather than label noise.

**GROBID.** HALvest’s text is produced by running GROBID over PDFs. More accurate extraction methods exist: OCR and vision-language models. GROBID’s layout-based extraction preserves document structure essential for our contrastive sampling. GROBID can introduce systematic biases: Arabic scripts are inverted (left-to-right assumption), CJK coverage is sparser due to whitespace-based filtering, and mathematical notation is often stripped. As a rough estimate, pre-processing of HAL’s French split (Antoun et al., 2024) retained about 52% of tokens after formula handling.

**Learned patching is unfinished.** We attempted an end-to-end trainable patching policy but found that simple regularizers either collapse the policy

to token-level scoring (when the regularizer is unbounded below) or to single-patch sequences (when constraining only the population mean of cut probabilities below the inference threshold). See Appendix G for the mathematical analysis and the empirical failure-mode characterization. A successful learned variant would require multi-term regularization in the BLT style (Pagnoni et al., 2025), with hyperparameter tuning beyond our compute budget for this contribution. The cross-attention compressor variant of Pagnoni et al. (2025) was explored alongside it and shares the same regularizer-design weaknesses.

**Empirical scaling fit.** The  $0.18 \cdot \sqrt{S}$  relationship in §5.5 is descriptive. Six points across two registers fit a one-parameter curve, but at three of the six the best patch size is separated from its neighbor by less than 0.004 Recall@20, and at PAN19 two patch sizes tie exactly, so the six points constrain the curve less tightly than the point estimates suggest. We do not provide a derivation, and we do not yet know whether the same coefficient transfers to longer context lengths. It should be read as a practical summary for choosing a starting patch size.

**No generative baselines.** Our comparison covers contrastive encoder methods, which is the current state of the art in retrieval-style AA. Prompting a large decoder for AA defines a different paradigm. The decoder produces a verdict per pair rather than an embedding. It does not index a corpus. It does not expose a scoring granularity to vary. Our question is where in the interaction hierarchy stylistic signal is best matched. A generative verifier has no corresponding control and does not isolate the variable we study. Whether generative verification outperforms multi-vector retrieval on HALvest-Contrastive remains an open question.

## Ethics

**Licensing.** HAL is an open-access repository. The documents in HALvest were selected from papers deposited under permissive licenses compatible with derivative redistribution.

**Author identity.** The author labels in HALvest are exactly those that appear in the public metadata of the source papers. No personally identifiable information beyond what is already published is introduced by the derivative corpus. Author names are tied to signed, publicly archived academic works;

the corpus does not include private correspondence or unpublished drafts.

**Dual-use considerations.** Stylometric authorship attribution is a dual-use technology. It enables beneficial applications: plagiarism detection, LLM-generated-text detection, and verification of anonymous peer review integrity. The same methods can in principle be used to deanonymize pseudonymous writers, including whistleblowers and politically vulnerable authors. HALvest contains only signed academic work. The corpus-specific deanonymisation risk is low. Models trained on HALvest may be portable. Users deploying HALvest-trained models in downstream tasks should consider the risk surface of their own application domain.

## Acknowledgments

The authors are grateful to the CCSD staff, the service unit in charge of HAL, in particular Achraf Azhar. We also thank Patrice Lopez for providing resources and support to better handle GROBID, as well as Arij Riabi, Brahim Talb and Menel Mahamdi for the productive discussions. We extend our thanks to Yannis Karmim and Lydia Nishimwe for their proofreading. This work was partially realized on computing HPC and storage resources provided by IDRIS thanks to the grant GCDA1016807 on the DALIA supercomputer.

## References

- Bo Ai, Yuchen Wang, Yugin Tan, and Samson Tan. 2022. [Whodunit? learning to contrast for authorship attribution](#). In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing, Volume 1: Long Papers*, pages 1142–1157, Online only. Association for Computational Linguistics.
- Milad Alshomary, Nikhil Reddy Varimalla, Vishal Anand, Smaranda Muresan, and Kathleen McKeown. 2025. [Layered insights: Generalizable analysis of human authorial style by leveraging all transformer layers](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 10279–10292, Suzhou, China. Association for Computational Linguistics.
- Malik Altakrori, Jackie Chi Kit Cheung, and Benjamin C. M. Fung. 2021. [The topic confusion task: A novel evaluation scenario for authorship attribution](#). In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4242–4256, Punta Cana, Dominican Republic. Association for Computational Linguistics.

- Wissam Antoun, Francis Kulumba, Rian Touchent, Éric de la Clergerie, Benoît Sagot, and Djamé Seddah. 2024. [CamemBERT 2.0: A smarter french language model aged to perfection](#). *Preprint*, arXiv:2411.08868.
- Shlomo Argamon-Engelson, Moshe Koppel, and Galit Avneri. 1998. Style-based text categorization: What newspaper am i reading? In *Proceedings of the AAAI Workshop on Learning for Text Categorization*.
- Steven Bird, Robert Dale, Bonnie Dorr, Bryan Gibson, Mark Joseph, Min-Yen Kan, Dongwon Lee, Brett Powley, Dragomir Radev, and Yee Fan Tan. 2008. [The ACL anthology reference corpus: A reference dataset for bibliographic research in computational linguistics](#). In *Proceedings of the Sixth International Conference on Language Resources and Evaluation*, Marrakech, Morocco. European Language Resources Association.
- John Burrows. 2002. [Delta: A measure of stylistic difference and a guide to likely authorship](#). *Literary and Linguistic Computing*, 17(3):267–287.
- Florian Cafiero and Jean-Baptiste Camps. 2019. [Why molière most likely did write his plays](#). *Science Advances*, 5(11):eaax5489.
- Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. [A simple framework for contrastive learning of visual representations](#). In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *ICML '20*, pages 1597–1607. JMLR.org.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. [Unsupervised cross-lingual representation learning at scale](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.
- Edwin Dauber, Aylin Caliskan, Richard Harang, Gregory Shearer, Michael Weisman, Frederica Nelson, and Rachel Greenstadt. 2019. [Git blame who? stylistic authorship attribution of small, incomplete source code fragments](#). *Proceedings on Privacy Enhancing Technologies*, 2019(3):389–408.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1: Long and Short Papers*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. [NCBI disease corpus: A resource for disease name recognition and concept normalization](#). *Journal of Biomedical Informatics*, 47:1–10.
- William Falcon and The PyTorch Lightning team. 2019. [PyTorch Lightning](#).
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. [SimCSE: Simple contrastive learning of sentence embeddings](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- GROBID Repository. 2008–2024. GROBID. <https://github.com/kermitt2/grobid>.
- Sebastian Hofstätter, Omar Khattab, Sophia Althammer, Mete Sertkan, and Allan Hanbury. 2022. [Introducing neural bag of whole-words with ColBERTer: Contextualized late interactions using enhanced reduction](#). In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management, CIKM '22*, pages 737–747, New York, NY, USA. Association for Computing Machinery.
- Javier Huertas-Tato, Adrián Girón-Jiménez, Alejandro Martín, and David Camacho. 2024. [Isolating authorship from content with semantic embeddings and contrastive learning](#). *arXiv preprint*. ArXiv:2411.18472 [cs].
- Eric Jang, Shixiang Gu, and Ben Poole. 2017. [Categorical reparameterization with gumbel-softmax](#). In *International Conference on Learning Representations*.
- Patrick Juola. 2015. The rowling case: a proposed standard analytic protocol for authorship questions. *Digital Scholarship in the Humanities*, 30(suppl\_1):i100–i113.
- Hyukkyu Kang, Injung Kim, and Wook-Shin Han. 2025. [TRIAL: Token relations and importance aware late-interaction for accurate text retrieval](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 16864–16877, Suzhou, China. Association for Computational Linguistics.
- Anjali Kantharuban, Aarohi Srivastava, Fahim Faisal, Orevaghene Ahia, Antonios Anastasopoulos, David Chiang, Yulia Tsvetkov, and Graham Neubig. 2026. [IDIOLEX: Unified and continuous representations for idiolectal and stylistic variation](#). *arXiv preprint*. ArXiv:2604.04704 [cs].
- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. [Dense passage retrieval for open-domain question answering](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 6769–6781, Online. Association for Computational Linguistics.
- Vlado Kešelj, Fuchun Peng, Nick Cercone, and Calvin Thomas. 2003. N-gram-based author profiles for authorship attribution. In *Proceedings of the Conference of the Pacific Association for Computational Linguistics*, pages 255–264.

- Mike Kestemont, Efstathios Stamatatos, Enrique Manjavacas, Walter Daelemans, Martin Potthast, and Benno Stein. 2019. [PAN19 authorship analysis: Cross-domain authorship attribution](#).
- Omar Khattab and Matei Zaharia. 2020. [ColBERT: Efficient and effective passage search via contextualized late interaction over BERT](#). In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, pages 39–48, New York, NY, USA. Association for Computing Machinery.
- Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2021. [Supervised contrastive learning](#). *Preprint*, arXiv:2004.11362.
- Moshe Koppel, Jonathan Schler, and Shlomo Argamon. 2009. [Computational methods in authorship attribution](#). *Journal of the American Society for Information Science and Technology*, 60(1):9–26.
- Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. [Latent retrieval for weakly supervised open domain question answering](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.
- Guangyi Liu, Zichao Yang, Tianhua Tao, Xiaodan Liang, Junwei Bao, Zhen Li, Xiaodong He, Shuguang Cui, and Zhiting Hu. 2022. [Don't take it literally: An edit-invariant sequence loss for text generation](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2055–2078, Seattle, United States. Association for Computational Linguistics.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. [RoBERTa: A robustly optimized BERT pretraining approach](#). *Preprint*, arXiv:1907.11692.
- Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. [S2ORC: The semantic scholar open research corpus](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, Online. Association for Computational Linguistics.
- Sean MacAvaney, Antonio Mallia, and Nicola Tonelotto. 2025. [Efficient constant-space multi-vector retrieval](#). In *Advances in Information Retrieval: 47th European Conference on Information Retrieval, ECIR 2025, Lucca, Italy, April 6–10, 2025, Proceedings, Part III*, pages 237–245, Berlin, Heidelberg. Springer-Verlag.
- Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. 2017. [The concrete distribution: A continuous relaxation of discrete random variables](#). In *International Conference on Learning Representations*.
- Frederick Mosteller and David L Wallace. 1963. Inference in an authorship problem: A comparative study of discrimination methods applied to the authorship of the disputed federalist papers. *Journal of the American Statistical Association*, 58(302):275–309.
- Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. [Representation learning with contrastive predictive coding](#). *arXiv preprint*. ArXiv:1807.03748 [cs, stat].
- Artidoro Pagnoni, Ramakanth Pasunuru, Pedro Rodriguez, John Nguyen, Benjamin Muller, Margaret Li, Chunting Zhou, Lili Yu, Jason E. Weston, Luke Zettlemoyer, Gargi Ghosh, Mike Lewis, Ari Holtzman, and Srini Iyer. 2025. [Byte latent transformer: Patches scale better than tokens](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics, Volume 1: Long Papers*, pages 9238–9258, Vienna, Austria. Association for Computational Linguistics.
- Rafael A. Rivera-Soto, Olivia Elizabeth Miano, Juanita Ordonez, Barry Y. Chen, Aleem Khan, Marcus Bishop, and Nicholas Andrews. 2021. [Learning universal authorship representations](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 913–919, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. [Okapi at trec-3](#). In *Text Retrieval Conference*.
- Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2022. [ColBERTv2: Effective and efficient retrieval via lightweight late interaction](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3715–3734, Seattle, United States. Association for Computational Linguistics.
- Upendra Sapkota, Steven Bethard, Manuel Montes, and Tamar Solorio. 2015. [Not all character n-grams are created equal: A study in authorship attribution](#). In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–102, Denver, Colorado. Association for Computational Linguistics.
- Jonathan Schler, Moshe Koppel, Shlomo Argamon, and James W. Pennebaker. 2006. Effects of age and gender on blogging. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, volume 6, pages 199–205.
- Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. 2008. [Collective classification in network data](#). *AI Magazine*, 29(3):93–93. Number: 3.
- Yanir Seroussi, Ingrid Zukerman, and Fabian Bohnert. 2014. [Authorship attribution with topic models](#). *Computational Linguistics*, 40(2):269–310.

Pucklada Treeratpituk and C. Lee Giles. 2009. [Disambiguating authors in academic publications using random forests](#). In *Proceedings of the 9th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '09*, pages 39–48, New York, NY, USA. Association for Computing Machinery.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Jan Philip Wahle, Terry Ruas, Saif Mohammad, and Bela Gipp. 2022. [D3: A massive dataset of scholarly metadata for analyzing the state of computer science research](#). In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2642–2651, Marseille, France. European Language Resources Association.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2024. [Text embeddings by weakly-supervised contrastive pre-training](#). *Preprint*, arXiv:2212.03533.

Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning, ICML '20*. JMLR.org.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Griffin Thomas Adams, Jeremy Howard, and Iacopo Poli. 2025. [Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics, Volume 1: Long Papers*, pages 2526–2547, Vienna, Austria. Association for Computational Linguistics.

Anna Wegmann and Dong Nguyen. 2021. [Does it capture STEL? a modular, similarity-based linguistic style evaluation framework](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7109–7130, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Anna Wegmann, Marijn Schraagen, and Dong Nguyen. 2022. [Same author or just same topic? towards content-independent style representations](#). In *Proceedings of the 7th Workshop on Representation Learning for NLP*, pages 249–268, Dublin, Ireland. Association for Computational Linguistics.

Yuxuan Zong and Benjamin Piwowarski. 2025. [Towards lossless token pruning in late-interaction retrieval models](#). In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '25*,

pages 2407–2417, New York, NY, USA. Association for Computing Machinery.

## A InfoNCE gradient derivation

We derive Equation 2 from Equation 1. Writing  $Z = \exp(s_p/\tau) + \sum_{n' \in \mathcal{N}} \exp(s_{n'}/\tau)$  for the partition function and  $s_p = s(a, p)$ ,  $s_n = s(a, n)$  for the anchor-positive and anchor-negative similarities, the InfoNCE loss for a single anchor is

$$\mathcal{L} = -\log \frac{\exp(s_p/\tau)}{Z} = -\frac{s_p}{\tau} + \log Z.$$

Differentiating with respect to a specific negative similarity  $s_n$  (treating  $s_p$  and the other negatives  $\{s_{n'} : n' \neq n\}$  as fixed inputs),

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial s_n} &= \frac{1}{Z} \cdot \frac{1}{\tau} \cdot \exp(s_n/\tau) \\ &= \frac{1}{\tau} \cdot \frac{\exp(s_n/\tau)}{\exp(s_p/\tau) + \sum_{n' \in \mathcal{N}} \exp(s_{n'}/\tau)}. \end{aligned}$$

The right-hand side is exactly the softmax weight of  $n$  in the contrastive denominator, scaled by  $1/\tau$ . Negatives with high similarity to the anchor receive exponentially more gradient than negatives with low similarity. In the limit of one dominant negative, the gradient collapses onto that negative alone, reproducing the behavior of hard-negative mining without an explicit mining procedure.

The comparison with the triplet loss follows from inspecting its form directly. With a fixed margin  $m$ , the triplet loss is  $\mathcal{L}_{\text{trip}}(s_p, s_n) = [m - s_p + s_n]_+$ , whose gradient with respect to  $s_n$  is  $+1$  when  $s_p - s_n < m$  and  $0$  otherwise. The gradient does not depend on how close  $s_n$  is to  $s_p$  within the active region, and it vanishes entirely once the margin is satisfied. Triplet loss therefore does not concentrate gradient on hard negatives in the InfoNCE sense: the loss treats all active negatives identically and ignores all inactive ones.

## B HALvest language and domain statistics

We selected eight features to represent HAL submissions:

- `halid`: submission’s unique identifier assigned by HAL.
- `lang`: the language of the document, as filled by the depositor.
- `title`: title of the document.

| Language                    | # Docs       | # Tokens      |
|-----------------------------|--------------|---------------|
| English                     | 464,679      | 8,158,933,235 |
| French                      | 199,216      | 9,018,529,985 |
| Spanish                     | 2,975        | 69,221,667    |
| Italian                     | 1,172        | 48,747,986    |
| Portuguese                  | 934          | 32,918,832    |
| German                      | 652          | 12,225,960    |
| Russian                     | 245          | 5,763,532     |
| Chinese                     | 160          | 2,861,585     |
| ...                         | ...          | ...           |
| <b>Total (56 languages)</b> | <b>778k+</b> | <b>17.4B</b>  |

Table 5: **Language distribution of HALvest.** Full counts (56 languages, including Basque, Catalan, Persian, and other low-resource entries) are included with the dataset release.

| Domain                         | Code | # Documents | # Tokens      |
|--------------------------------|------|-------------|---------------|
| Humanities and Social Sciences | shs  | 156,566     | 5,614,423,171 |
| Computer Science               | info | 148,316     | 2,573,673,455 |
| Life Sciences                  | sdv  | 115,744     | 3,145,323,780 |
| Engineering Sciences           | spi  | 102,751     | 2,254,653,825 |
| Physics                        | phys | 65,991      | 1,503,190,749 |
| Mathematics                    | math | 62,921      | 1,638,500,361 |
| Chemical Science               | chim | 40,012      | 899,507,319   |
| Environmental Science          | sde  | 31,575      | 579,076,669   |
| Sciences of the Universe       | sdu  | 23,557      | 682,356,264   |
| Cognitive Science              | scco | 11,772      | 227,487,096   |
| Statistics                     | stat | 10,579      | 184,678,350   |
| Quantitative Finance           | qfin | 3,451       | 68,518,636    |
| Nonlinear Sciences             | nlin | 1,972       | 30,694,088    |

Table 6: **Disciplinary domains retained for HALvest-Contrastive.** Hard-negative mining requires field-matched documents.

- domain: list of fields of study<sup>4</sup>.
- timestamp: time of access.
- year: publication year of the document if relevant. Otherwise, it is set to year 1.
- url: URL to access the PDF.
- authors: list of authors.

Table 5 reports per-language document and token counts for HALvest. Table 6 reports the 13 domains retained for HALvest-Contrastive triplet mining.

An author comprises three features:

- name: string for the author name, as filled by the depositor.
- affiliations: list of unique identifiers attributed by HAL to the institutions where the author belongs.
- halauthorid: unique identifier assigned by HAL to each author registered on the online repository. If an author is not registered on HAL, they are considered unidentified and are assigned a halauthorid of “0”.

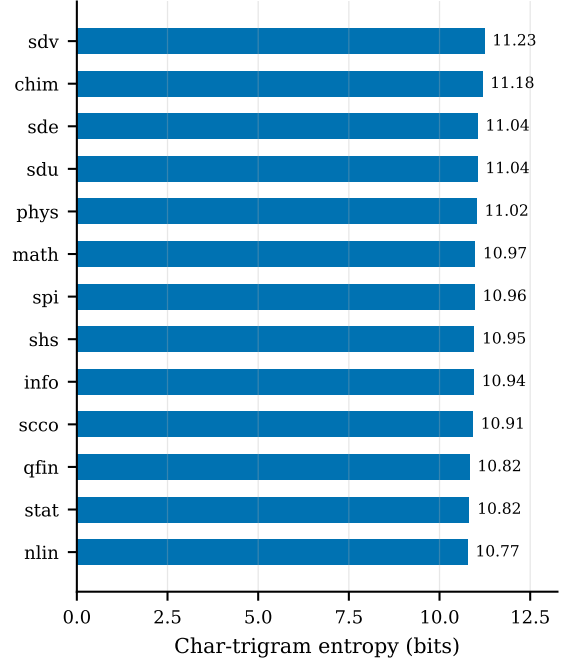


Figure 5: **Token-level entropy across HALvest domains.** Char-trigram entropy (in bits) for the 13 disciplinary domains retained in HALvest-Contrastive. Values range from 10.77 (Nonlinear Sciences) to 11.23 (Life Sciences), a narrow 0.46-bit spread. The near-uniform entropy confirms that academic writing remains low-entropy throughout the fields, supporting the claim that the stylistic residual is easier to isolate from topical noise in this register.

## C Full hyperparameters and setup

Unless stated otherwise, all trained models share a ModernBERT-base backbone (Warner et al., 2025) and are trained for a single epoch on the base configuration of HALvest-Contrastive. Experiments use four H100 GPUs with a batch size of 32 per device and gradient accumulation of 2, so that the effective batch is 64 per device. Each triplet contributes its positive and its mined negative to the all-gather, so each anchor is scored against a pool of  $2 \times 4 \times 32 = 256$  candidates. Accumulation repeats the gather instead of widening it, so it raises the examples per optimizer step and leaves the pool size unchanged. The learning rate is  $3 \times 10^{-5}$ , the InfoNCE temperature  $\tau$  is 0.5, weight decay is 0.1, and we train in 16-bit mixed precision. Runs with four GB200 at a batch size of 64 per device (512 in-batch negatives) are reported in §5.5 as a scaling reference.

Table 7 reproduces the training configuration

<sup>4</sup><https://hal.science/browse/domain>

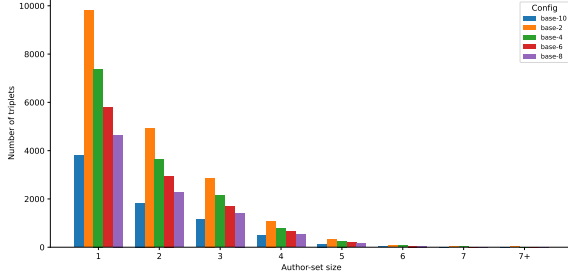


Figure 6: **Author-set size distribution in HALvest-Contrastive.** Each color represents a subset of  $k$  contiguous sentences (respectively  $k=10, 2, 4, 6, 8$ ).

| Hyperparameter                     | Value                                                             |
|------------------------------------|-------------------------------------------------------------------|
| Backbone                           | ModernBERT-base                                                   |
| Epochs                             | 1                                                                 |
| Batch size per device              | 32                                                                |
| Gradient accumulation              | 2                                                                 |
| GPUs                               | $4 \times \text{H100 (primary)}, 4 \times \text{GB200 (scaling)}$ |
| Gathered candidate pool per anchor | 256 (H100), 512 (GB200)                                           |
| Learning rate                      | $3 \times 10^{-5}$                                                |
| Weight decay                       | 0.1                                                               |
| Precision                          | 16-bit mixed                                                      |
| InfoNCE temperature $\tau$         | 0.5                                                               |

Table 7: **Shared training hyperparameters.** Every fine-tuned model uses these settings.

of every main-body model. All other settings are defaults of the PyTorch Lightning (Falcon and The PyTorch Lightning team, 2019) training pipeline that accompanies the release.

## D Per-split results

Tables 8 to 12 report retrieval metrics for every trained model on every test split (base-2, base-4, base-6, base-8, base-10), including the learned-PLI variants whose negative-result analysis is in Appendix G. We omit layerwise pooling variants, which were abandoned after validation-set evaluation showed only marginal improvement over mean pooling.

| Model             | R@20         | R@100        | nDCG@20      | nDCG@100     |
|-------------------|--------------|--------------|--------------|--------------|
| Mean pooling      | 0.060        | 0.173        | 0.032        | 0.058        |
| Mean pooling (LI) | 0.073        | 0.190        | 0.041        | 0.067        |
| LI                | 0.280        | 0.457        | <b>0.202</b> | <b>0.243</b> |
| PLI n-gram 2      | <b>0.284</b> | 0.465        | 0.200        | 0.242        |
| PLI n-gram 3      | 0.281        | <b>0.468</b> | 0.196        | 0.239        |
| PLI n-gram 4      | 0.265        | 0.451        | 0.181        | 0.224        |
| PLI n-gram 5      | 0.266        | 0.457        | 0.180        | 0.224        |
| PLI whole-word    | —            | —            | —            | —            |
| PLI learned       | 0.258        | 0.450        | 0.171        | 0.215        |

Table 8: **HALvest-Contrastive base-2** retrieval metrics.

## E Alignment and uniformity values

Table 13 reports alignment and uniformity (Wang and Isola, 2020) for every model. Mean pooling

| Model             | R@20         | R@100        | nDCG@20      | nDCG@100     |
|-------------------|--------------|--------------|--------------|--------------|
| Mean pooling      | 0.121        | 0.294        | 0.063        | 0.101        |
| Mean pooling (LI) | 0.147        | 0.328        | 0.082        | 0.121        |
| LI                | 0.485        | 0.678        | 0.364        | 0.408        |
| PLI n-gram 2      | <b>0.497</b> | <b>0.700</b> | <b>0.365</b> | <b>0.411</b> |
| PLI n-gram 3      | 0.495        | 0.699        | 0.361        | 0.407        |
| PLI n-gram 4      | 0.472        | 0.687        | 0.339        | 0.387        |
| PLI n-gram 5      | 0.478        | 0.693        | 0.341        | 0.390        |
| PLI whole-word    | 0.482        | 0.680        | 0.358        | 0.403        |
| PLI learned       | 0.452        | 0.670        | 0.314        | 0.363        |

Table 9: **HALvest-Contrastive base-4** retrieval metrics.

| Model             | R@20         | R@100        | nDCG@20      | nDCG@100     |
|-------------------|--------------|--------------|--------------|--------------|
| Mean pooling      | 0.163        | 0.377        | 0.087        | 0.132        |
| Mean pooling (LI) | 0.193        | 0.420        | 0.107        | 0.156        |
| LI                | 0.603        | 0.779        | 0.464        | 0.503        |
| PLI n-gram 2      | 0.616        | 0.797        | <b>0.466</b> | <b>0.507</b> |
| PLI n-gram 3      | <b>0.617</b> | <b>0.803</b> | 0.464        | 0.506        |
| PLI n-gram 4      | 0.593        | 0.788        | 0.442        | 0.485        |
| PLI n-gram 5      | 0.604        | 0.797        | 0.443        | 0.486        |
| PLI whole-word    | 0.601        | 0.776        | 0.457        | 0.496        |
| PLI learned       | —            | —            | —            | —            |

Table 10: **HALvest-Contrastive base-6** retrieval metrics.

achieves the lowest uniformity of any model ( $u = -2.0$ ) and a higher alignment loss than either late-interaction variant ( $\alpha = 0.154$  against 0.091 and 0.098): random pairs are well spread out, yet same-author pairs are not close. It is still better than PLI  $n=2$  on both axes, which is the cross-family failure discussed in §5.3. LI and whole-word PLI sit in the upper-left region. As patch size  $n$  grows, the fixed- $n$  PLI variants trade alignment for uniformity along a monotonic axis.

## F Retrieval failure modes

We inspect the failure modes of our strongest PLI baseline (PLI n-gram 2 on  $k = 4$ ) by ranking 500 sampled queries against a 1000-document validation pool (5 seeds  $\times$  100 queries). The model places the true positive at rank 1 in 20.8% of cases and within the top 5 in 39.8%, with a long tail (16.6% of true positives at ranks 101–500 and 8.0% beyond rank 500).

We call “distractors” highly ranked negative documents, and categorize them. Some distractors share the query’s domain at 61% across rank buckets, against 20% for random pairs. The model has learned domain as a strong similarity cue, consistent with HALvest-Contrastive’s field-matched negative sampling. The top-K distractors share at least one author with the query in 11.4% of cases at ranks 2–5 and 11.8% at ranks 6–20, but 0.0% across 500 random pairs. The author-overlap drops by a factor of 3.5 between the top-20 distractors

| Model             | R@20         | R@100        | nDCG@20      | nDCG@100     |
|-------------------|--------------|--------------|--------------|--------------|
| Mean pooling      | 0.211        | 0.450        | 0.114        | 0.164        |
| Mean pooling (LI) | 0.247        | 0.494        | 0.143        | 0.195        |
| LI                | 0.690        | 0.847        | 0.537        | 0.571        |
| PLI n-gram 2      | 0.706        | 0.862        | <b>0.545</b> | <b>0.580</b> |
| PLI n-gram 3      | <b>0.713</b> | <b>0.865</b> | <b>0.545</b> | 0.579        |
| PLI n-gram 4      | 0.688        | 0.855        | 0.521        | 0.558        |
| PLI n-gram 5      | 0.699        | 0.863        | 0.523        | 0.560        |
| PLI whole-word    | 0.688        | 0.849        | 0.534        | 0.570        |
| PLI learned       | 0.659        | 0.838        | 0.483        | 0.522        |

Table 11: **HALvest-Contrastive base-8** retrieval metrics.

| Model             | R@20         | R@100        | nDCG@20      | nDCG@100     |
|-------------------|--------------|--------------|--------------|--------------|
| Mean pooling      | 0.246        | 0.499        | 0.130        | 0.182        |
| Mean pooling (LI) | 0.287        | 0.547        | 0.163        | 0.217        |
| LI                | 0.740        | 0.880        | 0.586        | 0.616        |
| PLI n-gram 2      | 0.760        | 0.897        | <b>0.595</b> | <b>0.625</b> |
| PLI n-gram 3      | <b>0.765</b> | <b>0.901</b> | 0.594        | 0.624        |
| PLI n-gram 4      | 0.742        | 0.892        | 0.571        | 0.604        |
| PLI n-gram 5      | 0.754        | 0.899        | 0.576        | 0.608        |
| PLI whole-word    | 0.742        | 0.887        | 0.580        | 0.612        |
| PLI learned       | 0.714        | 0.872        | 0.534        | 0.572        |

Table 12: **HALvest-Contrastive base-10** retrieval metrics.

and the subsequent ones ranked lower. The Jaccard similarity with the query (unigram) is essentially flat across rank buckets (0.086, 0.087, 0.085, with random pairs at 0.073). The Jaccard similarity with the true positive is more informative: 0.129 at rank 1, 0.131 at ranks 2–5, 0.126 at ranks 6–20, dropping to 0.076 at lower ranks. High-ranked distractors do not look textually similar to the query, but they do look textually similar to the true positive. This is structurally different from a topic-retrieval failure mode, where lexical query overlap would dominate.

## G Learned PLI: failure-mode analysis

This appendix documents our attempt at learning differentiable patch boundaries. The architecture is simple. The difficulty lies in the regularizer. We tried two natural single-term forms. The first, not lower-bounded, collapsed the policy onto token-level scoring. The second was bounded but failed to interact with the inference threshold and collapsed the policy onto a single patch per sequence. A successful regularizer would need to combine two terms acting in concert, following the byte-latent-transformer (BLT) pattern (Pagnoni et al., 2025): one term penalizing excessively long patches and another penalizing excessively short ones, with a scheduled balance between the two. The hyperparameter sweep this requires was beyond our compute budget for this article.

| Model                      | $\alpha$ | $u$    |
|----------------------------|----------|--------|
| Pretrained, no fine-tuning | 0.069    | −0.171 |
| Mean pooling               | 0.154    | −2.000 |
| LI (full token)            | 0.091    | −0.331 |
| PLI whole-word             | 0.098    | −0.378 |
| PLI n-gram 2               | 0.201    | −0.717 |
| PLI n-gram 3               | 0.320    | −1.150 |
| PLI n-gram 4               | 0.438    | −1.565 |
| PLI n-gram 5               | 0.519    | −1.909 |

Table 13: **Alignment and uniformity per model.**  $\alpha$  is alignment loss (lower = positives closer in embedding space);  $u$  is uniformity (lower = random pairs more uniformly distributed).

## G.1 Architecture and notation

Let  $\mathbf{h}_t \in \mathbb{R}^d$  be the encoder’s hidden state at token position  $t$ , and let  $S$  denote the number of valid (non-padding) query positions in a sequence. A two-layer feed-forward network (a patch-boundary predictor) on top of the ModernBERT hidden states produces a boundary logit per position,

$$\ell_t = W_2 \phi(W_1 \mathbf{h}_t + \mathbf{b}_1) + b_2, \quad (4)$$

with  $W_1 \in \mathbb{R}^{(d/4) \times d}$ ,  $W_2 \in \mathbb{R}^{1 \times (d/4)}$ , and  $\phi$  a GELU. The cut probability is  $q_t = \sigma(\ell_t)$ . At training time, cut decisions are sampled with a Gumbel-Softmax alongside a straight-through gradient estimation. We anneal the temperature  $\tau_{\text{GS}}$  from 1.0 to 0.1 over the first 20,000 optimizer steps. The first valid position is forced to be a cut; this guarantees at least one patch regardless of the sampler. At inference time, the cut decision is hard-thresholded at 0.5.

The total training loss is  $\mathcal{L} = \mathcal{L}_{\text{InfoNCE}} + \lambda \cdot \mathcal{L}_{\text{patch}}$  with  $\lambda = 0.1$ , where  $\mathcal{L}_{\text{patch}}$  is the patch regularizer whose role is to prevent degenerate solutions.

## G.2 Unbounded log-sum penalty

The first form was

$$\mathcal{L}_{\text{patch}}^{(1)} = -\log\left(\sum_{t=1}^S q_t\right). \quad (5)$$

The intuition: as  $\sum_t q_t \rightarrow 0$ , the regularizer diverges to  $+\infty$  and rules out the single-patch degeneracy. Beyond that, the regularizer was supposed to be roughly flat at high cut counts, letting InfoNCE push the cut count freely.

For  $\sum_t q_t > 1$ , however, the function  $-\log(\sum_t q_t)$  is monotonically decreasing in  $\sum_t q_t$

and is not lower-bounded. The regularizer therefore supplies a uniform downward push on every cut probability, regardless of position.

What (5) needed was a regularizer that diverges to  $+\infty$  at *both* extremes of  $\sum_t q_t$ . Restoring the missing one requires committing to a target rate, which is what the second attempt did.

### G.3 Target-rate squared error

The second form was a target-rate-matching squared-error penalty,

$$\mathcal{L}_{\text{patch}}^{(2)} = \left( \frac{1}{S} \sum_{t=1}^S q_t - p_{\text{target}} \right)^2, \quad (6)$$

with  $p_{\text{target}} = 1/3$ . This target rate corresponds to a mean patch length of  $1/p_{\text{target}} = 3$  tokens, which lands in the middle of the empirical  $n$ -gram optima we identified (best  $n$  is 2 at base-2 and base-4, 3 at base-6 onward). The hyperparameter is therefore not tuned but set from independent empirical evidence. Equation (6) has a minimum at  $\bar{q} := \frac{1}{S} \sum_t q_t = p_{\text{target}}$ , is lower-bounded at zero, symmetric, and differentiable everywhere.

**The retrained model satisfies the regularizer exactly** and produces a single patch per sequence. After retraining under  $\mathcal{L}_{\text{patch}}^{(2)}$ , the diagnostic check on the resulting checkpoint reveals  $\bar{q} \approx 1/3$  for every sequence:  $\mathcal{L}_{\text{patch}}^{(2)}$  is doing exactly what was asked of it. The inference threshold, however, is 0.5. If every position has  $q_t \approx 1/3$ , then no position satisfies  $q_t > 0.5$ , and the inference rule produces zero cuts.

The MSE regularizer only constrains the population mean of cut probabilities. It places no constraint on the distribution shape. InfoNCE provides no pressure to break the uniform policy. The regularizer does not align its constraint with the inference threshold. With  $p_{\text{target}} = 1/3$  and threshold 0.5, the model can satisfy the regularizer exactly while producing zero cuts at inference. Setting  $p_{\text{target}} = 0.5$  does not fix this, as  $q_t = 0.5$  everywhere produces random cuts under Gumbel sampling.

### G.4 What a successful regularizer would look like

The BLT (Pagnoni et al., 2025) approach provides a template. A multi-term regularizer would combine (a) a *minimum-rate* term that penalizes sequences whose mean cut probability falls below a lower

threshold, ensuring enough cuts are placed to avoid the single-patch degeneracy; and (b) a *maximum-rate* term that penalizes sequences whose mean cut probability exceeds an upper threshold, preventing collapse to token-level scoring. A per-position entropy bonus could additionally encourage the network to produce high-confidence cuts (probabilities near 0 or 1) rather than uniform values near the threshold. The challenge is that the three coefficients interact: the minimum-rate and maximum-rate bounds must be calibrated jointly with the inference threshold, and the entropy bonus must not dominate early training when the boundary predictor has not yet converged. This hyperparameter sweep, combined with the Gumbel-Softmax temperature schedule, amounts to a five-dimensional search that we leave to future work.

## H E5 zero-shot full metrics

| Metric                            | Dense | LI    |
|-----------------------------------|-------|-------|
| <i>HALvest-Contrastive base-4</i> |       |       |
| R@20                              | 0.167 | 0.203 |
| R@100                             | 0.269 | 0.307 |
| nDCG@20                           | 0.124 | 0.152 |
| nDCG@100                          | 0.146 | 0.175 |
| Triplet accuracy                  | 0.789 |       |
| <i>PAN19 zero-shot</i>            |       |       |
| R@20                              | 0.097 | 0.179 |
| nDCG@20                           | 0.029 | 0.064 |
| Triplet accuracy                  | 0.628 |       |

Table 14: **E5 zero-shot metrics.** E5 (Wang et al., 2024) on HALvest-Contrastive base-4 and PAN19.

We include the late-interaction scorer (token-level MaxSim) as an ablation to verify that the conclusions of §5.4 are not an artifact of the chosen scorer. The late-interaction scorer improves retrieval metrics uniformly over the dense scorer. Still, the overall trend remains the same: E5 retains a high triplet accuracy (0.789 on HALvest-C, 0.628 on PAN19) while struggling on retrieval. The gap between the two metric families is the motivating observation of §5.4.

## I Sequence length

Increasing sequence length from 2 to 8 sentences amplifies the lexical signal for both restricted and unrestricted triplet types. The effect is far more pronounced for unrestricted triplets. The signal for restricted triplets also increases but remains substantially weaker and begins to plateau after 6 sentences. More text provides more stylistic

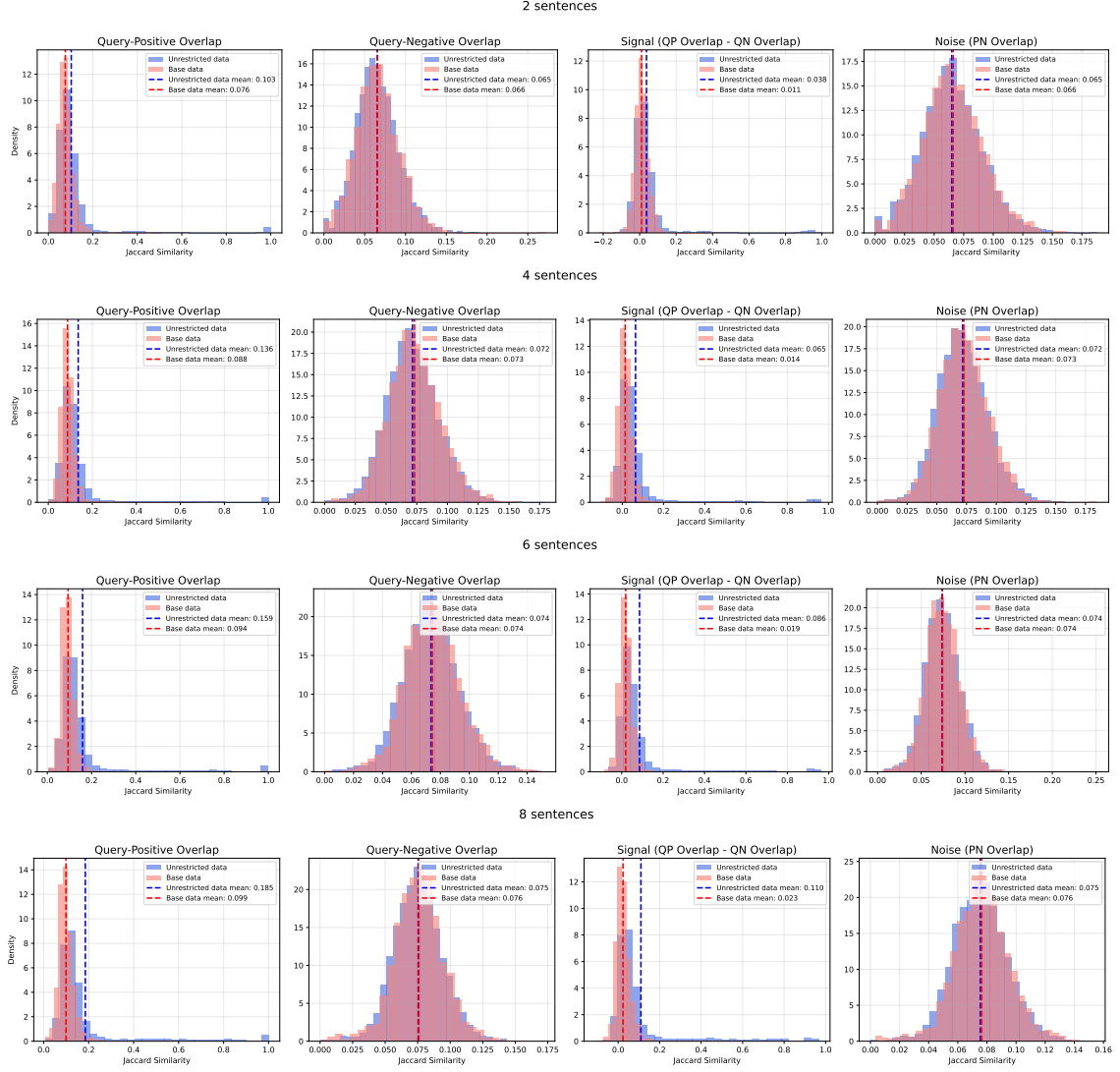


Figure 7: **Lexical signal and noise across test splits.** We sample 10,000 examples, split by word, and compute Jaccard similarities. **Signal** (third column) is defined as the difference between query-positive and query-negative overlap. **Noise** (fourth column) is the Jaccard similarity between the positive and the negative of a triplet. Rows correspond to span lengths  $k \in \{2, 4, 6, 8\}$ . Blue: unrestricted configuration. Red: base configuration.

evidence. The amount of unique author-specific evidence in topic-decoupled scenarios appears to saturate quickly, as shown in Figure 8.

We also analyzed the inherent lexical signal within contrastive triplets using Jaccard similarity before scaling HALvest-Contrastive. Recall that we tested three different configurations before settling on our current version of HALvest-Contrastive. Queries denote what we call anchors in the main paper. As shown in Figure 7, the lexical signal is substantially higher in the unrestricted triplets, where the query and positive can come from the same paper. For the 8-sentence split, the mean signal is 0.110 for unrestricted triplets but only 0.023 for the base configuration. This con-

firms that base HALvest-Contrastive successfully minimizes keyword overlap, forcing any successful model to rely primarily on signals beyond shared tokens.

## J Computational cost of late interaction and PLI

We derive the scoring cost of MaxSim for late interaction and for PLI with patch size  $n$ , then instantiate the expressions under our training configuration.

### J.1 MaxSim cost per document pair

Let  $\mathbf{Q} \in \mathbb{R}^{S_q \times d}$  and  $\mathbf{D} \in \mathbb{R}^{S_d \times d}$  be the token-embedding matrices of a query (anchor) and a doc-

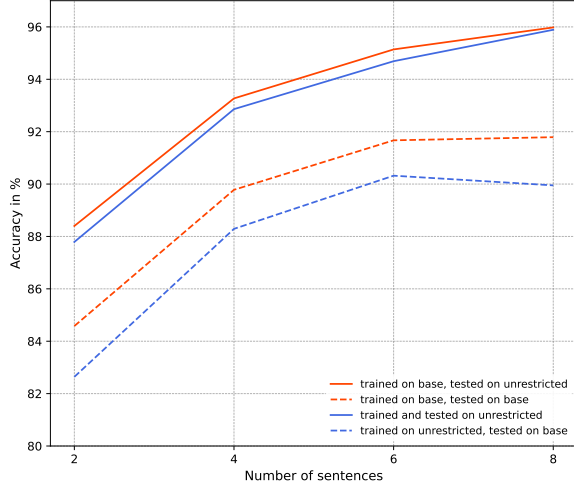


Figure 8: **Accuracy vs. span length for two fine-tuned RoBERTa models.** Blue: trained on unrestricted data. Red: trained on base data. Solid lines track performance on unrestricted test. Dashed lines track base test.

ument (candidate), where  $S_q, S_d$  are the respective valid token counts and  $d$  is the embedding dimension. Assuming  $\ell_2$ -normalised rows, MaxSim is computed as

$$\text{MaxSim}(\mathbf{Q}, \mathbf{D}) = \sum_{i=1}^{S_q} \max_{j \in [S_d]} \mathbf{q}_i^\top \mathbf{d}_j. \quad (7)$$

The computation decomposes into three stages:

1. **Pairwise dot products.** The full similarity matrix  $\mathbf{QD}^\top \in \mathbb{R}^{S_q \times S_d}$  requires  $S_q \cdot S_d$  inner products of dimension  $d$ , each costing  $2d - 1$  FLOPs ( $d$  multiplications and  $d - 1$  additions). The total is  $S_q S_d (2d - 1)$ .
2. **Row-wise maxima.** Extracting  $\max_j$  for each of the  $S_q$  rows requires  $S_q (S_d - 1)$  comparisons.
3. **Summation.** Summing the  $S_q$  maxima costs  $S_q - 1$  additions.

Since  $d \gg 1$ , the pairwise dot-product term dominates, and we write the cost as

$$C_{\text{MaxSim}}(S_q, S_d, d) \approx 2 S_q S_d d. \quad (8)$$

For the symmetric case  $S_q = S_d = S$  common in our setup, this simplifies to  $2S^2 d$ .

## J.2 PLI: patching overhead and reduced scoring

With a fixed patch size  $n$ , each sequence of  $S$  tokens is compressed into  $\mathcal{P} = \lceil S/n \rceil$  patch vectors by mean-pooling contiguous groups of  $n$  token embeddings.

**Patching cost.** For each of  $\mathcal{P}$  patches, averaging  $n$  vectors of dimension  $d$  requires  $(n-1) \cdot d$  additions and  $d$  scalar divisions, totalling approximately  $n \cdot d$  FLOPs per patch and  $\mathcal{P} \cdot n \cdot d = S \cdot d$  FLOPs per sequence. For two sequences, the patching cost is  $2Sd$ .

**Reduced MaxSim cost.** After patching, the similarity matrix is  $\mathbf{Q}_{\text{patch}} \mathbf{D}_{\text{patch}}^\top \in \mathbb{R}^{P_q \times P_d}$ , and the scoring cost becomes

$$C_{\text{MaxSim}}^{\text{PLI}}(S, d, n) \approx 2 \left\lceil \frac{S}{n} \right\rceil^2 d \approx \frac{2S^2 d}{n^2}. \quad (9)$$

The total PLI cost per pair is therefore  $C_{\text{MaxSim}}^{\text{PLI}} + 2Sd$ . Since  $2Sd \ll 2S^2 d/n^2$  for all practical  $S$  and  $n \leq S$ , the patching overhead is negligible, and the scoring cost reduction factor is  $n^2$ .

## J.3 Scaling to a full training step

Under InfoNCE with full-gather, each anchor scores against every passage in the gathered pool. With  $A = 128$  anchors and a pool of  $C = 256$  candidates, one of which is the anchor’s own positive, a micro-batch costs  $A \cdot C = 32,768$  MaxSim evaluations. Since each candidate’s embedding matrix is computed once by the encoder and reused across all anchors, the  $n^2$  speedup per pair carries through unchanged. Every one of the  $A \cdot C$  pairs gets cheaper by a factor of  $n^2$ , so the aggregate scoring cost drops by the same factor. Table 15

| Model      | Patches/seq | FLOPs/pair         | Speedup          |
|------------|-------------|--------------------|------------------|
| LI (token) | 512         | $4.03 \times 10^8$ | $1 \times$       |
| PLI $n=2$  | 256         | $1.01 \times 10^8$ | $4 \times$       |
| PLI $n=3$  | 171         | $4.49 \times 10^7$ | $\sim 9 \times$  |
| PLI $n=4$  | 128         | $2.52 \times 10^7$ | $16 \times$      |
| PLI $n=5$  | 103         | $1.63 \times 10^7$ | $\sim 25 \times$ |

Table 15: **MaxSim scoring cost per document pair** at  $S = 512, d = 768$ . The speedup column reports  $C_{\text{MaxSim}}^{\text{LI}}/C_{\text{MaxSim}}^{\text{PLI}}$ . The “ $\sim$ ” prefix reflects the ceiling effect in  $\lceil S/n \rceil$ .

reports the per-pair cost for our primary configuration ( $d = 768$ ) at the maximum sequence length  $S = 512$ .

**Storage.** Each document requires  $S \cdot d$  float values under LI and  $\lceil S/n \rceil \cdot d$  under PLI, yielding a storage reduction factor of  $n$  (because storage scales with the number of vectors, not with pairwise interactions). At  $S = 512$  and  $d = 768$  in float16, a single document occupies 768 KB under LI, 384 KB under PLI  $n=2$ , and 154 KB under PLI  $n=5$ .

**Mixed sequence lengths.** Training uses a mix of span lengths (Table 4), with mean token counts ranging from  $\bar{S} \approx 70$  (base-2) to  $\bar{S} \approx 328$  (base-10). Since MaxSim cost is quadratic in  $S$ , longer spans dominate the compute. The  $n^2$  reduction factor applies uniformly regardless of  $S$ , so the relative advantage of PLI over LI is independent of the span-length distribution.