Some LLMs Are Smaller Than Others: A Lightweight Linear Model for Relation Extraction Applied to Historical Documents

HIPE at CLEF 2026

Thomas Checchin[1,*,†], Adrien Guille[1,*,†] and Nicolas Gutehrlé[1,*,†]

[1]Université Lumière Lyon 2, Université Claude Bernard Lyon 1, ERIC, 69007, Lyon, France

Abstract

This paper presents the results of the ROSTI team for the HIPE-2026 shared task. Our aim was to propose the most frugal approach possible that could achieve Accuracy scores as close as possible to the baseline system. Thus, our system relies on Logistic Regression models, and determines the *at* or *isAt* relation between a pair of entities (*person, location*) based on pairs of sentences where these entities occur. The sentences, as well as the entities mentions occurring in these sentences, are encoded as TF-IDF weights. When predicting the *at* or *isAt* relation between a pair of entities, we aggregate the predictions made by the corresponding model to the pairs of sentences where these entities occur, and select the most predicted label. Our approach achieves low to average scores in the Accuracy profile that are lower than the baseline, but achieves better scores in the Efficency profile, ranking 6 overall and notably ranking first on the German dataset. This shows the validity of our approach, which is significantly smaller and more frugal than the baseline or other proposed approaches, while being independent from any other external resources or other models.

Keywords

Historical documents, Relation Classification, Logistic Regression, Frugal

1. Introduction

The HIPE-2026 task focuses on the extraction of person-place relation from noisy historical texts in English, French and German[1]. Given a document and a pair of entities, the systems are asked to determine two types of relation: whether the person has ever been to this place (*at* relation), and whether the person was located at this place around the publication time of the document (*isAt* relation). Systems are evaluated according to three profiles: their accuracy, their computational efficiency and their generalization capabilities [1]. 17 teams participated, amounting to a total of 45 runs [2, 3].

HIPE-2026 fits within the Relation Extraction (RE) task, which aims at determining the relation existing between multiple entities (often two), either at the sentence-level or at the document level. This task is usually performed after performing Named Entity Recognition (NER), although alternative approaches propose to perform NER and RE tasks in a joint manner, thus avoiding the risk of propagating errors from NER results to the RE task. Most available datasets, such as TACRED [4] or DocRED [5], focus on contemporary English, although recent works have focused on the processing of historical documents. RE allows to produce knowledge graphs that can assist in exploring dataset or downstream tasks such as generating documents or biographies. For a more complete description of the RE tasks, especially applied to historical documents, see [1, 6].

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

*Corresponding author.

†These authors contributed equally.

thomas.checchin@univ-lyon2.fr (T. Checchin); adrien.guille@univ-lyon2.fr (A. Guille); nicolas.gutehrle@univ-lyon2.fr (N. Gutehrlé)

https://cv.hal.science/thomas-checchin (T. Checchin); https://eric.univ-lyon2.fr/aguille/ (A. Guille); https://nicolasgutehrle.github.io (N. Gutehrlé)

0009-0007-6788-8891 (T. Checchin); 0000-0002-1274-6040 (A. Guille); 0009-0000-1958-215X (N. Gutehrlé)

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

In this paper, we present the results of the ROSTI team in this shared task. Our aim while approaching it was to propose the most frugal possible approach that could achieve Accuracy scores as close as possible to the baseline system. Thus, our approach relies on Logistic Regression models, which determine the relation between a pair of entities, whether *at* or *isAt*, based on pairs of sentences where these entities occur. Our code is available on GitHub[2].

The rest of this paper is organized as follows: in Section 2, we first describe our pipeline for generating a synthetic dataset from the HIPE-2026 dataset, before presenting our approach. We discuss our results in Section 3, and finally propose our conclusion, as well as avenues for future works, in Section 4.

2. Method

We propose a frugal approach based on Logistic Regression models that determines the *at* or *isAt* relation between a pair of entities (*person, location*) according to sentences where these entities occur. Our approach solely relies on the textual content of the documents, and do not exploit other metadata such as the publication date of the document, nor external resources or other preprocessing models.

2.1. Dataset

Hipe-2026 provides two training datasets: a silver dataset (Silver Train A and Silver Dev A), automatically labeled with an LLM, and a manually annotated gold dataset (Gold Train A) . The former contains 617 documents and 16,502 relations, whereas the latter includes 104 documents for 2,502 relations. For a more detailed description of those datasets, see [3]. To obtain a dataset suited for our approach, we apply the following pipeline to produce a synthetic dataset in a weak supervision manner from the provided HIPE data:

First, we divide the documents into sentences using the NLTK Python package [7]. Then, for all person and location mentions of an entity pair *(person, location)*, we create a sextuple *(s1, s2, e1, e2, at, isAt)*, where:

- *e1* is one of the mentions corresponding to the first entity, i.e. the person
- *e2* is one of the mentions corresponding to the second entity, i.e. the location
- *s1* and *s2* are sentences where *e1* and *e2* each appear in at least one of them. *s1* always precedes *s2* in the document.
- *at* and *isAt* are the *at* relation (either *TRUE*, *PROBABLE* or *FALSE*) and *isAt* (either *TRUE* or *FALSE*) relation for this *(person, location)* pair in the HIPE dataset

An example of sextuples generated for a specific entity pair (*person, location*) is shown in Table 1.

Table 1
Sextuples synthetically generated for the entity pair (["Venard", "Yenard"], ["Nevada, Col."]). The occurrence of the entity mention in each sentence is highlighted in bold. Sentences are shortened for space reasons.

Sentence 1	Sentence 2	Entity 1	Entity 2	at	isAt
[...] coming to **Nevada, Col.**	**Venard** leveled [...]	Venard	Nevada, Col.	PROBABLE	FALSE
[...] coming to **Nevada, Col.**	**Venard** covered [...]	Venard	Nevada, Col.	PROBABLE	FALSE
[...] coming to **Nevada, Col.**	great rock **Yenard** [...].	Yenard	Nevada, Col.	PROBABLE	FALSE
[...] coming to **Nevada, Col.**	be pointing at **Yenard**.	[...]Yenard	Nevada, Col.	PROBABLE	FALSE

Using this pipeline, we created three synthetic datasets:

- *dataset_1* which corresponds to the entirety of provided silver dataset (Silver Train A and Silver Dev A)
- *dataset_2* which combines the entirety of the silver dataset with half of the provided train set
- *dataset_3* which combines the entirety of the silver dataset with the entirety of the train set

The statistical description and the distribution of classes in these datasets are shown in Table 2 and 3.

[2]https://github.com/ThomCh1/hipe-2026-rosti

Table 2
Description of the produced synthetic datasets

	dataset_1			dataset_2			dataset_3		
	en	de	fr	en	de	fr	en	de	fr
Unique sentences	695	3,655	8,587	695	3,774	8,651	705	3,902	8,794
Unique mentions of person	207	698	2,783	207	719	2161	210	731	2,834
Unique mentions of location	249	667	1,784	249	679	1,419	253	707	1,806
Unique pair mentions	678	2,617	7,969	678	2,697	6,150	688	2,784	8,179
Sextuples	2,787	31,344	89,255	3,788	37,542	95,318	4,332	40,584	97,830

Table 3
Distribution of classes in the synthetic datasets *dataset_1*, *dataset_2* and *dataset_3*

		en		de		fr	
		at	isAt	at	isAt	at	isAt
dataset_1	FALSE	836	2,228	10,546	24,326	40,663	74,737
	PROBABLE	1,212	0	10,693	0	26,922	0
	TRUE	739	559	10,105	7,018	21,670	14,518
dataset_2	FALSE	1,208	2,966	12,561	28,774	43,218	78,988
	PROBABLE	1,280	0	11,634	0	26,980	0
	TRUE	1,300	822	13,347	8,768	19,286	25,120
dataset_3	FALSE	1,339	3,325	13,718	31,516	43,710	79,985
	PROBABLE	1,393	0	11,991	0	27,324	0
	TRUE	1,600	1,007	14,875	9,068	26,796	17,845

2.2. Implementation

For each language, we train two models, one for the *at* label, and another for the *isAt* label. The model is trained to predict a label for a given a quadruple *(s1, s2, e1, e2)* , which is encoded as follows:

The sentences *s1* and *s2* are each encoded in a bag-of-word fashion, with a vocabulary of 5,000 unigrams and bigrams, using *tf·idf* weighting. The 5,000 cap has been selected after experimenting with sizes of 1,000, 5,000 and 10,000, as it provided the best balance between accuracy and number of parameters. The mentions of entities *e1* and *e2* that occur in the sentences *s1* and *s2* are encoded as two one-hot vectors, respectively of size P and L, where P is the total number of unique mentions of people, and L is the total number of unique mentions of locations found in the dataset.

The *tf·idf* statistics for each vector are computed independently within its respective set, i.e., *s1* weights from all *s1* sentences, *s2* weights from all *s2* sentences, *e1* weights from all *e1* mentions and *e2* weights from all *e2* mentions. Finally, the vectors are concatenated into one input vector I, as follows:

$$I = [s1, s2, e1, e2] \in \mathbb{R}^{5000+5000+P+L}$$

When predicting the *at* or *isAt* relation between a pair of entities (*person, location*) in a given language, we use the corresponding model to predict the relation for every quadruple *(s1, s2, e1, e2)* corresponding to that entity pair. Then, we aggregate the predictions made by the model, and select the label most predicted. In case of a tie, the first label that appears is selected.

For this task, we performed three runs, where models were trained on the various synthetic datasets: thus, the models in *run_1* were trained on *dataset_1*, the models in *run_2* were trained on *dataset_2*, and the models in *run_3* were trained on *dataset_3*. In total, we trained six models for one run, amounting to 18 models. For this experiment, we use the implementation of the Logistic Regression model provided by the *scikit-learn* library [8]. We train our models by setting the `class_weights` hyperparameter to *balanced*, and the `warm_start` hyperparameter to *True*.

Since the size of our vocabulary is much larger than the number of samples in our dataset, we

performed a grid search on *dataset_1* to search for the best inverse regularization strength values for each language and for each task. We tried values ranging from 0.2 to 5.8, with a step of 0.2. The best values found for this hyperparameter for each run, language and task are show in Table 4, whereas the number of parameters and size in bytes of each model is shown in Table 5. As Table 5 highlights, the vectorizer accounts for the majority of the pipeline's memory space, due to the storage of the vocabulary and the inverted vocabulary. The final models themselves are barely over 100 KB.

Table 4
Best values of regularization strength found for each model on each language and each task

	English		French		German	
	at	isAT	at	isAT	at	isAT
run_1	3.4	3.4	3.4	3.4	3.4	3.4
run_2	5.6	5.4	5.4	5.6	3.4	5.8
run_3	1.	1.	4.8	4.8	2.2	2.2

Table 5
Number of parameters and size (in bytes) of Logistic Regression models (including the vectorizer) trained with best combinations of hyperparameters in each run and for each language. The number of parameters and size are identical for both classification tasks (*at* and *isAt*).

	English		French		German	
	Parameters	Size	Parameters	Size	Parameters	Size
run_1	10,855	0.690MB	16,474	1MB	12,279	0.800MB
run_2	10,855	0.691MB	16,550	1MB	12,365	0.807MB
run_3	10,878	0.691MB	16,605	1.1MB	12,399	0.809MB

3. Results and Discussion

3.1. Evaluation Protocol

The task was evaluated according to three profiles, Accuracy, Generalization and Efficiency : in the Accuracy profile, a model is evaluated based on its performances on the *at* and *isAt* tasks, which are measured in terms of macro Recall score on the French, German and English. A score called Impresso Profile Scores (IPS) is computed as the mean of the macro Recall scores on both tasks for each language.

In the Generalization profile, a Surprise Profile Score (SPS) is calculated as the macro Recall score on the *at* label. This profile is only evaluated on French documents, and only on the *at* task.

In the Efficiency profile, a model is given two ranks, one based on its number of parameters and another based on its size in bytes. A score called Mean Efficiency Profile Rank (MEPR) is calculated as the mean of its parameter, size and accuracy rankings. This score is not calculated on the Generalization dataset. For a more detailed presentation of the evaluation protocols, see [1, 2, 3].

3.2. Results

The scores and ranks obtained by our approach in the Accuracy, Generalization and Efficiency profiles are shown in Table 6, and compared to the scores obtained by the baseline and best systems. The scores shown are the best obtained by each system, as calculated by the organisers[3].

In the Accuracy profile, our approach achieves low to average scores: it achieves Impresso Profile Scores of 0.454, 0.393 and 0.522 on the English, French and German test sets respectively, amounting

[3]The scores are available at https://hipe-eval.github.io/HIPE-2026/results and at https://hipe-eval.github.io/HIPE-2026/HIPE_2026_evaluation_results/

Table 6
Overall and detailed Accuracy (Acc.), Generalization (Gen.) and Efficiency (Eff.) rankings for the best, baseline and ROSTI teams. The Accuracy scores are shown as best Impresso Profile Scores (IPS), Generalization scores are shown as best Surprise Profile Scores (SPS), and Efficiency scores are shown as best Mean Efficiency Profile Rank (MEPR), as calculated by the organisers. Best scores are highlighted in bold. The best team may differ between datasets and profiles.

		English		French		German		Overall	
	System	Score	Rank	Score	Rank	Score	Rank	Score	Rank
Acc. (IPS)	Best	**0.749**	**1** (team8-run1)	**0.755**	1 (team13-run1)	**0.771**	**1** (team13-run3)	**0.747**	**1** (team13-run1)
	ROSTI	0.454	37 (run3)	0.393	44 (run3)	0.522	23 (run3)	0.456	37 (run3)
	baseline	0.555	22 (run1)	0.634	13 (run1)	0.555	19 (run1)	0.581	17 (run1)
Gen. (SPS)	Best			**0.816**	**1** (team8-run3)				
	ROSTI			0.384	32 (run3)				
	baseline			0.506	25 (run1)				
Eff. (MEPR)	Best	**10.333**	**1** (team14-run3)	**9**	**1** (team15-run2)			**9.666**	**1** (team14-run3)
	ROSTI	14	5 (run3)	15.666	9 (run1)	**8.666**	**1** (run1)	13.666	6 (run1)
	baseline	18	14 (run1)	14.666	8 (run1)	16.333	12 (run1)	15.666	9 (run1)

to a mean 0.456 IPS and ranking 37 overall. Our approach achieves lower scores than the baseline for every language, although the difference is small on the German test set (0.555 IPS for the baseline and 0.5222 for our approach). Similarly, in the Generalization profile, it achieves a 0.384 SPS, ranking 32.

However, our approach achieves better scores in the Efficiency profile, as it achieves an MEPR score of 8.666 on German, thus ranking first. Furthermore, our approach achieves a score of 14 and ranks 5 in English, a score of 15.666 and ranks 9 in French, and a score of 13.666 and ranks 6 overall. On the German dataset, it achieves similar Impresso Profile Scores in the Accuracy profile than the baseline, while being significantly smaller and more frugal, as shown in Table 7. Moreover, it is independent from any other external resources or other models.

Table 7
Number of parameters (Param.) and size (in MB) of the best models in each language and overall for the top, baseline and ROSTI teams.

	English		French		German		Overall (mean)	
System	Param.	Size	Param.	Size	Param.	Size	Param.	Size
Best team	**277,730,309**	**1,111**	**0**	**0**			**277,730,309**	**1,111**
ROSTI	12,399	0.81	12,279	0.8	**12,279**	**0.8**	12,279	0.8
baseline	3B	2,147.023	3B	2,147.023	3B	2,147.023	3B	2,147.023

3.3. Discussion

Confusion matrices on each language and on each task are shown in Table 8. The *Generalization* column in the *isAt* row in this table is empty, as the evaluation is only done on the *at* task. As the Table denotes, the models tend to predict the class *FALSE* more than the others, despite being trained with balanced weights. This behaviour was expected, due to the unbalanced nature of the provided data.

Most incorrect predictions are caused by a lack of explicit lexical evidence in the sentences that could lead to a strong *TRUE* prediction. This issue is furthermore reinforced by the size of the sentences' vocabularies, which is limited to 5,000 n-grams. Due to this lack of evidence, many *TRUE* labels are

incorrectly categorised as *FALSE*. Thus, potential solutions to this issue would be to either increase the size of the vocabularies, or switch to another text encoding method such as fastText word embeddings [9]. Other incorrectly *TRUE* classes incorrectly labeled *FALSE* were caused by error in the data preprocessing step, which led to a *FALSE* prediction by default.

Moreover, the models for the *at* task seems to predict the *PROBABLE* class more often than the *TRUE* class. This would suggest the lexical evidence used by the models are not sufficiently strong to distinguish between the *TRUE* and *PROBABLE* classes. This issue may be reinforced by samples incorrectly labeled *PROBABLE* in the synthetic training sets. A post-processing step to remove these incorrect instances could help solve this issue. This issue may also be reinforced by the majority voting system, which treats each class equally. A solution could be to replace this system by a more subtle one, such as a weighted vote or a graduated classification system, where only one *PROBABLE* or *TRUE* prediction would be sufficient.

Table 8
Confusion matrices between the real and predicted labels for the *at* and *isAt* tasks for the best model on each language and the Generalization (Gen.) datasets.

		English			French			German			Gen.		
	True / Pred.	True	Prob.	False	True	Prob.	False	True	Prob.	False	True	Prob.	False
at	True	**9**	9	57	**10**	17	58	**4**	16	34	**7**	41	82
	Probable	2	**6**	20	5	**1**	13	3	**15**	20	3	**20**	37
	False	5	5	**49**	12	34	**88**	13	13	**120**	13	61	**216**
isAt	True	**16**		49	**28**		53	**29**		38			
	False	20		**77**	51		**106**	35		**136**			

4. Conclusion

In this paper, we presented the results of the ROSTI team for the HIPE 2026 shared tasks. We proposed a frugal approach based on Logistic Regression models which determine the *at* and *isAt* relation between a pair of entities (*person, location*) based on pairs of sentences where these entities occur. In order to train these models, we generate a synthetic dataset combining sentences and entity mentions using a weak supervision approach.

As shown by the official evaluation, our approach achieves low to average scores in the Accuracy profile in every language and on the Generalization dataset, which are lower than those obtained by the baseline system. However, our approach ranks 6 overall in the Efficiency profile, and notably ranks first on the German dataset. This validates our approach, which shows a favorable efficiency trade-off, while being smaller than the baseline or other proposed approaches, and independent from any other external resources or other models.

Since many errors made by our approach are caused by a lack of sufficiently strong lexical evidence, we intend to keep experimenting with the model's vocabulary, as well as with other encoding methods such as word embeddings, in future works. Moreover, as some incorrect predictions appears to be caused by the majority voting system used to aggregate the predictions of our model, we intend to experiment with other voting method, such as a weighted or a graduated system.

Declaration of use of Generative AI

The authors declare that no Generative AI has been used for this work, neither for performing the experiments nor for writing the paper.

References

[1] J. Opitz, M. Ehrmann, S. Clematide, R. Corina, E. Boros, A. Michail, M. Romanello, CLEF HIPE-2026 - Shared Task Participation Guidelines, 2026. URL: https://doi.org/10.5281/zenodo.20082076. doi:10.5281/zenodo.20082076.

[2] 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.

[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] 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), 2017, pp. 35–45. URL: https://nlp.stanford.edu/pubs/zhang2017tacred.pdf.

[5] 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 ACL 2019, 2019.

[6] N. Gutehrlé, Information extraction from unstructured documents for the valorisation of historical periodicals : application to the heritage of the Bourgogne Franche-Comté Region in France, Theses, Université Bourgogne Franche-Comté, 2024. URL: https://theses.hal.science/tel-04719778. doi:10.70675/076a7fdfzf7a8z490bz8d87zf3dac2103ce3.

[7] S. Bird, E. Loper, NLTK: The natural language toolkit, in: Proceedings of the ACL Interactive Poster and Demonstration Sessions, Association for Computational Linguistics, Barcelona, Spain, 2004, pp. 214–217. URL: https://aclanthology.org/P04-3031/.

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[9] 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. URL: https://aclanthology.org/Q17-1010/. doi:10.1162/tacl_a_00051.