Lightweight Person–Place Relation Extraction from Historical Newspapers with Dependency Graphs and Proximity Features

Working Notes paper for the HIPE-2026 Shared Task at CLEF 2026

Mlen-Too Wesley[1]

[1]*Georgia Institute of Technology, Atlanta, GA, USA*

Abstract

The HIPE-2026 shared task introduces person–place relation extraction from multilingual historical newspapers as a new evaluation track, classifying the `at` and `isAt` relations between pre-annotated person and location mentions in English, French, and German. Motivated by the cost of processing historical archives at scale, our team (DS@GT HIPE, team 2 in the official results) investigates how far a lightweight, interpretable system can go without any pretrained language model at the relation classification stage. Our approach builds a document-level graph from dependency parses, extracts proximity-based and part-of-speech features for each entity pair, and classifies them with small scikit-learn ensembles or compact Graph Attention Networks, keeping every submitted run under 847K parameters. On the official evaluation (Test A, the newspaper test set), our best run reached a macro recall of 0.5142, ranking 3rd on the Efficiency profile while placing mid-table on Accuracy among the 17 participating teams. Two findings stand out. First, minimum character distance alone captures most of the classification signal; adding further engineered features yields inconsistent gains and sometimes degrades performance, echoing prior evidence that argument distance dominates relation extraction. Second, document-grouped cross-validation is essential on this corpus: pair-level splits inflate scores by 25–37 percentage points because entity mentions recur across documents, a data-leakage effect that grouped cross-validation removes.

Keywords

relation extraction, historical newspapers, dependency parsing, graph attention networks, feature engineering, efficiency

1. Introduction

HIPE-2026 is the third edition of the HIPE shared task series [1, 2], organized as part of CLEF 2026. While earlier editions focused on named entity recognition (NER) and entity linking in historical documents, HIPE-2026 introduces relation extraction (RE) between pre-annotated person and location entity mentions [3, 4].[1] For each (person, location) pair in a document, participating systems classify two relations. The `at` relation captures whether the text supports the inference that the person was present at the location at some point in time; it is annotated as a graded judgement with the labels TRUE, PROBABLE, or FALSE. The `isAt` relation captures the narrower notion of the person being described as physically located at the place, annotated as a binary TRUE or FALSE. The official evaluation metric is macro recall, also called balanced accuracy [5], and systems are assessed under three profiles: Accuracy, Efficiency, and Generalization. A total of 17 teams submitted 45 runs.

We approach the task with a focus on efficiency. Historical newspaper processing at scale requires lightweight models that can run on modest hardware [6, 7], yet most relation extraction systems rely on large pretrained language models with substantial parameter budgets and inference costs. We ask how far engineered features derived from syntactic structure can carry relation classification without any pretrained language model at the relation classification stage. FastText embeddings [8] are used

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany

mwesley32@gatech.edu (M. Wesley)

0009-0002-7979-1599 (M. Wesley)

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

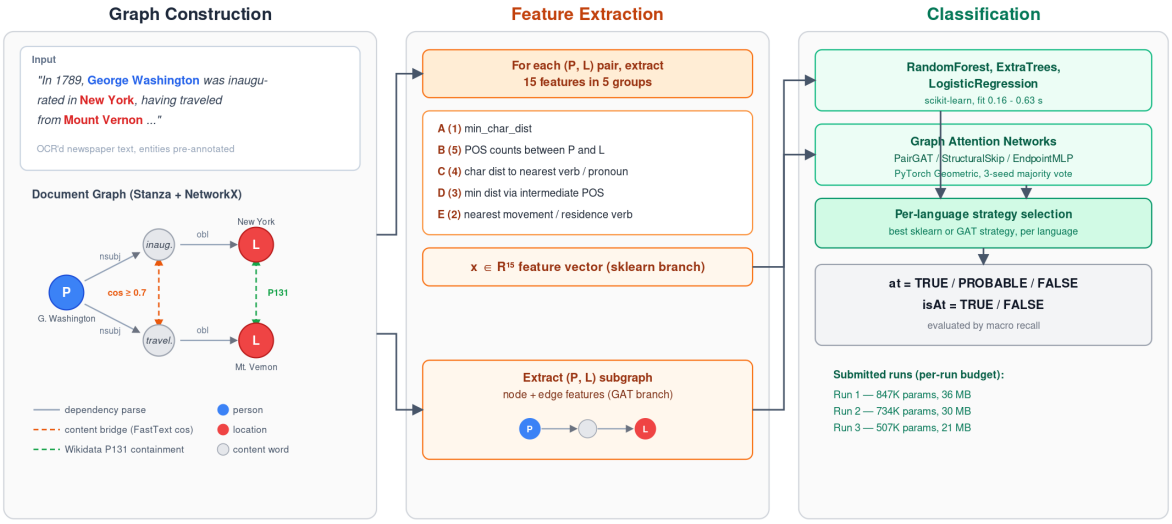

Figure 1: Three-stage pipeline. The first stage assembles Stanza dependency parses into a document-level graph with cross-sentence bridges (content-word cosine similarity and Wikidata P131 containment). The second stage extracts 15 proximity-based features and a per-pair subgraph for each entity pair. The third stage classifies via scikit-learn ensembles (from the feature vector) or Graph Attention Networks (from the subgraph), with per-language strategy selection.

during graph construction for cross-sentence bridging, but no embedding or transformer representation enters the classification pipeline itself.

Figure 1 illustrates the system architecture. Our system follows a three-stage pipeline consisting of graph construction from dependency parses, extraction of 15 proximity-based and POS-based features, and classification via scikit-learn ensembles [9] or small Graph Attention Networks built with PyTorch Geometric [10]. Per-run parameter counts range from 506,923 to 846,701, with model sizes of 21–36 MB. Run 1 achieved a macro recall of 0.5142, ranking 26th of 46 on accuracy and 3rd overall in the efficiency profile.

The remainder of this paper is organized as follows. Section 2 surveys related work. Section 3 describes the data. Section 4 presents the system in detail. Section 5 reports experiments and results. Section 6 concludes.

2. Related Work

HIPE shared task lineage. CLEF HIPE-2020 introduced NER evaluation on historical newspapers with 13 participating teams [1], and HIPE-2022 expanded to NER and entity linking across multiple multilingual historical datasets [2]. HIPE-2026 [3, 4] introduces relation extraction as a new task type. OCR noise, domain heterogeneity, and diachronic language change remain persistent challenges for NLP on historical text [11, 12], and these challenges compound for RE, where upstream parse errors propagate into both graph structure and feature quality.

Dependency-based and graph-based RE. Bunescu and Mooney [13] established that the shortest dependency path between two entities carries strong signal for relation classification. Subsequent work applied graph neural networks over dependency trees. Zhang et al. [14] proposed C-GCN over pruned trees, and Guo et al. [15] introduced attention-guided graph convolutions. Nan et al. [16] constructed document-level graphs from dependency parses using spaCy and NetworkX, the closest published precedent to our graph construction approach. Peng et al. [17] addressed cross-sentence RE through graph LSTMs over merged sentence-level dependencies. Document-level RE benchmarks such as DocRED [18] formalize the multi-sentence setting that our document graphs target. Our system

Table 1
Training data by language, showing document and entity pair counts from the Impresso corpus.

Language	Documents	Entity Pairs
English	35	307
French	35	478
German	34	466
Total	104	1,251

differs from these approaches in that it uses the graph primarily for feature computation rather than learned message passing, with the GAT branch as a secondary pathway.

Feature engineering and proximity dominance. Jiang and Zhai [19] systematically explored RE feature spaces and found that complex subspaces provide marginal gains over simpler sets; Zhou et al. [20] catalogued the relative contributions of lexical, chunk, and parse-tree features. Zeng et al. [21] showed that position features are the single most informative group for relation classification, position-aware attention over the TACRED benchmark likewise relies heavily on argument position [22], and Alt et al. [23] confirmed via probing experiments that argument distance accounts for more RE performance than verb or POS semantics. These findings align with our result that minimum character distance alone captures most of the classification signal (Section 5.3). For biographical RE specifically, de-Dios-Flores et al. [24] found that basic lemmatized features outperform deep syntactic dependency features for person-place relations in Portuguese and Spanish, which directly parallels our negative finding on verb semantic features (Group E).

Lightweight RE and historical/biographical text. Giuliano et al. [25] showed that shallow linguistic features consistently outperform deep syntactic parsing for RE on biomedical text, where parser errors propagate through complex feature representations; OCR-degraded historical text presents an analogous noise profile. Minard et al. [26] achieved 3rd place in the i2b2 2010 clinical RE shared task using SVMs over engineered surface and distance features with no neural components. Lightweight feature-based systems such as LightRel [27] likewise remain competitive on standard relation-classification benchmarks with minimal engineering. Plum [28] developed language-agnostic biographical RE using Wikipedia-to-Wikidata alignment, providing precedent for our Wikidata containment edges. HIPE-2026 is, to our knowledge, the first shared task targeting person–place RE from historical newspapers, and its explicit efficiency evaluation profile [6, 7] aligns with this lightweight tradition.

3. Data

The HIPE-2026 training data is drawn from the Impresso collection of digitized historical newspapers spanning the 19th and 20th centuries [3, 4]. Person and location entity mentions are pre-annotated. The task is to classify the relation between sampled (person, location) pairs, not to identify the entities. Table 1 summarizes the training data.

The test data consists of Impresso newspaper test sets for each language, with sizes determined by the organizers (collectively, Test A, the newspaper test set), plus a surprise French literary test set (Test B).[2] We addressed the surprise set by reusing models trained on French newspaper data without domain adaptation.

Three characteristics of the corpus shaped our system design. First, OCR noise degrades entity surface forms and surrounding context [11, 12]. Second, the training sets are small, with only hundreds of entity pairs per language, which limits the viability of data-hungry approaches. Third, entity mentions

[2]We follow the organizers' dataset naming throughout: Test A / Newspaper test set and Test B / French literary surprise test set.

Table 2
Bridging ablation. Each row removes one mechanism from the full pipeline. "Disconnected pairs" is the percentage of entity pairs with no graph path between them.

Configuration	Disconnected (%)	Avg. components
Full pipeline	30.6	4.7
Skip entity merging	32.1	5.0
Skip content-word linking	64.5	11.8
Skip Wikidata containment	36.1	4.9
Skip all bridges	77.0	14.2

recur across documents within the same newspaper, creating the potential for information leakage in naive cross-validation splits (Section 5.2).

4. System Description

4.1. Graph Construction

The system constructs a document-level graph for each input document as follows. First, each sentence is parsed with Stanza [29], a neural NLP toolkit providing tokenization, part-of-speech tagging, and dependency parsing, using its Universal Dependencies models for English, French, and German. Each sentence becomes a directed tree in a NetworkX DiGraph. We map multi-token entity mention spans to their syntactic head node.

Three bridging mechanisms connect otherwise disjoint sentence-level trees into a single document graph. *Entity merging* collapses nodes that share the same pre-annotated entity identifier, that is, different mentions of the same person or location, into a single node. *Content-word linking* identifies non-stopword content words (tagged NOUN or PROPN) from different sentences whose FastText [8] embeddings have a cosine similarity ≥ 0.7 and merges them, creating cross-sentence edges. *Geographic containment* adds edges between location entities that stand in a Wikidata P131 ("located in the administrative territorial entity") relationship.

Table 2 shows the effect of each bridging mechanism. Content-word linking is the dominant mechanism. Removing it increases the proportion of disconnected entity pairs from 30.6% to 64.5%. Without any bridging, 77.0% of entity pairs are in disconnected graph components.

4.2. Feature Extraction

From each document graph, we extract 15 features per entity pair, organized into five groups (Table 3). All features are proximity-based or POS-based; no embedding features or transformer representations enter the classification pipeline. The feature design draws on prior work showing that distance-based features dominate in relation classification [20, 19, 21].

4.3. Scikit-learn Classifiers

Each scikit-learn strategy trains two relation classifiers on the 15 features, one for `at` (3-class) and one for `isAt` (binary). We use three classifier types, all implemented with scikit-learn [9]. RandomForest uses 500 estimators with max depth 10; ExtraTrees uses 300 estimators with max depth 10 or unbounded; LogisticRegression uses L2 penalty with C=1.0 and StandardScaler preprocessing. All classifiers use balanced class weights (`class_weight="balanced"`) to handle label imbalance.

Training variants include per-language models and a multilingual-concatenated variant (pass_c) that trains on data from all three languages. Training times range from 0.16 to 0.63 seconds per strategy.

Table 3
Feature groups A–E. All 15 features are computed from the document graph and surface-level token positions.

Group	#	Description
A	1	Minimum character distance between any mention pair of the two entities
B	5	POS-tagged token counts between entities (verbs, nouns, proper nouns, pronouns) and a binary pronoun indicator
C	4	Character distances from each entity to the nearest verb and pronoun
D	3	Minimum character distances routed through intermediary verbs, nouns, and pronouns
E	2	Character distance to the nearest movement and residence verb

4.4. Graph Attention Networks

We evaluated four architectures based on the Graph Attention Network [30], implemented with PyTorch Geometric [10], all operating on per-pair subgraphs extracted from the document graph. Each node carries a 336-dimensional feature vector (300 from FastText plus 36 structural one-hots encoding node kind, pair role, alignment status, and Universal Dependencies POS tag). Each edge carries a 38-dimensional feature vector encoding the dependency relation label, direction, and edge type. All architectures jointly predict both `at` and `isAt` through a shared hidden layer that feeds separate classification heads.

PairGAT. The base architecture projects the 336-dim node features through a linear layer into a hidden space of hidden_dim $\times$ n_heads dimensions (default: $32 \times 4 = 128$), then applies two GATv2Conv layers [31] with LayerNorm and dropout after each. GATv2Conv was chosen over the original GATConv because the v2 variant fixes a known limitation in which all attention heads attend to the same neighbor regardless of query. The person and location endpoint embeddings are concatenated into a pair representation, optionally augmented by a scalar side-channel that projects proximity features through a small feed-forward network. This side-channel handles the 30% of entity pairs with no graph path. When the subgraph is degenerate, the GAT produces near-zero embeddings and the side-channel carries the prediction. A shared linear layer with ReLU and dropout (0.5) feeds two output heads for `at` (3-class) and `isAt` (binary).

PairGATBranched. In PairGAT, the FastText block occupies 300 of 336 input dimensions (89%), so the structural one-hots compete for only 11% of the input projection capacity. PairGATBranched addresses this by projecting the FastText and structural slices through separate linear layers (64-dim and 32-dim respectively) before concatenation, giving structural features a fixed 33% share of the hidden space. The concatenated 96-dim representation then passes through the same GATv2Conv layers and readout as PairGAT.

PairGATStructuralSkip. This variant uses the same GAT backbone as PairGAT but concatenates the raw structural features of the person and location endpoints (a skip connection of $2 \times 36 = 72$ dimensions) alongside the GAT-produced embeddings at the readout layer. The motivation is to test whether structural information (node kind, role, POS) is lost during message passing. If the skip connection helps, the GAT layers are diluting information that matters for classification. This variant is less invasive than PairGATBranched because the GAT layers themselves are unchanged.

Table 4
GAT architecture selection via document-grouped 5-fold cross-validation on the training data; we report the CV winner per language. The submitted runs used the architecture chosen by the newspaper held-out sweep (Table 17), which matches the CV winner for English but differs for French and German (see Section 4.5).

Language	Winner	CV macro avg	Wall time (s)
EN	PairGATStructuralSkip	0.549 ± 0.047	53
FR	PairGAT	0.557 ± 0.018	599
DE	EndpointMLP	0.544 ± 0.022	35

Table 5
Run-to-strategy mapping. Each cell names the strategy used for that language in that run.

	Run 1	Run 2	Run 3
EN	GAT StructuralSkip	RF + ExtraTrees	ExtraTrees + LR
FR	LR + ExtraTrees	RF + ExtraTrees	GAT EndpointMLP
DE	RF + ET (multilingual)	GAT PairGAT	RF + ExtraTrees

Table 6
Per-run resource budgets. Parameter counts are summed from the strategies used in each run; surprise-FR reuses the FR model and is not double-counted.

Run	Parameters	Model size (MB)
Run 1	846,701	36
Run 2	733,751	30
Run 3	506,923	21

EndpointMLP. A non-graph baseline that concatenates the raw 336-dim feature vectors of the person and location endpoints ($2 \times 336 = 672$ dimensions) and passes them through a feed-forward network with no message passing or attention. If EndpointMLP matches the GAT architectures, it indicates that graph reasoning adds no value beyond what the endpoint features already encode. That the non-graph EndpointMLP was selected for French by the newspaper held-out sweep (Table 17) and used in the submitted French GAT run suggests that for some language/corpus combinations the graph topology carries little additional signal.

We selected architectures per language via document-grouped 5-fold cross-validation (StratifiedGroupKFold), following a similar small-data setup to Mandya et al. [32]. Table 4 shows the selected architecture per language.

All GAT variants use hidden_dim = 32, n_heads = 4, n_layers = 2, dropout = 0.3, attention_dropout = 0.3, and head_dropout = 0.5. PairGATBranched uses fasttext_hidden = 64 and structural_hidden = 32. All architectures share a 128-dim shared head layer before the classification outputs. We produce GAT predictions by 3-seed majority vote to reduce variance from random initialization. Parameter counts range from 335,247 (EndpointMLP) to 511,503 (PairGATStructuralSkip), with model sizes of 5–8 MB per architecture.

4.5. Strategy Portfolio and Run Composition

Each of the three submitted runs combines one strategy per language. Table 5 shows the mapping, and Table 6 shows the resource budget per run. All three runs reuse the corresponding French strategy for the surprise literary French test set.

Table 7
CV leakage on English data. Naive CV uses pair-level StratifiedKFold; grouped CV uses StratifiedGroupKFold by document. Gap is in percentage points.

Classifier	Relation	Naive CV	Grouped CV	Gap
ExtraTrees	`at`	0.730	0.363	−37
RF	`at`	0.709	0.372	−34
ExtraTrees	`isAt`	0.807	0.527	−28
LR	`at`	0.542	0.395	−15

5. Experiments and Results

5.1. Evaluation Setup

The evaluation metric is macro recall, also called balanced accuracy; the per-relation scores are $\mathrm{MacroRecall}_{\texttt{at}}$ and $\mathrm{MacroRecall}_{\texttt{isAt}}$, and we report a per-language *global score* equal to their arithmetic mean. The overall Test A score averages the global score across English, French, and German. We report results under three protocols, and each results-table caption states which one it uses: (i) *document-grouped 5-fold cross-validation* (StratifiedGroupKFold) on the training data, used for model selection and ablation; (ii) an *internal newspaper held-out set* of 7 documents per language, used for cross-strategy comparison before submission; and (iii) the *official evaluation* on Test A (newspaper) and Test B (French literary surprise set). Our scikit-learn strategies come in three passes: *pass A* trains a fixed classifier pair per language, *pass B* selects the per-language cross-validation winner for each relation, and *pass C* trains on multilingual-concatenated data from all three languages.

5.2. Cross-Validation Protocol and Leakage Finding

Our initial cross-validation protocol used StratifiedKFold with pair-level splits. We discovered that entity mentions recur across documents within the same newspaper, so a person or location appearing in one training document often appears in a held-out document as well. Naive CV splits that ignore this structure leak document-specific patterns into the validation set, producing inflated performance estimates [33].

We corrected this by switching to StratifiedGroupKFold, grouping by document, with 5 folds [34, 35]. Table 7 shows the magnitude of the inflation on English data. The gap ranges from 15 to 37 percentage points depending on the classifier and relation. Tree-based models suffer the largest drops (ExtraTrees loses 37 pp on `at`), because they can memorize document-specific feature patterns when the same document's entity pairs appear in both training and validation folds. LogisticRegression, which cannot capture such document-level structure as easily, drops only 15 pp. This differential confirms that document-grouped evaluation is essential for this task and that the magnitude of inflation depends on the model family. Figure 2 visualizes the gap across four classifier/relation combinations.

5.3. Feature Ablation

We evaluated feature group contributions using HistGBM (500 iterations, learning rate 0.05) with balanced sample weights and document-grouped 5-fold CV. Table 8 shows the cumulative tier ablation, where each column adds one more feature group to the classification pipeline.

Group A alone (minimum character distance) captures most of the classification signal. Adding further groups provides inconsistent gains across languages and relations. For DE `at`, performance degrades from 0.408 with Group A alone to 0.385 with the full feature set. For FR `isAt`, it declines from 0.592 to 0.547, a drop of 4.5 percentage points. These results align with prior findings that distance features dominate in relation classification [19, 21] and that complex features provide marginal or negative returns [23]. The pattern also echoes the findings of de-Dios-Flores et al. [24], who reported that basic lemmatized features outperformed deep syntactic features for biographical person-place

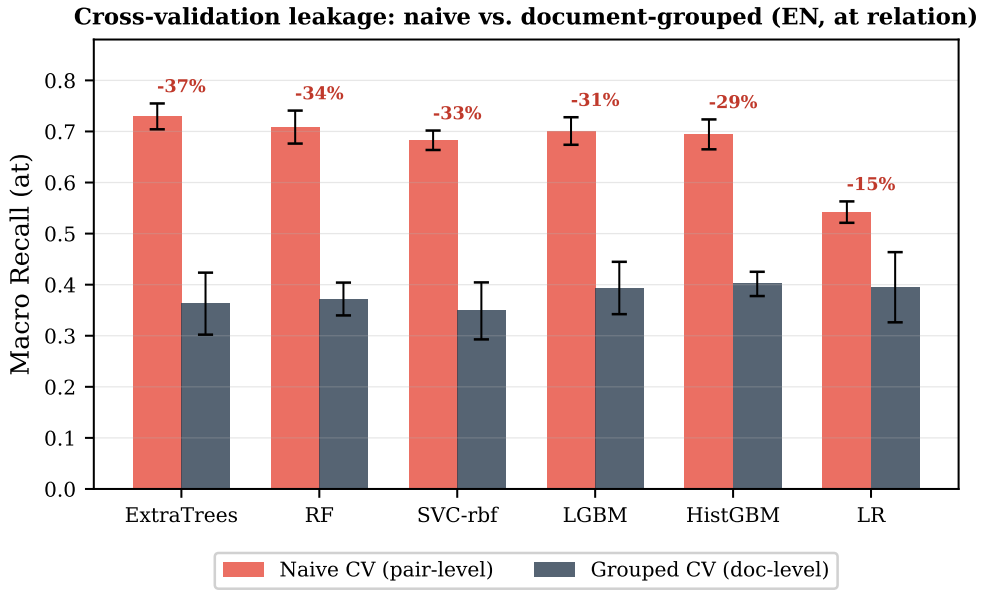

Figure 2: Naive pair-level CV versus document-grouped CV for four classifier/relation combinations on English data. The gap ranges from 15 to 37 percentage points.

Table 8
Cumulative feature ablation. Each column adds one more feature group. Values are mean macro recall ± standard deviation across 5 folds of document-grouped CV.

Lang	Rel	A	A+B	A+B+C	A+B+C+D	Full (A–E)
EN	`at`	.325 ± .042	.313 ± .047	.352 ± .043	.359 ± .062	.348 ± .059
EN	`isAt`	.471 ± .057	.553 ± .046	.523 ± .034	.546 ± .060	.542 ± .084
FR	`at`	.402 ± .013	.395 ± .017	.393 ± .010	.410 ± .009	.408 ± .014
FR	`isAt`	.592 ± .031	.586 ± .032	.557 ± .021	.552 ± .017	.547 ± .012
DE	`at`	.408 ± .050	.422 ± .059	.401 ± .069	.397 ± .059	.385 ± .053
DE	`isAt`	.530 ± .077	.557 ± .072	.513 ± .034	.511 ± .034	.504 ± .030

Table 9
Leave-one-group-out marginal contributions (percentage points). Positive values mean the group helped; negative values mean it was noise or harmful. Deltas smaller than ∼2 pp are within fold variance.

Lang	Rel	−A	−B	−C	−D	−E
EN	`at`	−0.4	−0.2	+2.8	−0.9	−1.2
EN	`isAt`	+2.2	+1.4	−0.8	+1.3	−0.4
FR	`at`	−0.4	+0.7	+0.8	+1.9	−0.1
FR	`isAt`	+0.1	+1.1	−1.2	+0.3	−0.5
DE	`at`	+0.1	−0.4	−1.8	−3.2	−1.2
DE	`isAt`	+0.5	+0.6	−0.5	−1.0	−0.7

extraction; our Group E verb-category features, which encode the closest analog to deep semantic knowledge in our pipeline, provide no consistent benefit.

To isolate the marginal contribution of each group, we also performed leave-one-group-out ablation. For each group, we trained the full pipeline with that group removed and computed the difference from the full model score. Table 9 reports these deltas in percentage points. Positive values indicate that the group helped, meaning that removing it dropped performance.

No single feature group helps uniformly across all six (language, relation) cells. Group A (minimum character distance) has the clearest positive signal for English `isAt` (+2.2 pp) but is within noise for most other cells. Group D (intermediary-routed distances) helps French `at` (+1.9 pp) but actively hurts

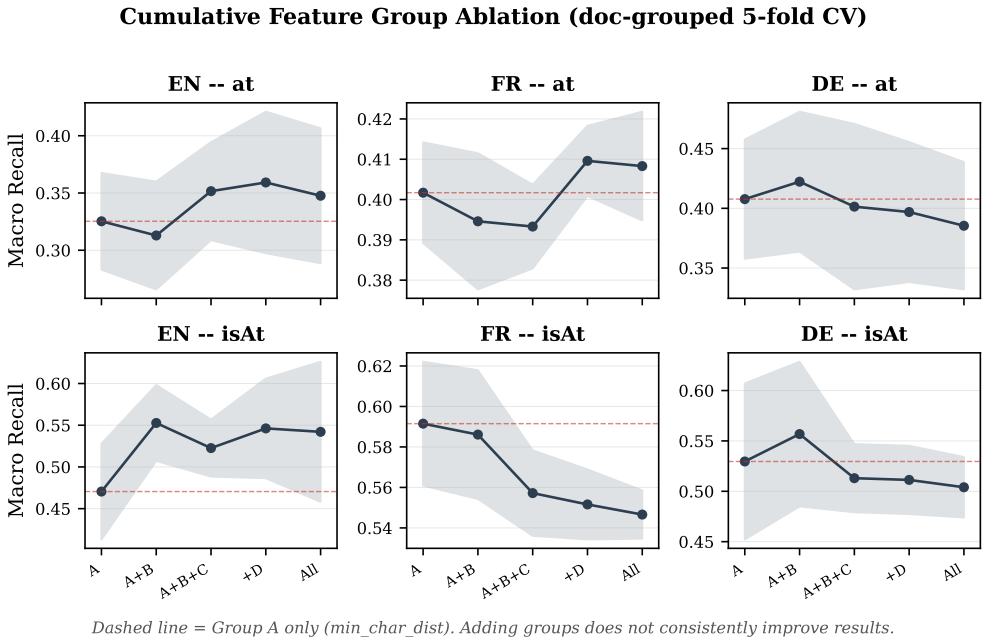

Figure 3: Cumulative feature ablation across languages and relations. Each line adds one feature group from left to right. Shaded bands show ±1 standard deviation across 5 folds.

German `at` (−3.2 pp), the largest negative delta in the table. Group E (verb-category features) provides no clear benefit anywhere, consistent with its reliance on a hand-curated lexicon that may not transfer well across languages. The DE `at` cell is the most striking example. Every group beyond A either has no effect or degrades performance, confirming the over-engineering pattern visible in the cumulative ablation (Figure 3).

RandomForest feature importance scores corroborate the ablation. For `at`, the top four features by importance are *l_position_norm* (0.151), *min_char_dist* (0.150), *p_position_norm* (0.124), and *sent_dist* (0.089); all four are proximity-based, and together they account for over 50% of total importance. For `isAt`, the same four features dominate, with *min_char_dist_via_pron* (0.051) entering the top five. The pronoun-bridge feature ranking is notable. It confirms that for `isAt`, the path from a person entity through a pronoun to a location carries discriminative signal, consistent with the observation that physical presence is often expressed through pronominal co-reference rather than direct mention. By contrast, features from Group E (verb-category distances) rank near the bottom for both relations, explaining why their removal in the leave-one-group-out analysis produces negligible or negative deltas.

5.4. Official Results

Our best submission, Run 1, reached a macro recall of 0.5142 on Test A, placing our team (DS@GT HIPE, team 2) 26th of the 46 ranked entries on the Accuracy profile and 3rd on the Efficiency profile; the 46 entries comprise 45 participant runs from 17 teams plus the organizers' baseline. We refer the reader to the overview papers for the full leaderboard [3, 4]. Table 10 presents the Accuracy profile results. All three runs score above the random baseline (0.4049) but below the Mistral-3B baseline (0.5818) [5]. The top-performing system achieved 0.7479.

Table 11 breaks down the accuracy results by language and relation. Two patterns emerge. First, `isAt` is consistently easier than `at` across all languages and runs, with scores 0.10–0.24 points higher. The binary `isAt` distinction (physically present or not) is more directly expressed by proximity features than the three-way `at` judgment (which requires abductive reasoning about whether a person has *ever* been at a location). Second, no single run dominates across all languages. Run 1 produces the best English `isAt` score (0.622) but its French `at` (0.397) trails Run 2's (0.415), and Run 3 is weakest

Table 10
Accuracy profile results on Test A (newspaper test set); official evaluation.

Run	Macro Recall	Rank (of 46)
Run 1	0.5142	26
Run 2	0.4836	30
Run 3	0.4771	31

Table 11
Per-language accuracy breakdown showing `at` and `isAt` macro recall components and per-language rank (of 46 runs). Columns EN/FR/DE are Test A; "Surp. FR" is Test B. Official evaluation; Surprise FR reuses the corresponding FR strategy.

	English			French			German			Surp. FR
Run	`at`	`isAt`	Rank	`at`	`isAt`	Rank	`at`	`isAt`	Rank	Avg
Run 1	.399	.622	27	.397	.635	26	.428	.604	28	.363
Run 2	.371	.510	40	.415	.619	25	.390	.598	33	.372
Run 3	.363	.604	30	.414	.471	37	.420	.590	30	.392

Table 12
Efficiency profile results on Test A (official evaluation). Ranks are among the 46 ranked entries. Parameter and size ranks reflect global totals reported to organizers rather than per-run values.

Run	Eff. Rank	Acc. Rank	Param Rank	Size Rank
Run 1	**3**	23	5	4
Run 2	4	27	5	4
Run 3	5	28	5	4

overall on newspaper data but produces the best surprise French score (0.392). The run ordering reversal on Test B suggests that the GAT EndpointMLP strategy used by Run 3 for French generalizes better to out-of-domain literary text than the sklearn strategies in Runs 1 and 2, consistent with the node-level FastText representation capturing some domain-transferable signal even without graph message passing.

Table 12 shows the efficiency profile. Run 1 ranked 3rd overall, with per-language efficiency ranks of 2nd in English and 5th in German; Run 2 ranked 2nd in French. The efficiency ranking combines accuracy rank, parameter count rank, and model size rank with equal weight. The parameter counts reported to the organizers used a global total (2,087,375 parameters, 87 MB) identical across all three runs rather than the per-run values in Table 6. This global figure, not the per-run counts, is what produced our official Efficiency ranks; we therefore present the per-run budgets of 507K to 847K parameters as a corrected re-analysis. Because the Efficiency profile is a relative ranking against other teams, we cannot recompute our exact rank from the corrected figures, but since our true per-run footprint is smaller than the submitted 2.09M total, our genuine efficiency is at least as good as the reported rank.

On Test B (the French literary surprise set), all three runs score below 0.40 (Table 11, last column), reflecting the domain shift from 19th–20th century newspapers to 18th century literary text. Run 3 (GAT EndpointMLP, 0.392) outperforms Run 1 (sklearn, 0.363), reversing the newspaper ordering. The EndpointMLP architecture operates on raw node feature vectors without graph message passing, which may reduce its dependence on newspaper-specific parse patterns. The absolute Test B scores nonetheless remain low for all runs, and the failure modes differ from the newspaper setting: the literary register uses longer, more subordinated sentences and archaic orthography that degrade the Stanza parses further, so a larger share of entity pairs fall in disconnected graph components, and the `at` label distribution shifts away from the newspaper prior the classifiers were tuned on. A dedicated domain-adaptation step, rather than reusing the French newspaper models unchanged, is the most

Table 13
Internal evaluation scorecard (newspaper held-out, 7 docs/lang). Mean macro recall across EN, FR, DE. Tiers reflect increasing methodological complexity.

Tier	Approach	Mean
1: Baseline	Heuristic	0.442
	Majority class	0.417
2: Rule	Path length tiered	0.549
	Same subtree	0.462
3: Tabular	sklearn pass_C (multilingual)	0.520
	sklearn pass_A (per-lang)	0.514
	sklearn pass_B (per-lang)	0.507
3: Graph	GAT 3-seed majority	0.519

promising remedy.

5.5. Internal Model Evaluation

Before submission, we evaluated all strategies on a newspaper-only held-out set (7 documents per language, with the remaining 27–28 documents used for training) using the official HIPE scorer. Table 13 summarizes results across four model tiers.

The most consequential finding from internal evaluation is that the path-length-tiered rule (Tier 2) achieves 0.549, outperforming all trained models on mean score. This rule classifies pairs as TRUE when the graph path between them is short and FALSE when long, using empirically tuned thresholds. That a deterministic rule with no trainable parameters beats both sklearn ensembles and GATs is consistent with the feature ablation finding. Minimum character distance carries most of the classification signal, and the path-length rule is essentially a graph-topology encoding of the same proximity principle. The 15 additional features and the learned decision boundaries of the trained models do not overcome the noise they introduce.

The trained models (Tier 3) cluster tightly between 0.507 and 0.520, with the GAT 3-seed majority vote (0.519) essentially matching the best sklearn pass (0.520). Per-language results diverge in a pattern consistent with the parse quality analysis (Section 5.8). The best English strategy is path_length_tiered (0.570, Tier 2), exploiting the fact that when English parses are noisy, a simple rule over the graph topology outperforms learned models. French and German are best served by sklearn pass_B (0.583) and sklearn pass_C (0.519) respectively, where cleaner parses and more training data allow learned models to surpass the rule. This per-language variation motivated our run composition strategy of selecting independently per language rather than using a single global model.

5.6. Efficiency Analysis

The classification step uses no pretrained language model. Stanza [29] dependency parsing is the only neural component in the inference pipeline, and its output serves as input to deterministic graph construction and feature extraction rather than providing learned representations for classification. FastText [8] embeddings are used only during graph construction for content-word bridging and do not enter the classification features.

Table 14 summarizes the computational profile of each pipeline component. Scikit-learn strategies train in under one second on a single CPU core. The GAT models have 335,247–511,503 parameters (5–8 MB each), making them small by current standards. Per-run parameter counts of 506,923–846,701 are two to three orders of magnitude smaller than the transformer-based systems that dominate the top of the accuracy leaderboard, where parameter counts range from hundreds of millions to over 100 billion.

Table 14
Computational profile by pipeline stage. All timings measured on a single machine (CPU only for sklearn; GPU optional for GAT).

Component	Parameters	Train time
Stanza parsing	(pretrained)	n/a (inference only)
Graph construction	0	<1 s per doc
Feature extraction	0	<1 s per doc
sklearn classifiers	<1K	0.2–0.6 s total
GAT architectures	335K–512K	35–599 s (5-fold CV)

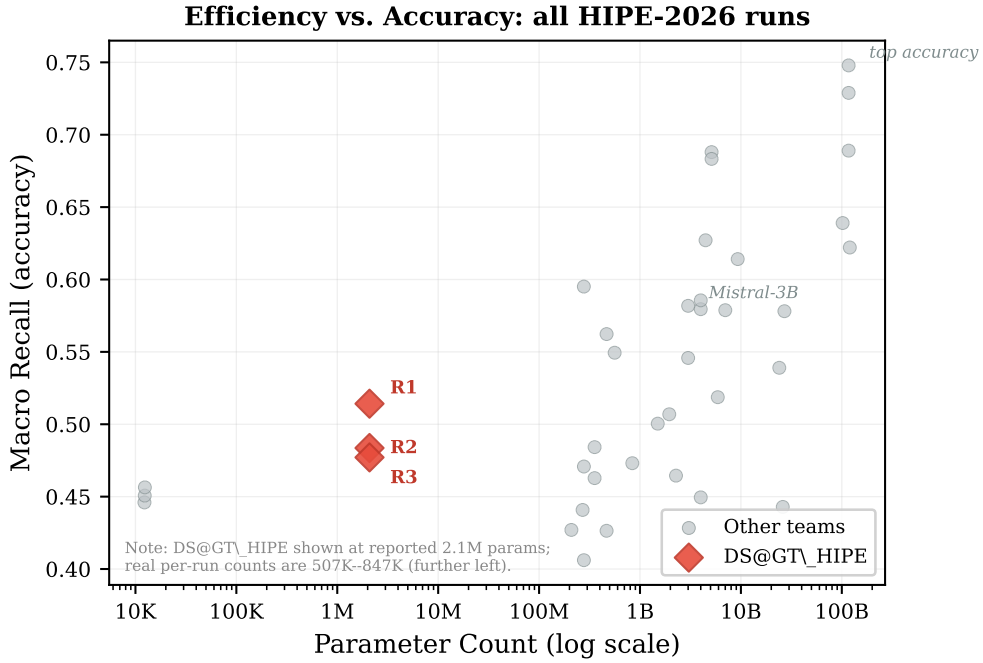

Figure 4: Accuracy versus efficiency rank for the three submitted runs (filled markers) and reference systems. Our runs cluster in the high-efficiency, mid-accuracy region of the plot.

The HIPE-2026 Efficiency profile ranks runs by the arithmetic mean of three ranks (accuracy rank, model parameter-count rank, and deployed model-size rank), so a lower efficiency score is better [3]. For run r the efficiency score is $\frac{1}{3}\left(\mathrm{rank}_{\mathrm{acc}}(r) + \mathrm{rank}_{\mathrm{params}}(r) + \mathrm{rank}_{\mathrm{size}}(r)\right)$. Run 1 placed 3rd overall despite ranking only 26th on accuracy, because its small parameter budget and model size compensated. Run 1 ranked 2nd in English and 5th in German; Run 2 ranked 2nd in French.

Figure 4 plots macro recall against efficiency rank for the submitted runs, illustrating the accuracy–efficiency trade-off. The accuracy gap to the top system (0.7479 versus our best of 0.5142) is 23 percentage points. However, the top systems rely on large language models with billions of parameters and substantially higher inference costs. Our system demonstrates that engineered features over dependency parses can reach mid-table accuracy at a fraction of the computational cost, which is the central trade-off this work explores. For historical newspaper processing at scale, where millions of entity pairs must be classified, this cost difference is substantial [6, 7].

5.7. Error Analysis

To characterize the failure modes of our system, we examined per-class performance on English held-out data (17 documents, 151 pairs) using the pipeline that produced the submitted runs. Table 15 reports precision and recall by class for both relations.

The classifier identifies FALSE accurately for both relations but performs poorly on TRUE. For `at`,

Table 15
Per-class precision and recall for `at` and `isAt` on English held-out data (17 documents, 151 entity pairs).

Relation	Class	Support	Precision	Recall
`at`	FALSE	68	0.56	0.78
`at`	PROBABLE	54	0.47	0.39
`at`	TRUE	29	0.42	0.17
`isAt`	FALSE	133	0.88	0.98
`isAt`	TRUE	18	0.25	0.06

Table 16
Parse quality metrics per language, aggregated across all datasets. Higher values indicate worse parse quality.

Language	No-path rate	Orphan PER	Orphan LOC
English	0.320	0.193	0.166
German	0.278	0.182	0.130
French	0.219	0.125	0.109

only 5 of 29 gold TRUE pairs are correctly identified (recall 0.17); for `isAt`, only 1 of 18 (recall 0.06). The confusion between PROBABLE and TRUE for `at` is the largest source of error, suggesting that the proximity features that separate FALSE from non-FALSE do not discriminate well between degrees of certainty. The extreme class imbalance in `isAt` (133 FALSE versus 18 TRUE on held-out) compounds this problem, as the model learns a strong FALSE prior that it rarely overrides.

Analysis of individual false negatives reveals two dominant failure patterns. First, OCR-broken entity surface forms prevent correct entity-to-head-word mapping. For example, in document `sn89058133-1920-07-08`, the person mention "Frank\nShirley" (with a line-break mid-name) and the location "Cookeville" are gold TRUE, but the system predicts PROBABLE because the broken mention disrupts the dependency parse path. Similarly, "Miss Williams" paired with "Shelby coun\nty" (split location) is misclassified for the same reason. Second, cross-paragraph entity pairs with large character distances are systematically missed. "OTTO" paired with "Hereford street" at a character distance of 765 has no graph path between the entities, forcing the classifier to rely on a single noisy feature. Of the 152 entity pairs with no graph path in the English training data, 93% are due to graph fragmentation (the person and location are correctly tagged but fall in different connected components) rather than entity-tagging failure, and most of these disconnected pairs are structurally cross-paragraph with no shared topical content.

False positives exhibit a complementary pattern. In short articles containing few entities, the model treats proximity as evidence of a TRUE relation. For instance, "Moses" (a religious reference, not a person at a location) paired with "Nashville, Tenn." is predicted TRUE because the character distance is small and the article is brief. "Mr. Pain Wareing" paired with "D. C." is similarly a proximity overfire in a short article where the person and location co-occur without a spatial relation. These cases suggest that the system conflates textual proximity with semantic relatedness, a limitation inherent to distance-based features that Giuliano et al. [25] also identified in biomedical RE. Shallow features are resilient to parser noise but cannot distinguish co-occurrence from genuine relation.

5.8. Parse Quality and the GAT–Sklearn Gap

The per-language gap between GAT and sklearn performance varies substantially. On internal evaluation, the GAT provides 2.8 percentage points of lift over the best sklearn pass on English but offers no improvement on French. We audited parse quality across languages to investigate this variation. Table 16 reports the no-path rate (fraction of entity pairs with no graph path) and orphan entity rates (fraction of person or location nodes with degree zero in the subgraph) per language.

French has the highest parse quality, with the lowest no-path rate (22%), the densest subgraphs, and

the lowest orphan rates. English has the worst, with roughly 1 in 3 entity pairs having no graph path and 1 in 5 person mentions having degree zero in the subgraph. German is intermediate.

These differences explain the GAT–sklearn pattern through two compounding effects. When parse quality is poor and training data is small (English, 307 pairs), the 15 scalar features extracted from noisy edges are themselves noisy, and the sklearn classifiers do not have enough training examples to denoise them. The GAT, operating on the same noisy graph, integrates evidence from per-node embeddings and multi-hop attention in a way that the 15-scalar projection cannot. When parse quality is good and training data is larger (French, 478 pairs), the scalar features are cleaner, and sklearn has enough examples to learn effective decision boundaries. The graph's marginal benefit shrinks, and tabular models close the gap. The submitted GAT architectures are consistent with this interpretation. For French, whose parses are cleanest, the newspaper held-out sweep selected the non-graph EndpointMLP (Table 17), indicating that when the parse is reliable the node features alone suffice; for English, whose parses are noisiest, the graph-based PairGATStructuralSkip was selected; and for the intermediate-quality German parses PairGAT was selected. Graph message passing thus appears to help more as parse quality degrades.

6. Conclusion

We presented a lightweight pipeline for person–place relation extraction from multilingual historical newspapers, combining dependency-parse graphs, 15 engineered proximity features, and small classifiers with per-run parameters under 847K.

The central finding is that minimum character distance dominates the classification signal. Feature ablation showed that adding groups beyond Group A provides inconsistent gains and sometimes degrades performance. A deterministic path-length rule outperformed all trained models on internal evaluation (0.549 versus 0.520 for the best sklearn pass), and the GAT branch offered no meaningful improvement over tabular classifiers. These results align with prior work on proximity dominance in RE [19, 21, 23] and suggest that the task partly rewards surface proximity rather than semantic understanding.

Error analysis revealed complementary failure modes. The system misses TRUE relations at distance (93% of no-path pairs are due to graph fragmentation, not tagging failure) and hallucinates them at proximity (short articles where co-occurrence does not imply a spatial relation). Parse quality varies across languages and explains the GAT–sklearn gap. English (no-path rate 32%, orphan PER 19%) benefits from the GAT's richer representation, while French (no-path 22%, orphan PER 13%) allows tabular models to match the graph-based approach. Document-grouped cross-validation proved essential, revealing 25–37 percentage point inflation over naive splits [34, 33].

Several directions remain open. New bridge types, for example coreference-based or discourse-based links, could extend coverage beyond the current 69% of reachable pairs, directly targeting the cross-paragraph failures identified in the error analysis. Reformulating `at` as a binary cascade (first FALSE versus non-FALSE, then TRUE versus PROBABLE) could address the confusion between TRUE and PROBABLE that dominates our `at` errors. A dedicated domain-adaptation path for Test B, in place of reusing the French newspaper models unchanged, would likely recover much of the accuracy lost to register shift. Class-imbalance handling for the rare TRUE class, and calibrated thresholds tuned per language, are natural extensions given the strong FALSE prior on `isAt`. Finally, evaluation on larger corpora would test whether the lightweight, feature-driven approach continues to hold its efficiency advantage as training data grows.

A. Supplementary Results

This appendix provides detailed results that support the analyses in the main text.

A.1. GAT Architecture Sweep

Table 17 reports the full architecture sweep across all four GAT variants on newspaper held-out data at seed 1. These results informed the per-language architecture selection summarized in Table 4.

Table 17
GAT architecture sweep on newspaper held-out data (seed 1). Global is the mean of `at` and `isAt` macro recall. Bold marks the per-language winner.

Language	Architecture	at	isAt	Global
EN	PairGAT	0.378	0.542	0.460
EN	PairGATBranched	0.356	0.583	0.469
EN	**PairGATStructuralSkip**	**0.459**	**0.625**	**0.542**
EN	EndpointMLP	0.333	0.583	0.458
FR	PairGAT	0.383	0.689	0.536
FR	PairGATBranched	0.333	0.595	0.464
FR	PairGATStructuralSkip	0.307	0.662	0.485
FR	**EndpointMLP**	**0.388**	**0.694**	**0.541**
DE	**PairGAT**	**0.411**	**0.629**	**0.520**
DE	PairGATBranched	0.389	0.564	0.477
DE	PairGATStructuralSkip	0.371	0.543	0.457
DE	EndpointMLP	0.268	0.447	0.357

A.2. Sklearn Pass Comparison on Held-Out Data

Table 18 compares the three sklearn pass designs on newspaper held-out data (7 documents per language). Pass A trains a fixed classifier pair (RandomForest for `at`, ExtraTrees for `isAt`) per language. Pass B selects the per-language CV winner for each relation independently. Pass C trains on multilingual-concatenated data from all three languages.

Table 18
Sklearn held-out pass comparison (7 docs/lang). Bold marks the best per-language global score. Mean global is the average across all three languages.

	English			French			German			Mean
Pass	at	isAt	Global	at	isAt	Global	at	isAt	Global	Global
A	0.421	0.625	**0.523**	0.383	0.732	0.558	0.331	0.593	0.462	0.514
B	0.495	0.487	0.491	0.406	0.759	**0.583**	0.331	0.564	0.448	0.507
C	0.385	0.583	0.484	0.395	0.716	0.555	0.403	0.636	**0.519**	**0.520**

A.3. Feature Importance Rankings

Table 19 reports RandomForest feature importance scores (Gini impurity-based) for the top-ranked features on English newspaper data. The four highest-ranked features for both relations are proximity-based, corroborating the ablation analysis in Section 5.3.

Table 19
Top 5 RandomForest feature importance scores for `at` and `isAt` on English newspaper data.

Rank	Feature	Importance
at		
1	l_position_norm	0.151
2	min_char_dist	0.150
3	p_position_norm	0.124
4	sent_dist	0.089
5	l_location_align	0.088
isAt		
1	l_position_norm	0.177
2	min_char_dist	0.162
3	p_position_norm	0.119
4	sent_dist	0.112
5	min_char_dist_via_pron	0.051

Declaration on Generative AI

During the preparation of this work, the author used generative AI tools for grammar and spelling checking, for aggregating and formatting outputs generated by the author's own analysis scripts, and for code generation, correction, and review. All research ideas, concepts, methods, experimental design, and interpretations are the author's own. After using these tools, the author reviewed and edited all content as needed and takes full responsibility for the publication's content.

References

[1] M. Ehrmann, M. Romanello, A. Flückiger, S. Clematide, Extended Overview of CLEF HIPE 2020: Named Entity Processing on Historical Newspapers, in: L. Cappellato, C. Eickhoff, N. Ferro, A. Névéol (Eds.), Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, volume 2696, CEUR-WS, Thessaloniki, Greece, 2020, p. 38. URL: https://infoscience.epfl.ch/record/281054.

[2] M. Ehrmann, M. Romanello, S. Najem-Meyer, A. Doucet, S. Clematide, Extended overview of HIPE-2022: Named Entity Recognition and Linking in Multilingual Historical Documents, in: G. Faggioli, N. Ferro, A. Hanbury, M. Potthast (Eds.), Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, volume 3180, CEUR-WS, 2022. URL: http://ceur-ws.org/Vol-3180/paper-83.pdf. doi:10.5281/zenodo.6979577.

[3] 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.

[4] 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.

[5] J. Opitz, S. Burst, A closer look at classification evaluation metrics and a critical reflection of common evaluation practice, Transactions of the Association for Computational Linguistics (TACL) 12 (2024).

[6] R. Schwartz, J. Dodge, N. A. Smith, O. Etzioni, Green AI, Communications of the ACM 63 (2020) 54–63.

[7] E. Strubell, A. Ganesh, A. McCallum, Energy and policy considerations for deep learning in NLP, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL), 2019, pp. 3645–3650.

[8] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Transactions of the Association for Computational Linguistics 5 (2017) 135–146.

[9] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[10] M. Fey, J. E. Lenssen, Fast graph representation learning with PyTorch Geometric, in: ICLR Workshop on Representation Learning on Graphs and Manifolds, 2019.

[11] M. Ehrmann, A. Hamdi, E. Linhares Pontes, M. Romanello, A. Doucet, Named entity recognition and classification in historical documents: A survey, ACM Computing Surveys 56 (2023) 27.

[12] A. Hamdi, E. Linhares Pontes, N. Sidère, M. Coustaty, A. Doucet, In-depth analysis of the impact of OCR errors on named entity recognition and linking, Natural Language Engineering 29 (2023) 1–24.

[13] R. Bunescu, R. Mooney, A shortest path dependency kernel for relation extraction, in: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP), 2005, pp. 724–731.
[14] Y. Zhang, P. Qi, C. D. Manning, Graph convolution over pruned dependency trees improves relation extraction, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2018, pp. 2205–2215.
[15] Z. Guo, Y. Zhang, W. Lu, Attention guided graph convolutional networks for relation extraction, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL), 2019, pp. 241–251.
[16] G. Nan, Z. Guo, I. Sekulić, W. Lu, Reasoning with latent structure refinement for document-level relation extraction, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL), 2020, pp. 1546–1557.
[17] N. Peng, H. Poon, C. Quirk, K. Toutanova, W.-t. Yih, Cross-sentence n-ary relation extraction with graph LSTMs, Transactions of the Association for Computational Linguistics 5 (2017) 101–115.
[18] Y. Yao, D. Ye, P. Li, X. Han, Y. Lin, Z. Liu, Z. Liu, L. Huang, J. Zhou, M. Sun, DocRED: A large-scale document-level relation extraction dataset, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL), 2019, pp. 764–777.
[19] J. Jiang, C. Zhai, A systematic exploration of the feature space for relation extraction, in: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT), 2007, pp. 113–120.
[20] G. Zhou, J. Su, J. Zhang, M. Zhang, Exploring various knowledge in relation extraction, in: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL), 2005, pp. 427–434.
[21] D. Zeng, K. Liu, S. Lai, G. Zhou, J. Zhao, Relation classification via convolutional deep neural network, in: Proceedings of the 25th International Conference on Computational Linguistics (COLING), 2014, pp. 2335–2344.
[22] Y. Zhang, V. Zhong, D. Chen, G. Angeli, C. D. Manning, Position-aware attention and supervised data improve slot filling, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2017, pp. 35–45.
[23] C. Alt, A. Gabryszak, L. Hennig, Probing linguistic features of sentence-level representations in neural relation extraction, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL), 2020, pp. 1547–1559.
[24] I. de-Dios-Flores, J. R. Pichel, F. Silva, Exploring the effectiveness of linguistic knowledge for biographical relation extraction, Natural Language Engineering (2015).
[25] C. Giuliano, A. Lavelli, L. Romano, Exploiting shallow linguistic information for relation extraction from biomedical literature, in: Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL), 2006, pp. 401–408.
[26] A.-L. Minard, A.-L. Ligozat, B. Grau, Multi-class SVM for relation extraction from clinical reports, in: Proceedings of Recent Advances in Natural Language Processing (RANLP), 2011, pp. 604–609.
[27] T. Renslow, G. Neumann, LightRel SemEval-2018 task 7: Lightweight and fast relation classification, in: Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018), 2018, pp. 778–782.
[28] A. Plum, Biographical Information Extraction: A Language-Agnostic Methodology for Datasets and Models, Ph.D. thesis, University of Wolverhampton, 2022.
[29] P. Qi, Y. Zhang, Y. Zhang, J. Bolton, C. D. Manning, Stanza: A Python natural language processing toolkit for many human languages, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 2020, pp. 101–108.
[30] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph attention networks, in: International Conference on Learning Representations (ICLR), 2018.
[31] S. Brody, U. Alon, E. Yahav, How attentive are graph attention networks?, in: International Conference on Learning Representations (ICLR), 2022.
[32] A. Mandya, D. Bollegala, F. Coenen, Graph convolution over multiple dependency sub-graphs

for relation extraction, in: Proceedings of the 28th International Conference on Computational Linguistics (COLING), 2020, pp. 6424–6435.

[33] A. Elangovan, J. He, K. Verspoor, Memorization vs. generalization: Quantifying data leakage in NLP performance evaluation, arXiv preprint arXiv:2102.01818 (2021).

[34] D. R. Roberts, V. Bahn, S. Ciuti, M. S. Boyce, J. Elith, G. Guillera-Arroita, S. Hauenstein, J. J. Lahoz-Monfort, B. Schröder, W. Thuiller, D. I. Warton, B. A. Wintle, F. Hartig, C. F. Dormann, Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure, Ecography 40 (2017) 913–929.

[35] A. Søgaard, S. Ebert, J. Bastings, K. Filippova, We need to talk about random splits, in: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL), 2021, pp. 1823–1832.