Person-Place Relation Extraction in Historical Texts: A Dual Approach with Fine-Tuned NLI Encoders and SLMs at HIPE-2026

Anderson Morillo[1,*], Edwin Puertas[1] and Juan Carlos Martinez-Santos[1]

[1]Universidad Tecnologica de Bolivar, Cartagena, Colombia

Abstract

Under the team name **VerbaNexAI II**, we participate in HIPE-2026, which focuses on the classification of temporally-scoped relations between person and location mentions in multilingual historical documents. We develop and evaluate two distinct approaches: (1) a multilingual Natural Language Inference (NLI) encoder adapted to relation classification on translated text, and (2) a compact, self-distilled language model optimized through programmatic instruction search and fine-tuned on *gold-filtered teacher predictions* that is, structured model outputs whose predicted relation label matches the gold annotation on the training set. We evaluate both systems on **Test A (newspaper test set)** and **Test B (French literary surprise test set)** using the official gold labels released for the evaluation files. Our self-distilled language model consistently outperforms the natural language inference encoder, achieving global macro recalls of 0.581, 0.588, and 0.570 on the English, French, and German Test A splits, respectively, and 0.534 on the Test B surprise split. Furthermore, the language model generates explanatory rationales for each prediction, offering practical utility for archival curation workflows.

Keywords

historical NLP, relation extraction, natural language inference, large language models, prompt optimization, supervised fine-tuning

1. Introduction

Extracting person-place relations from historical documents is a long-standing curation need in the digital humanities: who appears where, when, and on what document evidence, is the structural skeleton that underpins biographies, prosopographies and the geo-temporal browsing of archival collections. Doing it at scale is gated by OCR noise, language variability across English, French and German, and the quadratic growth of candidate person-place pairs with document length. HIPE-2026 [1, 2, 3][1], the latest CLEF lab in the HIPE series [4], recasts that need as a per-pair classification problem. Each candidate pair is assigned a three-way `at` relation on whether the person was at the location at any time before publication, and a binary `isAt` relation on the short-range temporal horizon of publication. Systems are ranked by macro recall averaged over the two relations and their classes a metric robust to the strong FALSE skew of historical text.

We contribute two complementary systems. The first is a fine-tuned Natural Language Inference (NLI) encoder,[2] adapted to relation classification by framing the candidate pairs as NLI premises and hypotheses to predict entailment, neutral, or contradiction classes on an OCR-repaired and translated English view of the corpus. This system constitutes our submitted (`VerbaNexAI II run1`). The second, which is our primary submission (`VerbaNexAI II run3`), is a compact open-weights language model[3] trained through teacher filtered distillation: a chain-of-thought program is evolved under a reflective prompt optimizer [7], then used to generate teacher completions on **Gold Train A**; only completions

CLEF 2026: Conference and Labs of the Evaluation Forum, September 2026, Jena, Germany

*Corresponding author.

amorillo@utb.edu.co (A. Morillo); epuerta@utb.edu.co (E. Puertas); jcmartinezs@utb.edu.co (J. C. Martinez-Santos)

0009-0005-7944-3342 (A. Morillo); 0000-0002-0758-1851 (E. Puertas); 0000-0003-2755-0718 (J. C. Martinez-Santos)

[1]https://hipe-eval.github.io/HIPE-2026/

[2]`mDeBERTa-v3-base-xnli-multilingual-nli-2mil7` [5]

[3]Nemotron-3-Nano-4B [6].

whose predicted label agrees with the gold annotation are kept, and the student is fine-tuned with a low-rank adapter [8] on this filtered corpus.

2. Background

Extracting entity relations from historical, unstructured corpora requires a formal definition of the target tasks and a solid alignment with established machine learning paradigms. This section provides the methodological foundation of our work. First, we outline the HIPE-2026 evaluation framework and the metrics used to assess system quality. Second, we review the three key modeling idioms natural language inference recasting, programmatic prompt optimization, and parameter-efficient fine-tuning and distillation that inform our system design and guide our dual-track architectures.

2.1. Task and Evaluation

HIPE-2026 [1, 2] defines the evaluation setting: candidate person-place pairs in historical multilingual documents are classified under macro recall averaged over the `at` (three-way classification) and `isAt` (binary classification) relations. This evaluation profile heavily penalizes the `FALSE`-dominated trivial classifier and prevents a model from collapsing its predictions onto a single class, emphasizing performance on rare positive classes in highly skewed historical texts.

2.2. Methodological Idioms

Three modeling idioms underpin our dual pipeline. We utilize these paradigms to navigate the trade-offs between computational efficiency, cross-lingual generalization, and structured output formatting:

1. **Natural Language Inference Recasting**: Multilingual transformer encoders remain highly competitive on cross-lingual classification when the task is recast as Natural Language Inference (NLI) [5]. This idiom exposes the target class set through a single premise-hypothesis schema, leveraging pre-trained entailment capabilities to generalize across languages without heavy architectural adaptation.
2. **Prompt Optimization with Declarative Programs**: Instead of manual prompt engineering, LLMs can be addressed as declarative programs whose instructions and prompt schemas are automatically evolved against a metric. Under the DSPy framework (*Declarative Self-improving Python*) [9], reflective optimizers such as GEPA (*Genetic-Pareto*) [7] use automated feedback loops to refine prompt structures iteratively.
3. **Parameter-Efficient Fine-Tuning and Distillation**: The formatting and reasoning patterns of optimized prompt programs can be distilled directly into a smaller student's weights. We employ Low-Rank Adaptation (LoRA) [8] as our parameter-efficient fine-tuning (PEFT) paradigm to fine-tune compact language models on teacher-generated completions, retaining only those whose predicted labels match the gold annotation.

The combination and empirical comparison of these three idioms on OCR-noisy multilingual historical documents is what the remainder of this paper develops.

3. System Overview

Our proposed dual-pipeline architecture is designed to handle the dual challenges of language variation and historical OCR noise under distinct resource constraints. In this section, we describe our system's design and structural workflows, starting with the composition of our datasets, our document preprocessing and translation strategies, the architecture and fine-tuning details of both relation classifiers, and finally our inference and parsing configurations. An overview of the integrated pipeline is shown in Figure 1.

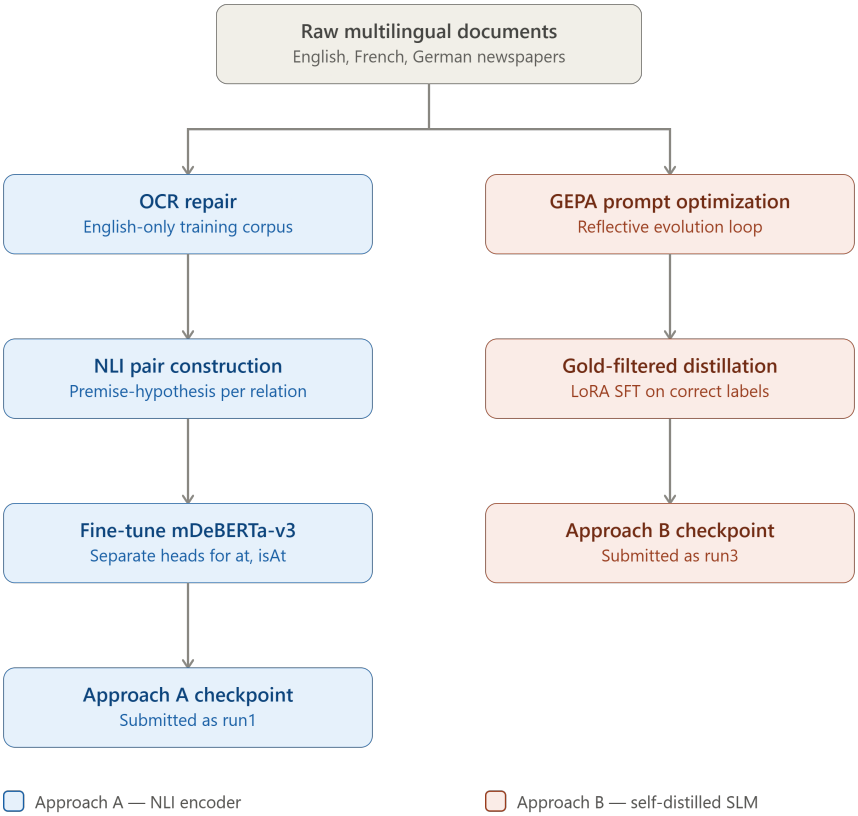

Figure 1: Training and deployment architecture of our dual relation extraction pipeline under the **VerbaNexAI II** framework. Raw multilingual documents are split into two parallel processing paths: (Left) **Approach A (NLI encoder)**, which repairs and translates documents into an English view using our distilled local LFM2.5 model, builds premise-hypothesis text pairs for NLI, and fine-tunes an mDeBERTa-v3 classifier head (submitted as `run1`); and (Right) **Approach B (Self-distilled SLM)**, which starts from an initial DSPy chain-of-thought signature, evolves the prompt program with Genetic-Pareto (GEPA), runs the optimized program over **Gold Train A** to obtain teacher completions, keeps only completions whose predicted label matches gold, and fine-tunes a local Nemotron-3-Nano-4B student under LoRA on this filtered corpus (submitted as `run3`).

3.1. Data

We use the manually annotated training splits of the three HIPE-2026 v1.0 impresso newspaper datasets, referred to officially as **Gold Train A** (newspaper training set), across English, French, and German (Table 1).[4] Both relation targets are heavily FALSE-skewed, and the positive PROBABLE class is highly rare (comprising less than 10% of all labels). The final evaluation covers 87 documents and 1,118 candidate pairs across the impresso splits, designated as **Test A (newspaper test set)**, alongside a French literary work surprise set, designated as **Test B (French literary surprise test set)** which evaluates only the three-class `at` relation. Each Test A file is provided in two variants: raw OCR text in its original source language, and a translated and corrected English view generated by our distilled local LFM2.5 rewriter.

The shared task provides two data regimes. **Gold Train A** (Table 1) is the manually annotated newspaper training set used for model development and supervised fine-tuning. The evaluation files for **Test A** and **Test B** were released initially without public gold labels; the organizers later published the official test annotations, whose label distribution is summarized in Table 2. Note that while Test B is ranked on `at` only in the official CLEF campaign, the released gold labels assign `isAt=FALSE` to every Test B pair, which makes the binary target artificially easy and should not be interpreted as part of the official Test B ranking profile.

[4]The HIPE-2026 data is released publicly at https://github.com/hipe-eval/HIPE-2026

Table 1
Composition of the HIPE-2026 v1.0 **Gold Train A (newspaper training set)** used in this work.

	Docs	Pairs	at (%)			isAt (%)	
Lang.			TRUE	PROB.	FALSE	TRUE	FALSE
en	35	307	40.7	9.8	49.5	27.0	73.0
fr	35	478	37.9	5.9	56.3	22.4	77.6
de	34	466	29.0	13.3	57.7	19.1	80.9
All	104	1 251	35.3	9.6	55.2	22.3	77.7

Table 2
Official gold label distribution for **Test A (newspaper test set)** and **Test B (French literary surprise test set)** evaluation files.

	Docs	Pairs	at (%)			isAt (%)	
Split			TRUE	PROB.	FALSE	TRUE	FALSE
impresso-test-en	19	162	46.3	17.3	36.4	40.1	59.9
impresso-test-fr	19	238	35.7	8.0	56.3	34.0	66.0
impresso-test-de	19	238	22.7	16.0	61.3	28.2	71.8
surprise-test-fr (B)	30	480	27.1	12.5	60.4	0.0	100.0
All	87	1 118	30.8	13.0	56.3	19.1	80.9

On the French surprise split (Test B), the official gold labels assign `isAt=FALSE` to every one of the 480 annotated pairs (0% TRUE, 100% FALSE).

3.2. Preprocessing

To navigate the linguistic variety and high levels of noise in historical OCR datasets, our dual pipeline uses two distinct representations. Approach A is trained and evaluated on a unified English view of the corpus: each document is first preprocessed to repair OCR transcriptions and translated from French or German into English before NLI examples are built. Approach B, by contrast, operates natively on the raw source language files, requiring the prompt-optimized language model to reason directly on noisy, multilingual tokens.

Motivated by the severe structural degradations common in historical newspapers (such as broken words, merged terms, and character-level substitutions), we employ OCR repair as a critical cleaning step to restore clean lexical boundaries. Furthermore, translation into a single high-resource language unifies the representation space, enabling smaller models to leverage extensive English pre-training.

To provide a computationally lightweight and cost-effective local implementation for translation and OCR correction, we distilled these capabilities into a self-hosted LFM2.5 rewriter.[5] This model leverages LoRA adapters [8] (r=α=16) applied to the attention and MLP projection layers, with a maximum sequence length of 2048, per-device batch size 2, and gradient accumulation of 4 (yielding an effective batch size of 8). Training is performed over 60 optimization steps with a learning rate of $2{\cdot}10^{-4}$ using 8-bit AdamW, a linear schedule, and loss masking applied exclusively to the assistant responses. At evaluation time, this local rewriter substitutes for commercial API engines when translating and repairing OCR noise in the test files.

3.3. Approach A: Fine-Tuned NLI Encoder

Approach A treats each candidate person-location pair as a natural language inference example. This formulation allows us to utilize a highly compact classifier backbone while leveraging existing linguistic

[5] An instruction-tuned 1.2B parameter model: `LFM2.5-1.2B-Instruct`.

entailment reasoning. The backbone is mDeBERTa-v3-base,[6] an ≈0.3B-parameter model pre-trained on the CC100 corpus [10] and fine-tuned on 2.7 million NLI examples in 26 languages.

For each pair, we construct a premise and a hypothesis. The premise contains the preprocessed English document context, any available temporal anchors, and a brief prompt directive to focus on the candidate entities. To preserve multi-mention entity references, various surface mentions of the target person or location are joined with commas. The hypothesis is conditioned on the relation: for `at`, the hypothesis states that the person was present at the location at some point before document publication, whereas for `isAt`, it asserts presence within a short-range temporal horizon of approximately one month before publication. The labels map directly onto the NLI categories: `TRUE` maps to `entailment`, `PROBABLE` to `neutral` (for the three-class `at` target), and `FALSE` to `contradiction`. For the binary `isAt` target, only the entailment and contradiction categories are mapped, although the encoder head retains the three standard NLI logits from its pre-trained state.

The two relation classifiers are fine-tuned independently. The training files are shuffled with seed 42 and split 80/20 into train and validation partitions. Tokenization feeds paired premises and hypotheses to the model under dynamic padding with a maximum sequence length of 512 tokens; text exceeding this length is truncated with no sliding-window aggregation or passage retrieval. Both training runs use identical hyperparameters: 2 epochs, a per-device batch size of 16, gradient accumulation of 2 (effective batch size of 32 on a single GPU), a learning rate of $9{\cdot}10^{-6}$ using AdamW (β_1=0.9, β_2=0.999, ϵ=10^{-8}), weight decay of 0.01, and a linear schedule with 6% warm-up under automatic FP16 precision. Regularization techniques like label smoothing or class weighting are disabled. The selected checkpoint for each relation is chosen based on the epoch maximizing validation macro recall. During inference, class prediction follows the argmax over the NLI softmax, mapped back to the HIPE-2026 schema as detailed in §3.6.

3.4. Implementation and Hardware Setup

Our computational setup is divided into local serving and cloud SFT environments. Inference and local optimization run on a single consumer workstation equipped with an AMD Ryzen 7 5700X CPU, 64 GB DDR4-3200 memory, and a single NVIDIA RTX 2060 6 GB GPU. The NLI encoder runs natively on the GPU, while language models are served locally via LM Studio under GPU execution. Supervised fine-tuning is offloaded to an NVIDIA A100 40 GB cloud instance hosted on Modal. Larger teacher calls during GEPA prompt optimization are routed through the OpenRouter API.

To support the replication and reuse of our findings, we release all trained parameters including the distilled LFM2.5 rewriter adapters, fine-tuned mDeBERTa checkpoints, and Nemotron student weights in a public Hugging Face collection.[7] The core modeling, data generation, and evaluation codebase is publicly hosted on GitHub.[8].

3.5. Approach B: Small Language Model with Prompt Optimization and SFT

Approach B centers on a compact Small Language Model (SLM), Nemotron-3-Nano-4B, and automated prompt optimization rather than hand-crafted instruction tuning. For each relation target (`at` and `isAt`), we instantiate a chain-of-thought DSPy program [9] from a compact *initial prompt*: a signature that specifies the inputs (raw multilingual OCR text, candidate person and location mentions, publication date) and the required outputs. Each completion returns a structured JSON object with two fields: a step-by-step `reasoning` trace of why the relation holds or does not hold, and a final `classification` label (`TRUE`, `FALSE`, or `PROBABLE` for `at`; `TRUE` or `FALSE` for `isAt`). Starting from this seed program, Genetic-Pareto (GEPA) [7] evolves the instructions and field schema on **Gold Train A** until the serialized prompt program converges; we do not perform separate manual prompt iteration beyond this initial template. The optimized program is then used to construct a gold-filtered dataset for the SFT stage:

[6] `mDeBERTa-v3-base-xnli-multilingual-nli-2mil7` [5]
[7] https://huggingface.co/collections/andersonscode/hipe
[8] https://github.com/VerbaNexAI/CLEF2026

we run the training dataset through the model using the optimized GEPA program, retaining only the completions where the predicted relation matches the correct gold labels, and use these correct reason label paths to teach the student model how to reason and format its responses. This gold-filtered corpus is distilled into the same backbone via LoRA supervised fine-tuning (SFT), yielding the submitted student checkpoint (`nemotron-3-nano-4b-hipe-po-geta`).

Each relation program is optimized independently with GEPA. The training pairs are split 80/20 into train and validation sets. GEPA is executed in its lightweight configuration (seed 42, two threads, and a reflection minibatch size of 40) using an exact-match metric that feeds back textual error descriptions and temporal anchors to the generator. High-scoring examples are bypassed during reflection to prioritize error-prone cases. The final instructions and field schemas are serialized as prompt templates, omitting static in-context demonstrations to preserve sequence length capacity.

Both prompt optimization and subsequent SFT label generation are driven by the same backbone: Nemotron-3-Nano-4B, hosted via a local OpenAI-compatible endpoint. This symmetric architecture means that the model acts as both the *teacher* during label generation and the *student* during SFT. Choosing self-distillation serves a specific methodological purpose: rather than attempting to inject external reasoning capacity from a larger closed-source model, self-distillation isolates the effect of structural alignment. It shapes the student's formatting behavior (specifically, schema compliance and the strict ordering of the `reasoning` and `classification` fields) while keeping the model's underlying epistemic boundaries and cross-lingual reasoning capacity unchanged.

Gold-filtered teacher completions. We use this term for the supervised fine-tuning corpus. After GEPA converges, the teacher runs the optimized DSPy program over every person location pair in **Gold Train A**. Each run yields a *teacher completion*: a structured assistant response containing `reasoning` and `classification`. We *gold-filter* these outputs by retaining a completion only when its `classification` exactly matches the dataset gold label; all mismatches are discarded. The surviving instruction completion pairs therefore pair the correct relation label with a teacher-generated rationale, without manual relabeling. In the submitted configuration, teacher and student share the same Nemotron backbone; ablations in Table 4 apply the same filtering rule with larger external teachers.

During training label generation, the model's temperature is restricted to 0.1. This low-entropy setting ensures deterministic adherence to the JSON schema and minimizes stochastic label flip. To process the corpus efficiently, predictions are generated asynchronously under a concurrency semaphore, with a single retry permitted per call.

Despite the optimized prompt, small language models frequently produce malformed JSON or conversational padding that escapes standard parser schemas during inference. To handle these edge cases at test time, we implement a tolerant, two-stage extraction system. The parser first attempts to extract a JSON block using string boundary slicing. If this fails due to truncated tokens or structural malformations, a regular expression fallback scans the assistant's terminal tokens using case-insensitive anchor patterns (e.g., `(true|false|probable)`). If both stages fail to resolve a valid target, the prediction is collapsed to FALSE (the dataset's majority class) to prevent execution crashes. Significantly, this recovery process is not applied during the creation of the gold-filtered SFT dataset; any candidate teacher completion that is malformed or requires parsing fallbacks is discarded to ensure that the student is fine-tuned exclusively on pristine, natively syntactically valid reasoning label pairs.

Each retained completion is formatted as a chat turn: the instruction prompt paired with the gold-consistent assistant response (`reasoning` and `classification`). The student is fine-tuned exclusively on these correct reasoning label paths.

The student model, Nemotron-3-Nano-4B, is fine-tuned on this gold-filtered dataset by applying the model's chat template and masking the loss to compute gradients only on the assistant responses. The SFT run utilizes a maximum sequence length of 2048 tokens, 16-bit precision, and LoRA adapters [8] (r=α=16) with zero dropout and no bias terms. The adapters target the attention projections (`q_proj`, `k_proj`, `v_proj`, `o_proj`), MLP layers (`gate_proj`, `up_proj`, `down_proj`), and Mamba-style projections (`in_proj`, `out_proj`). Training is conducted for 120 optimization steps with a learning rate of $2{\cdot}10^{-4}$

using 8-bit AdamW, a linear schedule with a 5-step warm-up, and seed 3407. The resulting model, `nemotron-3-nano-4b-hipe-po-geta`, is our submitted student checkpoint.

To explore the scaling properties of SFT, we also trained student variants using larger teachers specifically, GPT-OSS-20B [11] and Step-3.5-Flash [12] (see Table 4). While the Step-3.5-Flash-distilled student achieved the highest macro recall on the English split, the latency and cost of routing whole dataset calls through proprietary APIs made scaling these teachers to the entire multilingual corpus impractical during the official competition window. Consequently, the self-distilled student was selected for the final submission.

Significantly, our submitted configuration introduces a deliberate mismatch between training and inference: the fine-tuned student is paired with a prompt program evolved under GPT-OSS-20B, rather than the prompt program under which its training data was generated. This mismatch serves as a prompt-transfer robustness test. It evaluates whether a student model whose formatting and structural output habits have been stabilized via self-distillation can generalize its reasoning capabilities to instructions optimized for a larger, structurally different model. The serving configuration and temperature settings used to execute this combination are detailed in §3.6.

3.6. Inference and Submission

At inference time, predictions are generated concurrently across the two modeling tracks. For Approach A (the fine-tuned NLI encoder), we apply the deterministic premise-hypothesis pairing described in §3.3. Tokenized pairs are forwarded to the fine-tuned mDeBERTa-v3 checkpoint, and the predicted label corresponds to the argmax over the NLI softmax distribution, without further threshold calibration or class-weight scaling.

For Approach B (the prompt-optimized SLM), we reload the serialized DSPy program evolved under GPT-OSS-20B. These program nodes are dispatched asynchronously against a local OpenAI-compatible endpoint hosted on LM Studio, with our distilled Nemotron student (`nemotron-3-nano-4b-hipe-po-geta`) as the active model.

Crucially, while the SFT label generation stage runs at a restricted temperature of 0.1 to enforce structural conformity and avoid label drifting, inference is executed at a higher temperature of 1.0. This temperature contrast is a deliberate design choice: when running local GPU inference on long, noisy historical sequences, a deterministic temperature (0.1) often leads small models to get stuck in repetitive token loops or produce overly sterile, truncated reasoning paths. Elevating the temperature to 1.0 at inference encourages the SLM to produce more diverse, natural language reasoning rationales before emitting its final label.

To handle potential formatting failures under this higher temperature, we deploy the tolerant extraction stack outlined in §3.5. The system first attempts to parse a structured JSON block containing the `classification` field; if the block is malformed, a case-insensitive regex fallback scans the final tokens of the assistant completion to map free-text responses back to the target label set. The two target queries (`at` and `isAt`) are evaluated concurrently for each pair using an in-flight concurrency semaphore. Predictions are serialized as JSON Lines conforming exactly to the HIPE-2026 submission schema; the step-by-step `reasoning` trace is generated at inference time for auditability but stripped out prior to file submission.

4. Results

We evaluate both architectures under various training and test conditions to characterize their performance on historical datasets. This section presents our empirical findings, starting with our validation protocols and development results, proceeding to held-out test evaluations, and contextualizing our submitted runs on the official HIPE-2026 leaderboards. Finally, we provide a detailed analysis of the performance trade-offs inherent in each approach, examining the impact of machine translation and OCR correction across English, French, and German splits.

4.1. Experimental Protocol

Development experiments are evaluated on the manually annotated training splits (`train-en`, `train-fr`, and `train-de`), which correspond to the official **Gold Train A** (newspaper training set) summarized in Table 1.[9] Test-set evaluation in §4.3 uses the official gold labels for **Test A** and **Test B** summarized in Table 2. The Approach B SFT corpora are gold-filtered subsets of **Gold Train A** (teacher completions whose predicted label matches gold; see §3.5), so development scores for SFT configurations have a direct overlap between training supervision and evaluation gold; this structurally inflates absolute numbers on the training splits but does not reorder the relative ranking of model ablations. Model selection is based on global macro recall, averaged across both relations and their classes, with per-relation metrics as auxiliary diagnostics, and `null` predictions collapse to FALSE per the official scorer.[10]

4.2. Development-Set Results

Table 3 and Table 4 report the macro recall scores across the various experimental conditions and ablations on the development training splits. For the NLI encoder (Approach A), performance is highly sensitive to the document preprocessing and translation pipeline. On the English split (`train-en`), aligning and repairing the OCR text using DeepSeek-V3.2 yields a substantial increase in macro recall (0.762 → 0.815). Similarly, on the German split (`train-de`), both the LFM2.5 and DeepSeek-V3.2 translated views outperform the raw multilingual OCR text, lifting the macro recall score from 0.484 to 0.515. Conversely, on the French split (`train-fr`), we observe an inverse trend: the raw multilingual text achieves the highest recall (0.506), whereas translating the articles into English via LFM2.5 (0.485) or DeepSeek-V3.2 (0.480) results in a mild performance drop. We analyze the trade-offs of this translation pipeline in §4.5.

For the LLM-based program (Approach B), prompt optimization (GEPA) and supervised fine-tuning (SFT) yield consistent improvements over the stock Nemotron baseline (0.591 macro recall on English). Under our prompt-transfer ablation, configuring the student with a prompt optimized on the larger GPT-OSS-20B model yields a score of 0.654, which is further enhanced to 0.660 when fine-tuning the Nemotron student on correct reasoning chains distilled from Step-3.5-Flash. Our submitted Nemotron checkpoint which is self-distilled and executed at temperature 1.0 reaches 0.600 on English, 0.584 on French, and 0.417 on German when evaluated on the training splits. The performance drop observed on the German split is concentrated on the three-class `at` target (accuracy 0.577 versus 0.736 on French), reflecting the sparse distribution of the PROBABLE label and severe OCR degradations in historical German text.

Table 3
Development-set Macro Recall Performance: Encoder-based Classifier (mDeBERTa-v3). Each value represents global macro recall evaluated on the labeled training splits. **RAW** = original released data (no OCR repair or translation); **LFM2.5/DeepSeek-V3.2** = text preprocessed and translated using those model backbones.

Data Split	Preprocessing	Macro Recall
`train-en`	DeepSeek-V3.2 rewrite	**0.815**
	LFM2.5 rewrite	0.768
	RAW	0.762
`train-fr`	RAW	**0.506**
	LFM2.5 rewrite	0.485
	DeepSeek-V3.2 rewrite	0.480
`train-de`	LFM2.5 rewrite	**0.515**
	DeepSeek-V3.2 rewrite	0.515
	RAW	0.484

[9]The HIPE-2026 v1.0 data release is available at https://github.com/hipe-eval/HIPE-2026-data/releases/tag/v1.0
[10]https://github.com/hipe-eval/HIPE-2026-data/releases/tag/v1.0

Table 4
Development-set Macro Recall Performance: LLM-based (Nemotron). Each number is a macro recall point estimate on labelled training splits. **Step-3.5-Flash** is a compact StepFun model used as a GEPA comparator. For Approach B, SFT+GEPA+... (Step-3.5-Flash) means Nemotron trained on gold-filtered GEPA labels from that comparator, evaluated on DeepSeek-V3.2 English rewrite.

Model/Condition (English, `train-en`)	**Macro Recall**
SFT + GEPA + DeepSeek rewrite (Step-3.5-Flash)	0.660
GEPA GPT-OSS-20B	0.654
GEPA Step-3.5-Flash	0.609
GEPA Nemotron + DeepSeek rewrite	0.608
GEPA GPT-OSS-20B + DeepSeek rewrite	0.602
Nemotron baseline	0.591
GEPA GPT-OSS-20B + LFM2.5 OCR fix	0.586
Submitted checkpoint (`nemotron-3-nano-4b-hipe-po-geta`):	
`train-en`	0.600
`train-fr`	0.584
`train-de`	0.417

4.3. Held-out Test-Set Results

Table 5 compares the two systems on the official gold labels for **Test A** and **Test B**. Approach B (the prompt-optimized, self-distilled SLM) consistently achieves higher macro recall point estimates than Approach A (the fine-tuned mDeBERTa-v3 encoder) across all evaluation cells. This performance gap is widest on the French split (`Test A fr`), where the SLM outperforms the encoder by +0.128 in macro recall. On the German (`Test A de`) and French surprise literary splits (`Test B fr`), the SLM's advantage is +0.089 and +0.071, respectively, while the narrowest gap occurs on the English split (`Test A en`) at +0.020.

Analyzing per-target accuracies reveals that the performance gap is highly dependent on task complexity. On the binary `isAt` target, the encoder achieves competitive accuracies, lagging behind the SLM by only 0.02 to 0.09 points. However, on the three-way `at` target, the encoder's performance drops substantially, with accuracy gaps widening to 0.06 to 0.18 points. We offer a methodological and linguistic interpretation of these differences in §4.5.

Table 5
Held-out evaluation on **Test A (newspaper test set)** and **Test B (French literary surprise test set)**: mDeBERTa-v3 encoder (Approach A) versus submitted self-distilled Nemotron-3-Nano-4B student (Approach B, paired with the GEPA prompt evolved under GPT-OSS-20B). Bold values denote the higher macro recall per row; Δ is the SLM minus encoder macro recall.

	Approach A (Encoder)			Approach B (SLM)			
Set	acc(`isAt`)	acc(`at`)	Macro Recall	acc(`isAt`)	acc(`at`)	Macro Recall	Δ
Test A en	0.654	0.599	0.561	0.691	0.660	**0.581**	+0.020
Test A fr	0.639	0.592	0.460	0.727	0.773	**0.588**	+0.128
Test A de	0.693	0.613	0.481	0.765	0.706	**0.570**	+0.089
Test B fr	0.969	0.629	0.463	0.990	0.771	**0.534**	+0.071

4.4. Official Ranking Context

Following the release of the official HIPE-2026 results, we contextualized our submitted runs against the leaderboards of the official evaluation campaign [1, 2]. Under the official shared-task mapping, our primary submission, `VerbaNexAI II run3`, corresponds to Approach B (the prompt-optimized,

self-distilled SLM), while `VerbaNexAI II run1` represents Approach A (the fine-tuned mDeBERTa-v3 encoder). In total, the CLEF shared task attracted **17 participating teams** who submitted a combined total of **45 official runs**.

Table 6 compares our submitted runs against the top-performing systems across the three official evaluation profiles: the **Accuracy profile**, the **Generalization profile**, and the **Accuracy–Efficiency profile**. For a complete ranking of all participant runs, we refer readers to the official shared-task overview papers [1, 2].[11]

In the Accuracy profile, our SLM run (`run3`) achieved 18th place with a score of 0.58, and our NLI encoder run (`run1`) placed 28th with a score of 0.50. In the Generalization profile, `run3` placed 18th (0.57), and `run1` placed 28th (0.44). Under the Accuracy–Efficiency profile, which evaluates relation extraction quality under tight computational constraints, our local configurations achieved highly competitive standings: our SLM run (`run3`) climbed to 12th place (achieving a mean rank score of 17.00 with 0.58 accuracy), and our NLI encoder run (`run1`) placed 17th.

As defined by the organizers, the efficiency score represents the arithmetic mean of three distinct ranks: accuracy rank, model parameter count rank, and deployed model size rank. This score is not a recall or accuracy percentage; rather, a lower rank score is superior. These results highlight that our compact, locally hosted, low-parameter setups can deliver viable extraction performance while minimizing hardware footprints and api-cost overheads relative to heavily parameterized, closed-source API ensembles.

Table 6
Official HIPE-2026 rankings: best teams and VerbaNexAI II runs. Systems are ranked by profile score across the Accuracy, Generalization, and Accuracy–Efficiency profiles (higher is better for Accuracy and Generalization; lower mean rank score is better for Accuracy–Efficiency, with official accuracy in parentheses).

System	Accuracy	Generalization	Acc.–Efficiency
Best official run	Spinfo: 1st (0.75)	MaxFo-Ajie: 1st (0.82)	MILRIT: 1st (9.67, 0.60 acc.)
VerbaNexAI II run3 (Approach B)	18th (0.58)	18th (0.57)	12th (17.00, 0.58 acc.)
VerbaNexAI II run1 (Approach A)	28th (0.50)	28th (0.44)	17th (18.67, 0.50 acc.)

4.5. Analysis and Discussion

The empirical comparison between our NLI encoder and SLM pipelines reveals several distinct methodological trade-offs in cross-lingual information extraction under historical noise.

First, we observe a substantial generalization gap when transferring systems from training to held-out test splits. For the NLI encoder, macro recall on the English split drops from a peak validation score of 0.815 on development to 0.561 on the held-out test set. This 0.254 drop highlights a strong evaluation-on-train bias, where hyperparameter configurations and checkpoints optimized against small, highly skewed training splits overfit the training domain and fail to generalize robustly.

Second, the preprocessing ablations in Table 3 demonstrate that machine translation and OCR correction act as a double-edged sword. Under Approach A, translating French text into English and repairing OCR errors slightly degrades performance (0.506 RAW versus 0.480 DeepSeek-V3.2 rewrite). We attribute this drop to the phenomenon of *entity translation drift*: translation models frequently misinterpret, transliterate, or entirely drop historical proper nouns or regional spelling variations (such as changing historical surnames or archaic village names into common English words). This semantic drift breaks the alignment between the entity mentions in the premise and those in the hypothesis, leading to false negatives.

Conversely, German text benefits markedly from preprocessing, with macro recall rising from 0.484 to 0.515. Historical German newspapers are heavily affected by compounding word structures, segmentation issues, and gothic-script (Fractur) OCR degradations. For German, the structural benefit of restoring word boundaries and resolving Fractur typeface errors outweighs the linguistic tax of

[11]The shared task submission files and evaluation orchestrator can be found at https://github.com/hipe-eval/HIPE-2026-eval

translation drift. Consequently, unifying the source text into high-resource English is highly beneficial for German, but counterproductive for French, where the original OCR is cleaner.

Third, we introduce a necessary note of scientific caution regarding the observed performance differences. While our SLM point estimates are higher than those of the encoder across all splits (e.g., +0.020 on English and +0.128 on French), these results are computed over small evaluation samples (such as the 162 English pairs in Test A) without associated standard deviations or confidence intervals. Accordingly, small point-estimate variations should be interpreted descriptively rather than as statistically established advantages.

The rise in German SLM performance from development to test (0.417 → 0.570) is similarly descriptive and most likely reflects a distributional artifact rather than a true learning effect: a higher proportion of FALSE labels on the test split structurally inflates the macro recall score under highly skewed three-class targets. Furthermore, on the Test B French surprise set, the official gold labels assign `isAt=FALSE` to every pair, which simplifies the binary classification head and accounts for the high binary accuracy (0.969 and 0.990).

Despite these quantitative caveats, Approach B retains a compelling qualitative advantage over the encoder pipeline. By outputting a structured JSON payload with a step-by-step `reasoning` trace, the SLM provides an auditable decision path. In professional archiving and document curation, having a compact, 4B-parameter model generate a self-contained justification for why a historical person was likely present in a specific city is of significant practical value for human-in-the-loop validation, even where the quantitative recall advantage over an encoder is narrow.

5. Future Work

Several promising avenues emerge for enhancing both the NLI encoder and prompt-optimized SLM architectures.

First, resolving input sequence limitations represents a critical bottleneck. Under Approach A, the mDeBERTa encoder's 512-token limit truncates longer historical documents, particularly in French and German. Future iterations will evaluate sliding-window context pooling, candidate-conditioned span selection, or a sparse retrieval-augmented generation (RAG) module to extract target entity passages before classification. These modifications can be evaluated directly against the official **Test A** and **Test B** gold labels without requiring a full retraining of the backbone.

Second, the natural language rationales generated by Approach B can be leveraged for active, self-reflective verification. Rather than treating the `reasoning` field as a passive logging output, future pipelines will introduce a multi-turn self-correction step. By prompting the student model to evaluate its own generated reasoning path for logical contradictions before emitting its final label, we can enforce consistency between the rationales and classifications.

Third, addressing the severe label imbalance remains essential. As shown in Table 1, the HIPE-2026 splits are highly skewed toward the FALSE class, with the PROBABLE class representing under 10% of overall samples. Implementing class-weighted or focal loss objectives for the encoder, alongside synthetic minority-class augmentation (such as generating additional positive cases using frontier LLMs) for the SLM's fine-tuning corpus, should be explored to elevate the recall floor of both systems.

6. Conclusion

In this work, we developed and compared two low-parameter, locally deployable architectures for person-place relation extraction on historical, OCR-noisy multilingual documents.

Evaluating both tracks on the official **Test A** and **Test B** gold labels reveals that our prompt-optimized, self-distilled SLM pipeline (Approach B) consistently outperforms our fine-tuned NLI encoder pipeline (Approach A) in macro-recall point estimates. This advantage is widest on French (+0.128 macro recall) and narrowest on English (+0.020), with German and Test B in between (+0.089 and +0.071).

While the SLM point estimates are higher, we advocate for scientific caution in interpreting these small differences due to the absence of uncertainty bounds and the restricted scale of the evaluation sets. Nonetheless, the SLM's capacity to generate natural language explanations along with its classifications offers substantial practical utility. For professional archivists and curators of historical newspapers, these self-contained reasoning paths provide an auditable signal that simplifies human-in-the-loop verification a benefit that remains valuable even when the quantitative performance gap between models is narrow.

Our official rankings on the CLEF shared-task leaderboard reflect the competitive efficiency of our approaches, showing that low-parameter local configurations can achieve viable classification accuracy while operating within tight consumer hardware and api-cost constraints.

Acknowledgments

We dedicate this work to the master's degree scholarship program in Engineering at the Universidad Tecnologica de Bolivar (UTB) in Cartagena, Colombia.

We want to express our gratitude to the team at the VerbaNex AI Lab [12] for their dedication, collaboration, and ongoing support of our research endeavors.

Declaration on Generative AI

During the preparation of this work, the author(s) used Composer 2.5 and DeepSeek V3.2 to make grammar and spelling checks, improve writing style, text translation and draft content. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] J. Opitz, M. Ehrmann, C. Raclé, A. Michail, M. Romanello, S. Clematide, Overview of HIPE-2026: Person–Place Relation Extraction from Multilingual Historical Texts, in: M. Hagen, M. Potthast, B. Stein, P. Schaer, E. Zangerle, S. MacAvaney, J. M. Struß, E. Sánchez Salido, A. Barrón-Cedeño, A. García Seco de Herrera (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Seventeenth International Conference of the CLEF Association (CLEF 2026), Lecture Notes in Computer Science (LNCS), Springer, 2026.

[2] J. Opitz, M. Ehrmann, C. Raclé, A. Michail, M. Romanello, E. Boros, S. Gabay, S. Clematide, Extended Overview of HIPE-2026: Evaluating Accurate and Efficient Person–Place Relation Extraction from Multilingual Historical Texts, in: E. Sánchez Salido, A. Barrón-Cedeño, A. García Seco de Herrera, S. MacAvaney, J. M. Struß (Eds.), CLEF 2026 Working Notes, CEUR Workshop Proceedings, CEUR-WS, 2026. doi:10.5281/zenodo.20344461.

[3] J. Opitz, C. Raclé, E. Boros, A. Michail, M. Romanello, M. Ehrmann, S. Clematide, CLEF HIPE-2026: Evaluating accurate and efficient person–place relation extraction from multilingual historical texts, in: Advances in Information Retrieval (ECIR 2026), Springer Nature Switzerland, Cham, 2026, pp. 354–363. doi:10.1007/978-3-032-21321-1_46.

[4] M. Ehrmann, M. Romanello, S. Najem-Meyer, A. Doucet, S. Clematide, Overview of HIPE-2022: Named entity recognition and linking in multilingual historical documents, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. CLEF 2022, Springer-Verlag, Berlin, Heidelberg, 2022, pp. 423–446. doi:10.1007/978-3-031-13643-6_26.

[5] M. Laurer, W. van Atteveldt, A. Casas, K. Welbers, Less annotating, more classifying: Addressing the data scarcity issue of supervised machine learning with deep transfer learning and bert-nli, Political Analysis 32 (2024) 84–100. doi:10.1017/pan.2023.20.

[12] https://verbanexai.github.io/

[6] NVIDIA, :, A. Blakeman, A. Grattafiori, A. Basant, A. Gupta, A. Khattar, A. Renduchintala, A. Vavre, A. Shukla, A. Bercovich, A. Ficek, Other, Nvidia nemotron 3: Efficient and open intelligence, 2025. URL: https://arxiv.org/abs/2512.20856. arXiv:2512.20856.

[7] L. A. Agrawal, S. Tan, D. Soylu, N. Ziems, R. Khare, K. Opsahl-Ong, A. Singhvi, H. Shandilya, M. J. Ryan, M. Jiang, C. Potts, K. Sen, A. G. Dimakis, I. Stoica, D. Klein, M. Zaharia, O. Khattab, GEPA: Reflective prompt evolution can outperform reinforcement learning, 2025. URL: https://arxiv.org/abs/2507.19457. arXiv:2507.19457.

[8] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, LoRA: Low-rank adaptation of large language models, 2022. URL: https://arxiv.org/abs/2106.09685. arXiv:2106.09685, published at ICLR 2022.

[9] O. Khattab, A. Singhvi, P. Maheshwari, Z. Zhang, K. Santhanam, S. Vardhamanan, S. Haq, A. Sharma, T. T. Joshi, H. Moazam, H. Miller, M. Zaharia, C. Potts, DSPy: Compiling declarative language model calls into self-improving pipelines, 2024. URL: https://arxiv.org/abs/2310.03714. arXiv:2310.03714, published at ICLR 2024.

[10] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, in: D. Jurafsky, J. Chai, N. Schluter, J. Tetreault (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 8440–8451. URL: https://aclanthology.org/2020.acl-main.747/. doi:10.18653/v1/2020.acl-main.747.

[11] OpenAI, :, S. Agarwal, L. Ahmad, J. Ai, S. Altman, A. Applebaum, E. Arbus, R. K. Arora, Y. Bai, B. Baker, H. Bao, B. Barak, Others, gpt-oss-120b & gpt-oss-20b model card, 2025. URL: https://arxiv.org/abs/2508.10925. arXiv:2508.10925.

[12] A. Huang, A. Li, A. Kong, B. Wang, B. Jiao, B. Dong, B. Wang, B. Chen, B. Li, B. Ma, C. Su, C. Miao, Others, Step 3.5 flash: Open frontier-level intelligence with 11b active parameters, 2026. URL: https://arxiv.org/abs/2602.10604. arXiv:2602.10604.