Leveraging Open Large Language Models for Historical Named Entity Recognition

Carlos-Emiliano González-Gallardo, Tran Thi Hong Hanh, Ahmed Hamdi, Antoine Doucet

Leveraging Open Large Language Models for Historical Named Entity Recognition

Carlos-Emiliano González-Gallardo[1,2] [0000−0002−0787−2990],
Hanh Thi Hong Tran[1,3,4] [0000−0002−5993−1630],
Ahmed Hamdi[1] [0000−0002−8964−2135], and Antoine Doucet[1] [0000−0001−6160−3356]

[1] L3i, University of La Rochelle, La Rochelle, France
{thi.tran,ahmed.hamdi,antoine.doucet}@univ-lr.fr
[2] LIFAT, University of Tours, Blois, France
carlos-emiliano.gonzalez-gallardo@univ-tours.fr
[3] Jožef Stefan International Postgraduate School, Ljubljana, Slovenia
[4] Jožef Stefan Institute, Ljubljana, Slovenia

Abstract. The efficacy of large-scale language models (LLMs) as few-shot learners has dominated the field of natural language processing, achieving state-of-the-art performance in most tasks, including named entity recognition (NER) for contemporary texts. However, exploration of NER in historical collections (e.g., historical newspapers and classical commentaries) remains limited. This presents a greater challenge as historical texts are often noisy due to storage conditions, OCR extraction, and spelling variation. In this paper, we conduct an empirical evaluation comparing different Instruct variants of open-access and open-sourced LLMs using prompt engineering through deductive (with guidelines) and inductive (without guidelines) approaches against the fully supervised benchmarks. In addition, we study how the interaction between the Instruct model and the user impacts the entity prediction. We conduct reproducible experiments using an easy-to-implement mechanism on publicly available historical collections covering three languages (i.e., English, French, and German) with code-switching on Ancient Greek and four open Instruct models. The results show that Instruct models encounter multiple difficulties handling the noisy input documents, scoring lower than fine-tuned dedicated NER systems, yet the resulting predictions provide entities that can be used in further tagging processes by human annotators.

Keywords: Digital humanities · Historical documents · NER · Instruct large-scale language models

1 Introduction

Digital libraries are centralized platforms dedicated to cultural heritage preservation. They are a major element of the digital revolution and provide researchers, students, and users on social sciences and humanities (SSH) with rich collections of digitized multimedia sources, including newspapers, books, manuscripts, and

images. Enriching these digital repositories with automatic document analysis techniques provides further value and furnishes the digital humanities (DH) community with automatic and semiautomatic tools that facilitate their analysis. A clear example is the mass digitization process initiated in the 1980s that led to the "rise of digitization" with large-scale digitization campaigns across the industry and further automatic text recognition (ATR) processes[9,48]. By providing access to transcriptions generated through these techniques, natural language processing (NLP) and information extraction (IE) can be leveraged to develop methods for semantic enrichment and advanced analysis. In this context, named entity recognition (NER) is a fundamental task of IE that serves as a pillar to other tasks like entity linking, event detection, and knowledge graph creation [6,30,32]. Its main objective is to identify and classify entities automatically. These entities may be universal and refer to persons, places, organizations, or dates; nevertheless, they also vary depending on the corpus and may refer to domain-specific concepts [18].

Developing NER methods for historical documents is a multi-factor challenge linked to the need to handle documents deteriorated by the effect of time, the poor quality printing materials, the quality of digitization, and the inaccurate scanning processes [24]. In addition, NER systems based on language models trained on contemporary data face a diachronic challenge given the language change and evolution of historical documents that traverse an enormous span [18]. NER in historical corpora dates back to the early 2000s with the development of rule-based models. However, the aforementioned characteristics of historical corporal posed major challenges leading to a paradigm shift towards machine learning models and, latterly, towards deep learning and large-scale language models (LLMs). Lastly, Instruct models like ChatGPT[5], Llama-chat[6], and Mixtral[7] have proven their usefulness in multiple NLP tasks by reducing the need for considerable annotated data and providing a natural language interface between the user and the LLM.

In this study, we investigate the potential of open-access and open-sourced Instruct models in recognizing and classifying named entities of three publicly available historical collections covering English, French, and German with code-switching on Ancient Greek. We experiment with deductive prediction, where we provide the Instruct model with the guidelines that were supplied to humans to annotate the corpora. We compare the performance against an inductive prediction, where we prompt the model with just a few examples of annotated sentences. We also investigate the influence of single- vs multi-turn interaction in entity prediction. To our knowledge, this is the first work that studies historical NER with open Instruct models in a multidimensional perspective[8].

[5] `https://openai.com/chatgpt`
[6] `https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md`
[7] `https://mistral.ai/fr/technology/#models`
[8] Code and data are available in `https://github.com/cic4k/LLMs_for_historical_NER`

The paper is structured as follows: we first present the work that has been done concerning NER on historical documents in Section 2. Second, in Section 3, we explain the workflow we follow to perform NER on multilingual historical corpora using open Instruct models, including the prompt designs, the post-processing steps, and the evaluation metrics. Then, in Section 4, we present the obtained results followed by a detailed error analysis where we highlight the main complications Instruct models encounter when facing this kind of corpora. Finally, in Section 5, we conclude and discuss future work.

2 Related work

2.1 Named Entity Recognition in Historical Corpora

Historical named entity recognition (HNER) systems have evolved in the last 20 years, adapting their architectures to the following paradigms: (1) rule-based models, (2) traditional machine learning models, and (3) deep learning or neural models.

Rule-based models Traced back to the beginning of the 2000s, most of the existing NER works for historical documents used rule-based or so-called symbolic systems, which were applied to various document types like newspapers [10,34] and poetry [11]; domains, including legal [3] and literature [1,4]; and periods (e.g., 12^{th}-15^{th} [11], 18^{th} [3], and 19^{th} [4,1] centuries). These systems are often modular and almost systematically include gazetteer lookup [1,10,31], rule incremental application [21], and variant matching [11]. Despite their suitability for non-experts, they have difficulties in dealing with noisy and historical input, requiring normalization rules and additional linguistic knowledge. Thus, research moved away from such systems in favor of machine-learning ones.

Machine learning models Unlike rule-based models, machine learning ones inductively learn statistical models from annotated training data given the manually selected features. Two approaches have been considered: applying already existing models [29,40,51] and training new ones [17,33,35]. The settings are quite diverse, such as using a single classifier (e.g., CRF [35,37]), a single classifier with additional features (e.g., CRF + PCFG [12], CRF + gaz [42]), or ensemble [34,51]. Unlike their high performance in contemporary corpora, the performances in HNER are often relatively low (around 60-70%) and can be slightly higher if the systems were trained on in-domain data. Deep learning models emerged as a solution to address performance limitations in machine learning methods.

Deep learning models Recent advances in representation learning and deep neural networks have also influenced HNER tasks. On one hand, several embedding techniques have been investigated for the task at hand to evaluate which type of embedding is best, e.g., static [45], character-level [47], word-level [52], or stack of modern embeddings [39]. While character and sub-word information

deal better with out-of-vocabulary words, the word-based contextualized embeddings perform better than static ones given the same model. On the other hand, the potential of transferring knowledge from modern embedding to historical texts (*transfer learning*) and the impact of in-domain embeddings was also exploited by testing modern vs. historical static, historical char-level, historical word-level, and a stack of embeddings, respectively. The pre-trained modern embeddings transfer reasonably well to historical documents, especially when learned from huge textual corpora, and the combination of generic and historical prior knowledge is likely to improve the performance [14]. Besides, different neural architectures have been explored against the traditional CRF, such as LSTM [26], BiLSTM [46], softmax decoder [39], transformers [7], graph transformers [25], temporal transformers [18], and stacked transformers [5], to mention a few. State-of-the-art (SOTA) transformers achieve very good performance and surpass the traditional machine learning models and other language models. We consider the best transformer-based models, namely temporal NER [18] stacked NER [5], as the benchmarks in our study. A systematic review of HNER datasets and systems can be found at [14].

2.2 Large-scale Language Models

The rise of LLMs has dominated the field of NLP, achieving top performance in various tasks, including NER for contemporary corpora [2,28]. This advancement creates new opportunities for extracting named entities from historical sources, including historical newspapers and classical commentaries. Previous studies such as [19,20] explored the ability of zero-shot prompt engineering with close-sourced Instruct models like ChatGPT in HNER compared to the language model-based benchmarks. While promising, ChatGPT suffered several challenges in recognizing the historical entities, including the inconsistency of annotation guidelines, entity complexity, multilingualism, code-switching, and prompting specificity. Besides zero-shot prompting, as demonstrated in the work of [20], two popular strategies for incorporating LLMs include fine-tuning and in-context learning (ICL). While the fine-tuning involves initializing a pre-trained model and conducting additional training epochs on task-specific supervised data [22,36,38], ICL leverages the Instruct model's ability to generate texts with only a few task-specific examples as demonstrations (or few-shot demonstrations). However, none of these approaches have been explored in HNER tasks.

In the present study, we build upon the research conducted by [20] to recognize named entities in eight historical datasets. We focus on four open Instruct models using deductive and inductive approaches, and single- and multi-turn interaction.

3 Few-shot Prompting for Historical NER

In this section, we present the workflow and experimental setup we followed to study the potential of open Instruct models to perform NER on historical doc-

uments. We first describe the general architecture of the NER system workflow, followed by a detailed explanation of each component.

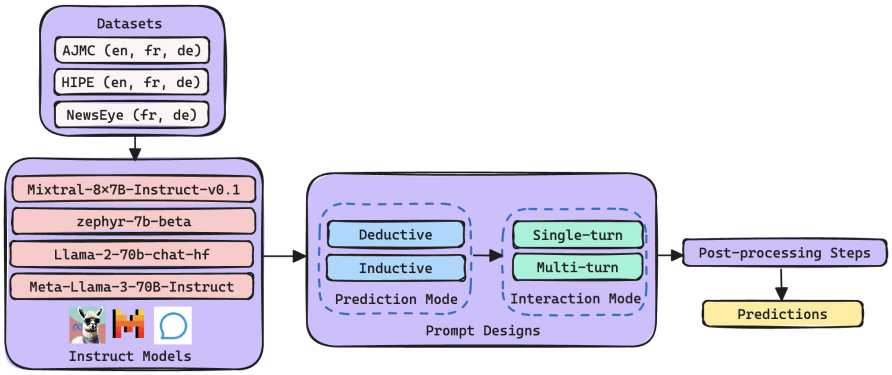

Fig. 1. General workflow for few-shot prompting HNER

The overall workflow of our proposed prompting system for HNER is visualized in Figure 1. It is composed of four stages, the first two in charge of the dataset and the Instruct model selection. The *Prompt Designs* stage is in charge of defining the prediction and the interaction mode with the model. Lastly, the LLM-generated text has to be parsed, aligned, and transformed into IOB, which is the format that most NER datasets and evaluators use. The Post-processing stage performs these tasks.

3.1 Datasets

In this study, we used a subsection of the corpora proposed by the CLEF-HIPE-2022 evaluation lab on historical newspapers and classical commentaries [9]. The selected corpora comprise three historical document datasets spanning roughly 200 years between the 19^{th} and 20^{th} centuries. These datasets include ancient commentaries (i.e., *AJMC*) and historical newspapers (i.e., *HIPE* and *NewsEye*), gathered from digital libraries through different national and international research projects, including Ajax Multi-Commentary[10], impresso[11], and NewsEye[12].

The *AJMC* dataset [41] is composed of classical commentaries from the Ajax Multi-Commentary project that includes digitized 19^{th} century commentaries published in English, French, and German. These commentaries provide in-depth analysis and explanation of Sophocles' Ajax Greek tragedy.

[9] https://hipe-eval.github.io/HIPE-2022/about
[10] https://mromanello.github.io/ajax-multi-commentary/
[11] https://impresso-project.ch/
[12] https://www.newseye.eu/

The *HIPE* dataset [15] is composed of Swiss, Luxembourgish, and American newspaper articles in French, German, and English comprising the 19^{th} and 20^{th} centuries. It has been collected mainly through the National Library of Switzerland (NB[13]), the National Library of Luxembourg (BnL[14]), the Media Center and State Archives of Valais, and the Swiss Economic Archives (SWA[15]) as part of the impresso project.

The *NewsEye* dataset [23] is a collection of French, German, Finnish, and Swedish newspapers collected through the national libraries of France (BnF[16]), Austria (ONB[17]), and Finland (NLF[18]). The French corpus is composed of items from digitized archives of nine newspapers (i.e., *L'Oeuvre, La Fronde, La Presse, Le Matin, Marie-Claire, Ce soir, Marianne, Paris Soir* , and *Regards*) between 1854 and 1946. Meanwhile, the German corpus contains articles extracted from four newspapers (i.e., *Arbeiter-Zeitung, Mittags-Zeitung, Illustrierte Kronen Zeitung*, and *Neue freie Presse*) between 1864 and 1933. Finally, the Finnish and Swedish corpora, both contain articles from two newspapers (*Fraktur* and *Antique*) published from 1852 to 1918 for Finnish and from 1848 to 1918 for Swedish, respectively. In this study, we focused only on French and German languages.

All datasets were annotated with universal (i.e., person, location, organization) and domain-specific (i.e., bibliographic references to primary and secondary literature) entity types and subtypes for the NER task and split into train, development, and test partitions. The statistic details of the number and type of entities found in the specified datasets can be found at [16].

3.2 Instruct Models

This stage determines the Instruct model that will be prompted to annotate the dataset selected in the *Datasets* block. In this study, we focus on four open Instruct models available on the HuggingFace inference API[19] which permits access to models that can hardly be run on small or medium infrastructures.

Llama-2-chat, developed by MetaAI, is one of the pretrained and fine-tuned generative open-access text models of the Llama 2 family[20]. We used the 70 billion (70B) parameter fine-tuned model version, so-called *Llama-2-70b-chat-hf*[21] optimized for dialogue use cases. This is an auto-regressive language model with 4k context length and 2T tokens that uses an optimized transformer architecture whose tuned versions use supervised fine-tuning (SFT) and reinforcement learning with human feedback (RLHF) to align with human preferences for helpful-

[13] `https://www.nb.admin.ch`
[14] `https://bnl.public.lu`
[15] `https://wirtschaftsarchiv.ub.unibas.ch`
[16] `https://bnf.fr`
[17] `https://onb.ac.at`
[18] `https://kansalliskirjasto.fi`
[19] `https://huggingface.co/docs/api-inference/index`
[20] `https://llama.meta.com/llama2/`
[21] `https://huggingface.co/meta-llama/Llama-2-70b-chat-hf`

ness and safety with additional Grouped-Query Attention (GQA) for improved inference scalability for our chosen version.

The Llama 3 family[22] was released as the most capable openly available LLM to date (the latest version of LLMs in the Llama series) with several key improvements in comparison with Llama 2. Llama 3 uses a tokenizer with 15T+ tokens that encodes language much more efficiently, which leads to substantially improved model performance as well as improved inference efficiency with the GQA mechanism. The model was trained on sequences of 8,192 tokens, using a mask to ensure self-attention does not cross document boundaries. We used the *Meta-Llama-3-70B-Instruct*[23] Instruct model, which is a 70 billion instruction tuned generative text model.

Mistral[24] [27] is a pretrained generative text model with 7 billion (7B) parameters developed by Mistral AI. In our experiment, we focus on the up-to-date *Mixtral-8x7B-Instruct-v0.1*[25] open-sourced Instruct model, which is a sparse mixture-of-experts (MoE) model with 8 expert models, each with 7B parameters. This makes it one of the largest and most powerful LLMs available to the public and surpasses even Llama 2 on most tested benchmarks.

Zephyr is a collection of language models designed to serve as intelligent assistants. Our experiment focuses on the series's second version *zephyr-7b-beta*[26] Instruct model. This 7B parameter GPT-like model was also a fine-tuned version of Mistral AI's *Mistral-7B-v0.1* model trained on a mixed combination of publicly available, synthetic datasets using Direct Preference Optimization.

3.3 Prompt Designs

This stage defines the prediction mode and the type of interaction with the Instruct model. As visualized in Figure 2 it is a 4-block procedure composed of the task description, the prediction and interaction modes, and the input query.

1. Task Description This block defines the frame in which the NER will be performed. It first specifies the task that the Instruct model will perform (*"You are an excellent automatic named entity recognition (NER) system."*). Then, it indicates how the input will be presented (*"I will provide you the sentence delimited by double quotes ..."*). Finally, it specifies the named entity types that are required (*"... from which you need to identify and classify the named entities into the following types: {`CORPUS_DEPENDENT_ENTITY_TYPES`}"*). Each corpus has a specific number of entity types, in consequence, this section of the task description will vary from dataset to dataset. The example shown in Figure 2 belongs to *HIPE-2020*, which has named entities for persons, organizations, human productions, temporal expression, and locations.

[22] `https://llama.meta.com/llama3/`
[23] `https://huggingface.co/meta-llama/Meta-Llama-3-70B-Instruct`
[24] `https://mistral.ai/news/mixtral-of-experts/`
[25] `https://huggingface.co/mistralai/Mixtral-8x7B-v0.1`
[26] `https://huggingface.co/HuggingFaceH4/zephyr-7b-beta`

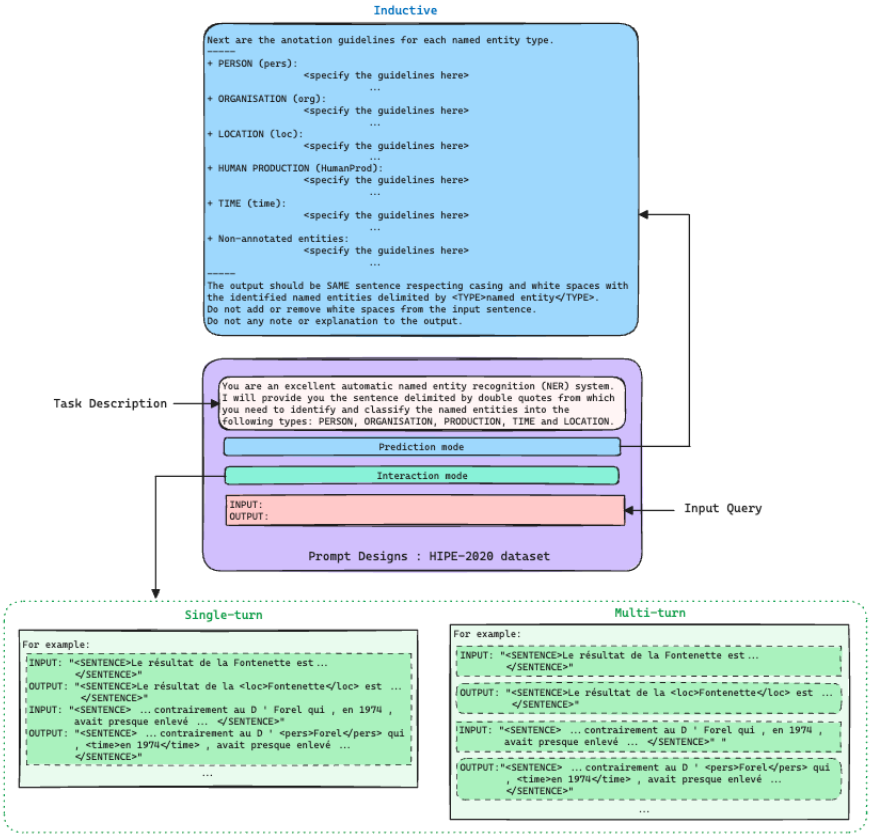

Fig. 2. The prompt designs scheme for HNER

2. Prediction Mode One of the difficulties of performing NER on historical documents is that the named entities they contain can differ from those that are found in contemporary corpora. These differences may be related to the definition and complexity of an entity. For example, the impresso guidelines [13] state that occupation, administrative function, and social role or status are part of a PERSON entity type; thus *"Tom Scarlett, Circuit Court Clerk for Putnam County"* corresponds to an entity of type PERSON in the phrase *"By virtue of a bil of sale issued by Tom Scarlett, Circuit Court Clerk for Putnam County,."*.

Different studies have investigated the capacity of LLMs to perform NLP tasks via in-context few-shot learning [8]. [44] explored the effects of the source and size of the training corpus and discovered that corpus sources play an important role in whether or not in-context learning ability will emerge in a LLM. [50] experimented with chain-of-thought prompting to enhance reasoning in language models. This permitted the model to tackle complex arithmetic, commonsense, and symbolic reasoning tasks. [49] created an information extraction

framework that models various information extraction tasks based on instruction tuning. They evaluate the framework on NER, relation extraction, and entity linking, achieving competitive results compared to fully supervised models. These studies follow an inductive process which means that given a set of specific examples, the model needs to generalize for each unseen case. [43] explored a different approach where an inductive process is used. They proposed a code-oriented fine-tuned model trained on information extraction-related task annotation guidelines. They observed that the model was able to apply and even follow unseen guidelines. We experimented with these two prediction modes. In the **inductive** mode, we provide the model only a set of examples containing at least one instance of each entity type within the dataset. In the **deductive** prediction mode, we provide the model with the same guidelines that human annotators follow to annotate the datasets. The *Prediction Mode* block in Figure 2 exemplifies the deductive prediction with the *HIPE-2020* annotation guidelines.

3. Interaction Mode This block permits two types of interactions that influence the way manner in which the few-shot examples are provided to the Instruct model. The **single-turn** interaction communicates with the model only once. It sends in a unique prompt all the examples that are shown to the Instruct model along with the task description, the prediction mode, and the input query. In return, the model is expected to respond with the requested phrase labeled with the predicted named entities. In contrast, the **multi-turn** interaction utilizes the system, user, and end-of-sentence tokens of the Instruct model to simulate a conversation where an example is sent at each interaction. In this configuration, the first prompt comprises the task description, the prediction mode, and the first example. The subsequent examples are formatted, interleaving the input sentence as the user portion of the prompt and the desired output as the assistant portion.

4. Input Query Finally, this block shapes the sentence from which named entities will be extracted in the same format that the few-shot examples, delimiting it by double quotes and encapsulating it between the `<SENTENCE></SENTENCE>` tags. The model is expected to return the predictions in the required format as specified in the *Prediction mode* and *Interaction mode* blocks.

3.4 Post-processing Steps

The generative nature of the Instruct models impacts the predictions that are produced. Even with explicit instructions stating to respect the input sentence and not add any note or explanation to the out, most predictions have to follow a parsing and alignment process. This stage removes all the text that is not part of the original sentence (except of the predicted tags), inserts the text that has been removed from the sentence, and replaces the characters that have been modified. Lastly, the cleaned output is transformed into the IOB annotation scheme that permits the automatic evaluation of the predicted entities.

Let $D = \{\texttt{AJMC}, \texttt{HIPE}, \texttt{NewsEye}\}$ be the collection of all the datasets included in our study and $L = \{\texttt{en}, \texttt{fr}, \texttt{de}\}$ be the possible languages the datasets cover. As the NewsEye dataset does not contain an English version, overall, the cardinality of datasets is calculated as

$$|C_3^3| - 1 = \binom{3}{3} - 1 = 8. \tag{1}$$

Let $M = \{\texttt{Llama-3}, \texttt{Llama-2}, \texttt{Mistral}, \texttt{Zephyr}\}$ be the collection of open-sourced LLMs that correspond to *Llama-3-70B-Instruct*, *Llama-2-70b-chat-hf*, *Mixtral-8x7B-Instruct-v0.1*, and *zephyr-7b-beta* as described in Section 3.2; $F = \{\texttt{G}, \texttt{nG}\}$ be the deductive and inductive prediction modes; and $C = \{\texttt{R}, \texttt{nR}\}$ the multi- and single-turn interaction modes. For each data $D_i \times L_j$ (i.e., $\texttt{AJMC}\times\texttt{en}$), we run the experiments on the combinations of different RLHF models, and prompt designs as $M_k \times F_m \times C_n$, where M_k refers element k in the model collections M, F_n refers element n in the prediction mode, and C_n refers element n in the interaction mode of the conversation. Overall, 128 experiments were conducted in this study.

3.5 Evaluation Metrics

To evaluate the efficiency of the few-shot prompting historical NER workflow, we calculate precision (P), recall (R), and F1-score (F1) at the micro level (i.e., consideration of all true positives, false positives, true negatives, and false negatives over all samples) in a strict (exact boundary matching) and a fuzzy boundary matching set. These are also the evaluation metrics used in related works and benchmarks [5,18,20] on the same datasets, making the results comparable with SOTA systems[27].

4 Results

Results are presented from Tables 1 to 3. It can be observed that the use of LLMs for recognizing named entities within historical contexts has revealed several notable outcomes. Despite the diverse datasets employed and various prompt strategies tested, LLMs consistently exhibit low F1-scores, often falling below the 40% F1-score. This trend persists regardless of language or dataset, showcasing a notable challenge in leveraging LLMs for NER in historical contexts. Specifically, in `HIPE` and `AJMC`, LLMs yield scores comparable to [20], indicating a relative parity in performance. However, when confronted with `NewsEye`, LLMs exhibit markedly lower scores, trailing significantly behind ChatGPT's [20] performance. In addition, the NER performance significantly varies when employing fuzzy boundary matching compared to strict exact boundary matching. This discrepancy underscores the ability of LLMs to often detect parts of named entities while encountering challenges in precisely identifying all their tokens. Except for

[27] The scorer is available at `https://github.com/hipe-eval/HIPE-scorer`

Table 1. NER strict and fuzzy micro results in `NewsEye` dataset. For each evaluation metric, bold represents the highest score for each setting, and underline represents the highest score above all four settings.

		strict						fuzzy					
		fr			de			fr			de		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
G + R	Llama-3	**42.9**	16.0	**23.3**	**18.1**	15.2	**16.5**	**55.5**	20.7	30.2	**24.5**	20.5	22.3
	Llama-2	16.7	21.1	18.6	9.0	**21.2**	12.6	27.7	35.1	31.0	13.2	31.3	18.6
	Mistral	24.3	**21.9**	23.0	12.7	20.1	15.6	40.0	**36.1**	**38.0**	21.0	**33.3**	**25.7**
	Zephyr	31.5	17.9	22.8	12.9	11.0	11.9	49.7	28.2	36.0	21.7	18.5	20.0
G + nR	Llama-3	**40.7**	15.4	22.4	**18.3**	14.7	16.3	24.9	20.0	22.2	**50.2**	27.6	**35.7**
	Llama-2	31.7	**29.6**	**30.6**	12.6	**29.6**	**17.7**	**44.7**	**41.7**	**43.1**	17.7	**41.6**	24.9
	Mistral	21.8	13.2	16.4	10.5	13.9	12.0	38.9	23.4	29.2	19.1	25.2	21.7
	Zephyr	30.0	13.1	18.2	11.4	7.5	9.1	22.2	14.6	17.6	33.0	19.1	24.2
nG + R	Llama-3	**37.9**	16.6	23.0	**16.0**	**15.1**	**15.5**	50.4	22.0	30.7	**22.1**	20.9	**21.5**
	Llama-2	21.1	**22.8**	22.0	8.5	13.2	10.3	33.6	36.3	34.9	12.3	19.2	15.0
	Mistral	19.3	14.9	16.8	9.4	12.1	10.6	33.8	26.0	29.4	19.5	**25.1**	21.9
	Zephyr	35.9	19.7	**25.4**	14.0	10.1	11.7	**53.3**	**29.2**	**37.7**	21.5	15.6	18.1
nG + nR	Llama-3	**42.3**	10.4	16.7	**16.6**	14.9	15.7	**53.3**	13.1	21.0	**23.3**	20.9	22.0
	Llama-2	28.7	**36.3**	**32.1**	11.4	**29.0**	**16.3**	40.8	**51.6**	**45.6**	16.1	**41.0**	**23.1**
	Mistral	19.6	13.5	16.0	9.5	14.0	11.3	33.5	23.0	27.3	18.0	26.4	21.4
	Zephyr	28.5	11.1	16.0	10.9	4.7	6.6	43.2	16.8	24.2	**23.3**	10.1	14.1
SOTA	Stacked NER [5]	75.0	70.6	72.7	64.9	50.2	56.6	85.4	80.5	82.9	82.3	66.4	73.5
	ChatGPT [20]	70.9	72.3	71.6	-	-	-	77.8	79.4	78.6	-	-	-

Table 2. NER strict and fuzzy micro results on `HIPE` dataset. For each evaluation metric, bold represents the highest score for each setting, and underline represents the highest score above all four settings.

		strict									fuzzy								
		en			fr			de			en			fr			de		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
G + R	Llama-3	**25.5**	**23.8**	**24.6**	**36.9**	22.1	27.6	**32.6**	**30.1**	**31.3**	39.1	**36.5**	**37.7**	**47.2**	28.3	35.4	**42.6**	**39.3**	**40.9**
	Llama-2	20.0	21.2	20.6	25.7	25.6	25.6	20.4	26.5	23.0	32.0	33.9	32.9	36.5	36.3	36.4	28.3	36.8	32.0
	Mistral	19.5	18.0	18.8	25.8	**34.8**	**29.6**	20.6	21.4	21.0	34.2	31.6	32.9	38.1	**51.4**	**43.8**	32.1	33.3	32.7
	Zephyr	25.4	3.3	5.9	24.8	20.8	22.6	20.1	10.6	13.8	**45.8**	6.0	10.6	34.8	29.1	31.7	29.4	15.4	20.2
G + nR	Llama-3	**25.6**	22.9	24.2	**38.6**	21.3	27.4	**34.6**	29.6	**31.9**	**40.9**	36.8	38.7	**50.2**	27.6	35.7	**46.2**	39.6	**42.7**
	Llama-2	21.2	**32.7**	**25.7**	31.3	**37.2**	**34.0**	21.3	**36.2**	26.9	32.1	**49.7**	**39.0**	41.9	**49.9**	**45.6**	28.5	**48.3**	35.8
	Mistral	15.6	13.1	14.3	25.6	17.6	20.8	19.1	19.7	19.4	31.2	26.3	28.5	39.9	27.4	32.5	30.7	31.7	31.2
	Zephyr	21.2	6.2	9.6	23.4	13.5	17.1	18.8	10.7	13.7	34.1	10.0	15.5	33.0	19.1	24.2	28.4	16.2	20.7
nG + R	Llama-3	**24.5**	**25.2**	**24.8**	**34.6**	22.2	27.1	**29.2**	**30.1**	**29.6**	**37.7**	**38.8**	**38.2**	**44.9**	28.8	35.1	**39.9**	**41.1**	**40.5**
	Llama-2	21.1	22.1	21.6	26.5	**28.2**	**27.3**	21.6	29.0	24.7	33.0	34.5	33.7	38.0	**40.4**	39.2	30.0	40.4	34.4
	Mistral	17.1	19.2	18.1	26.8	26.9	26.9	19.5	21.0	20.2	30.8	34.5	32.6	39.2	39.4	**39.3**	29.8	32.3	31.0
	Zephyr	20.6	5.8	9.0	28.2	20.4	23.7	22.8	12.1	15.8	34.1	9.6	15.0	37.9	27.4	31.8	34.8	18.5	24.1
nG + nR	Llama-3	**28.3**	24.1	**26.0**	**41.1**	21.1	27.9	**35.6**	30.0	**32.5**	**44.0**	37.4	40.4	**54.0**	27.8	36.7	**47.7**	40.2	**43.6**
	Llama-2	23.8	**28.3**	25.9	31.7	**35.4**	**33.5**	24.6	**36.4**	29.4	39.0	**46.3**	**42.4**	43.8	**48.8**	**46.2**	33.3	**49.3**	39.8
	Mistral	16.9	14.0	15.3	30.7	23.6	26.7	20.2	18.7	19.5	34.6	28.7	31.4	45.2	34.7	39.3	33.2	30.7	31.9
	Zephyr	26.2	3.6	6.3	30.4	9.7	14.7	26.0	6.4	10.2	39.3	5.4	9.4	40.0	12.8	19.4	40.2	9.9	15.8
SOTA	Stacked NER [5]	-	-	-	83.5	84.9	84.2	78.6	78.7	78.7	-	-	-	91.3	92.9	92.1	91.3	92.9	92.1
	Temporal NER [18]	64.3	61.7	63.0	76.5	76.5	76.5	75.9	76.7	76.3	78.7	80.0	79.3	86.7	86.7	86.7	85.2	85.7	85.4
	ChatGPT [20]	-	-	-	32.5	50.0	39.4	-	-	-	-	-	-	49.0	75.4	59.4	-	-	-

`Llama-2`, which demonstrates consistent results in the two scenarios, it indicates its capability to either accurately identify complete named entities or fail to

Table 3. NER strict and fuzzy micro results on AJMC dataset. For each evaluation metric, bold represents the highest score for each setting, and underline represents the highest score above all four settings.

		strict									fuzzy								
		en			fr			de			en			fr			de		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
G + R	Llama-3	27.4	9.2	13.8	24.3	5.0	8.3	35.2	11.8	17.7	37.6	12.6	18.9	31.1	6.4	10.6	40.6	13.6	20.4
	Llama-2	16.8	8.1	10.9	**41.6**	17.8	24.9	29.6	12.6	17.7	44.3	21.3	28.7	**52.6**	22.5	31.5	46.9	19.9	27.9
	Mistral	26.0	**33.1**	**29.1**	25.4	**30.8**	**27.9**	27.9	**41.6**	**33.4**	40.1	**50.9**	**44.8**	33.2	**40.3**	**36.4**	38.7	**57.9**	**46.4**
	Zephyr	**36.8**	16.4	22.7	41.1	17.2	24.3	**42.5**	18.6	25.9	**47.7**	21.3	29.4	51.7	21.7	30.5	**48.5**	21.2	29.5
G + nR	Llama-3	26.5	8.9	13.3	27.6	6.7	10.7	40.2	10.7	16.9	35.9	12.1	18.1	32.2	7.8	12.5	**48.0**	12.8	20.3
	Llama-2	21.8	10.6	14.3	**48.7**	**21.1**	**29.5**	29.2	13.1	18.1	**51.8**	**25.3**	**34.0**	**62.2**	**27.0**	**37.6**	**48.0**	21.5	**29.7**
	Mistral	23.6	**14.4**	**17.9**	27.3	14.7	19.1	21.3	**20.4**	**20.8**	36.3	22.1	27.5	30.9	16.7	21.7	30.0	**28.8**	29.4
	Zephyr	**32.6**	8.9	14.0	36.5	8.6	13.9	**41.3**	10.0	16.0	43.2	11.8	18.5	45.9	10.8	17.5	46.7	11.3	18.1
nG + R	Llama-3	21.3	9.2	12.9	19.5	6.1	9.3	23.6	10.0	14.0	28.7	12.4	17.3	25.7	8.1	12.3	26.7	11.3	15.8
	Llama-2	18.6	10.1	13.1	36.2	**21.1**	**26.7**	28.7	15.7	20.3	**43.6**	23.6	**30.6**	**48.6**	**28.3**	**35.8**	**42.6**	23.3	**30.1**
	Mistral	18.9	**21.6**	20.2	20.8	16.9	18.7	21.7	**21.5**	**21.6**	28.5	**32.5**	30.4	27.0	21.9	24.2	30.2	**29.8**	30.0
	Zephyr	**30.0**	15.5	**20.5**	**38.0**	15.0	21.5	**34.0**	13.6	19.4	37.8	19.5	25.8	47.9	18.9	27.1	41.8	16.8	23.9
nG + nR	Llama-3	22.4	9.2	13.0	22.6	6.4	10.0	24.3	8.9	13.0	30.8	12.6	17.9	27.5	7.8	12.1	28.6	10.5	15.3
	Llama-2	22.3	14.1	**17.3**	**47.0**	**21.7**	**29.7**	**34.4**	17.5	**23.2**	**50.9**	**32.2**	**39.4**	**56.0**	**25.8**	**35.4**	**50.3**	**25.7**	**34.0**
	Mistral	13.8	**15.8**	14.7	15.0	16.1	15.5	25.9	**18.1**	21.3	21.6	24.7	23.0	18.6	20.0	19.3	32.3	22.5	26.5
	Zephyr	**29.4**	5.8	9.6	29.5	6.4	10.5	25.0	6.0	9.7	39.7	7.8	13.0	42.3	9.2	15.1	31.5	7.6	12.2
SOTA	Temporal NER [18]	86.6	88.8	87.7	84.8	83.9	84.4	92.1	91.1	91.6	92.2	94.5	93.3	90.2	89.2	89.7	87.0	87.2	87.1
	ChatGPT [20]	-	-	-	21.8	26.1	23.8	-	-	-	-	-	-	25.5	30.6	27.8	-	-	-

detect them altogether. Furthermore, a recurring pattern emerges where recall consistently lags behind precision across all datasets and prompt strategies. Particularly concerning is the observation of very poor recall over AJMC, signifying a significant deficiency in the ability of LLMs to accurately identify named entities within this specific historical context. While Zephyr demonstrates a degree of consistency in its results across different languages and prompts, its overall performance remains very low, particularly when evaluated against AJMC. Moreover, it is noteworthy that Mistral consistently benefits from prompts with interactions, a recommendation not extended to Llama-2. Interestingly, Llama-2, Mistral, and Zephyr display improved performance when applied to French texts, suggesting a potential language-specific advantage for these LLMs. Conversely, Llama-3 mostly exhibits superior performance in German, except for NewsEye, where it achieves better results in French. These findings underscore the nuanced challenges associated with deploying LLMs for named entity recognition within historical contexts, emphasizing the importance of dataset-specific considerations and language nuances in optimizing performance. The low recall observed can be attributed to issues with output structure. In our methodology for automatically analyzing LLM outputs, we employ a structured format within the prompt, such as requesting the use of tags to delineate named entities within a sentence. However, it is worth noting that LLMs occasionally deviate from this prescribed format. For instance, instead of systematically tagging named entities within the original sentence, they may opt to restate the input sentence and subsequently list the named entities in a separate sentence. This inconsistency in output formatting poses a challenge to accurate recall, as it requires hard post-processing to reconcile the named entities with their corresponding

context. Likewise, LLMs almost present deviations from the specified named entity classes outlined in our prompts. Instead, they often align with similar but not identical classes, resulting in variations such as *prod* being replaced by *humprod*, *humanprod*, *production*, and so forth. To address this issue, we implemented post-processing methodologies aimed at normalizing these classes and ensuring comprehensive coverage. However, our analysis also revealed an interesting trend wherein LLMs tend to propose named entity classes that were not explicitly included in the prompts. These unexpected classes encompass a wide range, including *animal*, *attribute*, *ideology*, *relation*, *class*, *currency*, *amount*, and *disease*. While the absence of guidelines in the prompts can partially explain this behavior, it remains surprising when guidelines are indeed provided, suggesting potential areas for further investigation into LLM behavior and prompt adherence.

Using the metadata from `AJMC` to analyze the errors, LLMs clearly encounter difficulties in recognizing noisy named entities. For instance, in the case of German, approximately 95% of correctly recognized named entities are devoid of noise. Notably, no noisy named entity with more than 25% of erroneously recognized characters is accurately identified. Similarly, for French, only 5% of noisy named entities are successfully recognized. In the case of English, only one noisy named entity is recognized at all. These findings show the LLMs' struggle with noisy inputs across different languages, with varying degrees of success in noise mitigation depending on the language.

LLMs are also affected by orthographic changes. In the German corpus, for instance, between the years 1789 and 1838, `Llama-3` can only recognize 13% of named entities, a figure that increases to 27% during the period from 1848 to 1898. However, there is a notable improvement in performance for texts published after the 1900s, with an ability to identify approximately half of the named entities in the corpus. In the English corpus, around 30% of named entities before the 1900s are correctly recognized, and only 20% in the French corpus.

5 Conclusion

In this study, we conducted reproducible and easy-to-implement experiments to evaluate different open Instruct models using prompt engineering through deductive and inductive approaches against the fully supervised benchmarks. In addition, we investigated how the interaction between the Instruct models and the user impacts the entity prediction. The results demonstrated that inductive prompting surpasses the performance of deductive one by up to 5 percentage points. Against our intuition, the interaction between users and Instruct models in an inductive setting does not improve the predictive ability of the models, however, it reduces the noises of the outputs (e.g., output control, text variations) and eliminates the efforts of additional post-processing steps. Meanwhile, the non-interaction version shows the potential of capturing more candidate entities but introduces significant noise to the prediction. Overall, Instruct models do not overpass the performance of fined-tuned NER neural models trained on historical

corpora. However, they can assist the tagging process by human annotators. In future work, we want to explore few-shot learning on open Instruct models to help the model learn the difficulties of historical corpora.

Acknowledgments. This work has been supported by the ANNA (2019-1R40226), TERMITRAD (2020-2019-8510010), Pypa (AAPR2021-2021-12263410), and Actuadata (AAPR2022-2021-17014610) projects funded by the Nouvelle-Aquitaine Region (France). We also want to thank Daniel van Strien for the insightful discussions and Hugging Face, Inc. for providing partial access to its Inference API.

Disclosure of Interests. The authors have no competing interests to declare relevant to this article's content.

References

1. Alex, B., Grover, C., Tobin, R., Oberlander, J.: Geoparsing historical and contemporary literary text set in the city of edinburgh. Language Resources and Evaluation **53**, 651–675 (2019)
2. Bogdanov, S., Constantin, A., Bernard, T., Crabbé, B., Bernard, E.: Nuner: Entity recognition encoder pre-training via llm-annotated data. arXiv preprint arXiv:2402.15343 (2024)
3. Bontcheva, K., Maynard, D., Cunningham, H., Saggion, H.: Using human language technology for automatic annotation and indexing of digital library content. In: Research and Advanced Technology for Digital Libraries: 6th European Conference, ECDL 2002 Rome, Italy, September 16–18, 2002 Proceedings 6. pp. 613–625. Springer (2002)
4. Borin, L., Kokkinakis, D., Olsson, L.J.: Naming the past: Named entity and animacy recognition in 19th century swedish literature. In: Proceedings of the Workshop on Language Technology for Cultural Heritage Data (LaTeCH 2007). pp. 1–8 (2007)
5. Boros, E., Hamdi, A., Linhares Pontes, E., Cabrera-Diego, L.A., Moreno, J.G., Sidere, N., Doucet, A.: Alleviating digitization errors in named entity recognition for historical documents. In: Fernández, R., Linzen, T. (eds.) Proceedings of the 24th Conference on Computational Natural Language Learning. pp. 431–441. Association for Computational Linguistics, Online (Nov 2020). https://doi.org/10.18653/v1/2020.conll-1.35, https://aclanthology.org/2020.conll-1.35
6. Boros, E., Nguyen, N.K., Lejeune, G., Doucet, A.: Assessing the impact of ocr noise on multilingual event detection over digitised documents. International Journal on Digital Libraries pp. 1–26 (2022)
7. Boros, E., Pontes, E.L., Cabrera-Diego, L.A., Hamdi, A., Moreno, J.G., Sidère, N., Doucet, A.: Robust named entity recognition and linking on historical multilingual documents. In: Conference and Labs of the Evaluation Forum (CLEF 2020). vol. 2696, pp. 1–17. CEUR-WS Working Notes (2020)
8. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language models are few-shot learners. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H.

(eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 1877–1901. Curran Associates, Inc. (2020), https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf
9. Causer, T., Terras, M.: 'many hands make light work. many hands together make merry work'1: Transcribe bentham and crowdsourcing manuscript collections. In: Crowdsourcing our cultural heritage, pp. 57–88. Routledge (2016)
10. Crane, G., Jones, A.: The challenge of virginia banks: an evaluation of named entity analysis in a 19th-century newspaper collection. In: Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries. pp. 31–40 (2006)
11. Díez Platas, M.L., Ros Munoz, S., González-Blanco, E., Ruiz Fabo, P., Alvarez Mellado, E.: Medieval spanish (12th–15th centuries) named entity recognition and attribute annotation system based on contextual information. Journal of the Association for Information Science and Technology **72**(2), 224–238 (2021)
12. Dinarelli, M., Rosset, S.: Tree-structured named entity recognition on ocr data: Analysis, processing and results. In: Language Resources Evaluation Conference (LREC) (2012)
13. Ehrmann, Watter, Romanello, Clematide, Flückiger: Impresso Named Entity Annotation Guidelines (Jan 2020). https://doi.org/10.5281/zenodo.3604227, https://doi.org/10.5281/zenodo.3604227
14. Ehrmann, M., Hamdi, A., Pontes, E.L., Romanello, M., Doucet, A.: Named entity recognition and classification in historical documents: A survey. ACM Computing Surveys **56**(2), 1–47 (2023)
15. Ehrmann, M., Romanello, M., Clematide, S., Ströbel, P., Barman, R.: Language resources for historical newspapers: the impresso collection (2020)
16. Ehrmann, M., Romanello, M., Flückiger, A., Clematide, S.: Extended overview of clef hipe 2020: named entity processing on historical newspapers. In: CEUR Workshop Proceedings. No. 2696, CEUR-WS (2020)
17. Finkel, J.R., Grenager, T., Manning, C.D.: Incorporating non-local information into information extraction systems by gibbs sampling. In: Proceedings of the 43rd annual meeting of the association for computational linguistics (ACL'05). pp. 363–370 (2005)
18. González-Gallardo, C.E., Boros, E., Giamphy, E., Hamdi, A., Moreno, J.G., Doucet, A.: Injecting temporal-aware knowledge in historical named entity recognition. In: Kamps, J., Goeuriot, L., Crestani, F., Maistro, M., Joho, H., Davis, B., Gurrin, C., Kruschwitz, U., Caputo, A. (eds.) Advances in Information Retrieval. pp. 377–393. Springer Nature Switzerland, Cham (2023)
19. González-Gallardo, C.E., Boros, E., Girdhar, N., Hamdi, A., Moreno, J., Doucet, A.: Oui mais... chatgpt peut-il identifier des entités dans des documents historiques? In: 18e Conférence en Recherche d'Information et Applications\\16e Rencontres Jeunes Chercheurs en RI\\30e Conférence sur le Traitement Automatique des Langues Naturelles\\25e Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues. pp. 74–82. ATALA (2023)
20. González-Gallardo, C.E., Boros, E., Girdhar, N., Hamdi, A., Moreno, J.G., Doucet, A.: Yes but.. can chatgpt identify entities in historical documents? In: 2023 ACM/IEEE Joint Conference on Digital Libraries (JCDL). pp. 184–189 (2023). https://doi.org/10.1109/JCDL57899.2023.00034
21. Grover, C., Givon, S., Tobin, R., Ball, J.: Named entity recognition for digitised historical texts. In: LREC. Citeseer (2008)
22. Guu, K., Lee, K., Tung, Z., Pasupat, P., Chang, M.: Retrieval augmented language model pre-training. In: International conference on machine learning. pp. 3929–3938. PMLR (2020)

23. Hamdi, A., Linhares Pontes, E., Boros, E., Nguyen, T.T.H., Hackl, G., Moreno, J.G., Doucet, A.: A multilingual dataset for named entity recognition, entity linking and stance detection in historical newspapers. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 2328–2334 (2021)
24. Hamdi, A., Pontes, E.L., Sidere, N., Coustaty, M., Doucet, A.: In-depth analysis of the impact of ocr errors on named entity recognition and linking. Natural Language Engineering pp. 1–24 (2022)
25. Hanh, T.T.H., Doucet, A., Sidere, N., Moreno, J.G., Pollak, S.: Named entity recognition architecture combining contextual and global features. In: International Conference on Asian Digital Libraries. pp. 264–276. Springer (2021)
26. Hubková, H., Král, P., Pettersson, E.: Czech historical named entity corpus v 1.0. In: Proceedings of the Twelfth Language Resources and Evaluation Conference. pp. 4458–4465 (2020)
27. Jiang, A.Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D.S., Casas, D.d.l., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., et al.: Mistral 7b. arXiv preprint arXiv:2310.06825 (2023)
28. Kim, S., Seo, K., Chae, H., Yeo, J., Lee, D.: Verifiner: Verification-augmented ner via knowledge-grounded reasoning with large language models (2024)
29. Kogkitsidou, E., Gambette, P.: Normalisation of 16th and 17th century texts in french and geographical named entity recognition. In: Proceedings of the 4th ACM SIGSPATIAL Workshop on Geospatial Humanities. pp. 28–34 (2020)
30. Linhares Pontes, E., Cabrera-Diego, L.A., Moreno, J.G., Boros, E., Hamdi, A., Doucet, A., Sidere, N., Coustaty, M.: Melhissa: a multilingual entity linking architecture for historical press articles. International Journal on Digital Libraries **23**(2), 133–160 (2022)
31. Moncla, L., Gaio, M., Joliveau, T., Lay, Y.F.L.: Automated geoparsing of paris street names in 19th century novels. In: Proceedings of the 1st ACM SIGSPATIAL Workshop on Geospatial Humanities. pp. 1–8 (2017)
32. Nguyen, N.K., Boros, E., Lejeune, G., Doucet, A.: Impact analysis of document digitization on event extraction. In: 4th workshop on natural language for artificial intelligence (NL4AI 2020) co-located with the 19th international conference of the Italian Association for artificial intelligence (AI* IA 2020). vol. 2735, pp. 17–28 (2020)
33. Nissim, M., Matheson, C., Reid, J., et al.: Recognising geographical entities in scottish historical documents. In: Proceedings of the Workshop on Geographic Information Retrieval at SIGIR 2004. vol. 35 (2004)
34. Packer, T.L., Lutes, J.F., Stewart, A.P., Embley, D.W., Ringger, E.K., Seppi, K.D., Jensen, L.S.: Extracting person names from diverse and noisy ocr text. In: Proceedings of the fourth workshop on Analytics for noisy unstructured text data. pp. 19–26 (2010)
35. Passaro, L., Lenci, A.: Il piave mormorava...: recognizing locations and other named entities in italian texts on the great war. Il Piave mormorava...: recognizing locations and other named entities in Italian texts on the Great War. pp. 286–290 (2014)
36. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. The Journal of Machine Learning Research **21**(1), 5485–5551 (2020)
37. Ritze, D., Zirn, C., Greenstreet, C., Eckert, K., Ponzetto, S.P.: Named entities in court: The marinelives corpus. In: Language Resources and Technologies for Pro-

cessing and Linking Historical Documents and Archives-Deploying Linked Open Data in Cultural Heritage–LRT4HDA Workshop Programme. p. 26 (2014)
38. Roberts, A., Raffel, C., Shazeer, N.: How much knowledge can you pack into the parameters of a language model? arXiv preprint arXiv:2002.08910 (2020)
39. Rodrigues Alves, D., Colavizza, G., Kaplan, F.: Deep reference mining from scholarly literature in the arts and humanities. Frontiers in Research Metrics and Analytics p. 21 (2018)
40. Rodriquez, K.J., Bryant, M., Blanke, T., Luszczynska, M.: Comparison of named entity recognition tools for raw ocr text. In: Konvens. pp. 410–414 (2012)
41. Romanello, M., Najem-Meyer, S., Robertson, B.: Optical character recognition of 19th century classical commentaries: the current state of affairs. In: The 6th International Workshop on Historical Document Imaging and Processing. pp. 1–6 (2021)
42. Ruokolainen, T., Kettunen, K.: À la recherche du nom perdu–searching for named entities with stanford ner in a finnish historical newspaper and journal collection. In: 13th IAPR International Workshop on Document Analysis Systems. pp. 1–2 (2018)
43. Sainz, O., García-Ferrero, I., Agerri, R., de Lacalle, O.L., Rigau, G., Agirre, E.: GoLLIE: Annotation guidelines improve zero-shot information-extraction. In: The Twelfth International Conference on Learning Representations (2024), https://openreview.net/forum?id=Y3wpuxd7u9
44. Shin, S., Lee, S.W., Ahn, H., Kim, S., Kim, H., Kim, B., Cho, K., Lee, G., Park, W., Ha, J.W., Sung, N.: On the effect of pretraining corpora on in-context learning by a large-scale language model. In: Carpuat, M., de Marneffe, M.C., Meza Ruiz, I.V. (eds.) Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 5168–5186. Association for Computational Linguistics, Seattle, United States (Jul 2022). https://doi.org/10.18653/v1/2022.naacl-main.380, https://aclanthology.org/2022.naacl-main.380
45. Sprugnoli, R., et al.: Arretium or arezzo? a neural approach to the identification of place names in historical texts. In: Proceedings of the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018). pp. 360–365. aAccademia University Press (2018)
46. Suárez, P.J.O., Dupont, Y., Lejeune, G., Tian, T.: Sinner@ clef-hipe2020: Sinful adaptation of sota models for named entity recognition in french and german. In: CLEF 2020 Working Notes. Working Notes of CLEF 2020-Conference and Labs of the Evaluation Forum (2020)
47. Swaileh, W., Paquet, T., Adam, S., Rojas Camacho, A.: A named entity extraction system for historical financial data. In: Document Analysis Systems: 14th IAPR International Workshop, DAS 2020, Wuhan, China, July 26–29, 2020, Proceedings 14. pp. 324–340. Springer (2020)
48. Terras, M.M.: The Rise of Digitization, pp. 3–20. SensePublishers, Rotterdam (2011). https://doi.org/10.1007/978-94-6091-299-3_1, https://doi.org/10.1007/978-94-6091-299-3_1
49. Wang, X., Zhou, W., Zu, C., Xia, H., Chen, T., Zhang, Y., Zheng, R., Ye, J., Zhang, Q., Gui, T., Kang, J., Yang, J., Li, S., Du, C.: Instructuie: Multi-task instruction tuning for unified information extraction (2023)
50. Wei, J., Wang, X., Schuurmans, D., Bosma, M., ichter, b., Xia, F., Chi, E., Le, Q.V., Zhou, D.: Chain-of-thought prompting elicits reasoning in large language models. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.)

Advances in Neural Information Processing Systems. vol. 35, pp. 24824–24837. Curran Associates, Inc. (2022), https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf
51. Won, M., Murrieta-Flores, P., Martins, B.: ensemble named entity recognition (ner): evaluating ner tools in the identification of place names in historical corpora. Frontiers in Digital Humanities **5**, 2 (2018)
52. Yu, P., Wang, X.: Bert-based named entity recognition in chinese twenty-four histories. In: International Conference on Web Information Systems and Applications. pp. 289–301. Springer (2020)