Comparing NER approaches on French clinical text, with easy-to-reuse pipelines

Thibault Hubert, Ghislain Vaillant, Olivier Birot, Camila Arias, Antoine Neuraz, Adrien Coulet

Comparing NER approaches on French clinical text, with easy-to-reuse pipelines

Thibault HUBERT [a,b] , Ghislain VAILLANT [a,b] , Olivier BIROT [a,b] , Camila ARIAS [a,b] , Antoine NEURAZ [a,b] and Adrien COULET [a,b,1]
[a] *Inria, HeKA, PariSanté Campus, Paris, France;* [b] *Centre de Recherche des Cordeliers, Inserm, Université Paris Cité, Sorbonne Université, France*
ORCiD ID: Antoine NEURAZ https://orcid.org/0000-0001-7142-6728, Adrien COULET https://orcid.org/0000-0002-1466-062X

Abstract. The task of Named Entity Recognition (NER) is central for leveraging the content of clinical texts in observational studies. Indeed, texts contain a large part of the information available in Electronic Health Records (EHRs). However, clinical texts are highly heterogeneous between healthcare services and institutions, between countries and languages, making it hard to predict how existing tools may perform on a particular corpus. We compared four NER approaches on three French corpora and share our benchmarking pipeline in an open and easy-to-reuse manner, using the medkit Python library. We include in our pipelines fine-tuning operations with either one or several of the considered corpora. Our results illustrate the expected superiority of language models over a dictionary-based approach, and question the necessity of refining models already trained on biomedical texts. Beyond benchmarking, we believe sharing reusable and customizable pipelines for comparing fast-evolving Natural Language Processing (NLP) tools is a valuable contribution, since clinical texts themselves can hardly be shared for privacy concerns.

Keywords. Clinical texts, Named Entity Recognition, Benchmark, Open science

1. Introduction

Named Entity Recognition (NER) is a central task in Natural Language Processing (NLP), which identifies specific type of entities in text, such as the mention of persons and locations in the general domain, or more specifically, drugs and disorders in biomedicine. This task is of particular importance for observational studies based on the secondary analysis of healthcare data. Indeed, clinical texts (e.g., exam reports, discharge summaries) available in Electronic Health Records contain a large part of patient clinical information that is not available elsewhere. This is why patient inclusion/exclusion, or the extraction of patient outcomes and covariates for these studies often necessitates to take text into consideration. Information extraction tasks, such as NER, are complex with clinical text, because their form and content are highly heterogeneous between healthcare services and institutions, between countries and languages, making hard to predict how existing tools will perform on a particular corpus. We compared four NER approaches over three manually annotated corpora in French, including a dictionary-based approach,

[1] Corresponding Author: Adrien COULET, adrien.coulet@inria.fr

two transformers, and a generative approach. In addition to the results of the benchmark, we share the programmatic pipelines as open source to facilitate reuse, adjustment and comparison of NER tools to other clinical corpora. These pipelines rely on medkit [1], an open source Python library specifically designed for this purpose.

2. Material and Methods

We consider three corpora, which content is similar to some extent to clinical text. Table 1 shows an overview of their content in term of size, annotations and split in three sets.

QUAERO corpus [2] is composed of MEDLINE article titles and European Medicines Agency drug inserts manually annotated with semantic groups of the Unified Medical Language System (UMLS) [3], which are mainly Chemical and Drugs, Disorder, Procedures, plus others (Anatomy, etc.).

CASM2 corpus is composed of clinical cases of the CAS corpus [4], annotated collaboratively by master students from the Université Paris Cité, with the following types of entities: problem, test and treatment. We align these types with UMLS semantic groups to ensure the compliance of corpora annotations.

E3C corpus [5] gathers clinical cases in five languages, with different annotation methods. Here, only the French and manually annotated subset is used. The only type of entity in this corpus is Disorder, which we align with the Disorder UMLS semantic group.

An entity matcher analyzes a text and automatically annotates it with chosen labels, given a certain algorithm. The medkit library provides a unified interface to manipulate several of these matchers, hereby simplifying their comparison at scale.

UMLS matcher is a similarity-based fuzzy matcher [6] that uses a dictionary based on the UMLS, and its semantic groups previously presented. In this paper, we used a similarity threshold of 0.9 and lowered case of dictionary terms and the text to annotate.

Table 1. Overview of the corpora and their annotated entities. MLS stands for mean sentence length

	QUAERO			**CASM2**			**E3C**		
	train	**val**	**test**	**train**	**val**	**test**	**train**	**val**	**test**
Documents	844	844	848	424	106	133	36	18	45
Sentences	1569	1514	1454	6320	1613	2173	509	241	642
MSL (char.)	96	92	93	140	140	136	146	149	139
Entities (#)	**4513**	**4121**	**4084**	**14566**	**3628**	**4744**	**596**	**272**	**731**
Disorder (%)	29	25	24	48	46	47	100	100	100
Chemical (%)	21	25	25	23	26	23			
Procedure (%)	20	18	19	29	28	29			
Others (%)	30	32	32						

Two **BERT-based matchers** were considered, both of which are pre-trained RoBERTa based models that were fine-tuned on a NER task to assess their respective efficiency. Indeed, language models can be first massively pre-trained for a global purpose and then lightly fine-tuned on specific NLP tasks or types of text. The first one is **DrBERT** [7], which is pre-trained on an open source corpus of French medical crawled texts called NACHOS. Among its various versions associated with different number of parameters, we use the 4GB one. The second model is **CamemBERT-bio** [8], which is built by continual-pretraining from CamemBERT-base, on three corpora (417 million words).

GPT matcher uses the ChatGPT 3.5-turbo conversational model through the spaCy-llm library [9]. This library, along with prompting, enables NER by taking an annotation schema, i.e., types of entities and their definition in natural language, and a text to annotate as inputs. Additional context may be provided by feeding examples of annotated texts to the model, following the chain of thoughts principle [10]. We provided twelve manually annotated sentences of the QUAERO train set.

We composed reusable pipelines by defining and chaining operations with the medkit library. Operations can be seen as nodes of a pipeline, including input and output data management, and operations that process data by either encapsulating external tools, such as an NER tool developed by a third part, or a model shared by Hugging Face or spaCy. We distinguish three pipelines: preprocessing, training and evaluation pipelines.

Preprocessing pipeline consists in three operations. The first consists in converting the corpus from its source format (XML, Brat and JSONL) to the internal representation used by medkit; the second filters out overlapping entities, as the UMLS and BERT-based matchers do not support them (shorter annotations are excluded); the third performs sentence tokenization to facilitate training and evaluation. Fig. 1 represents the chaining of these operations.

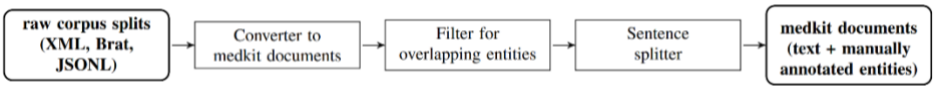

Figure 1. Preprocessing pipeline.

Training pipelines These pipelines were only used for fine-tuning the BERT-based matchers. Medkit provides facilities for easy training or fine-tuning any token classification model available on Hugging Face. For NER tasks, medkit takes a predefined set of types of entities and a training set with annotations as inputs. We distinguish two different pipelines for the fine-tuning of the BERT-based matchers. The first version, that we named specific, uses the train set of only one corpora; whereas the second version, named general, uses the aggregation of the train sets of the three corpora.

Evaluation pipeline A tenth of the original corpus is randomly sampled, and then annotated by one of the entity matcher. Predicted entities (i.e., the output of the matcher), alongside the original entities (i.e., the manual annotations from the corpus), are converted into tags using the IOB2 scheme (Inside, Outside, Beginning, version 2). Next, evaluation metrics such as the F1 score are computed with the seqeval library [11]. Fig. 2 presents the evaluation process of each NER tool. F1 scores are computed for each annotation type as a global weighted F1, averaged by the number of labels present in the test set. The specific "not in any chunk" label (i.e., the token tagged as “O”, meaning it is not associated with any annotation) is excluded from the weighted F1, therefore decreasing the final score in comparison to other authors.

Implementations of these pipelines, along with their documentation are available for reuse at https://medkit.readthedocs.io/en/stable/cookbook/ner_benchmark/.

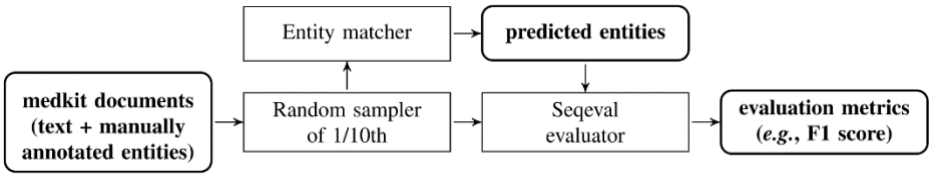

Figure 2. Evaluation pipeline.

3. Results

The preprocessing pipeline was run for each split (train, test and eval) of each corpus. The specific and general training pipeline was run once per corpus for each of the two BERT-based matchers, resulting in three specific fine-tuned models (one per corpus) and one general fine-tuned model (fine-tuned with all three corpora) for each of the two BERT-based matchers. The evaluation pipeline was run for every entity matcher, namely the UMLS matcher, the specifc vs. general versions for the BERT-based matchers and the GPT matcher on every corpus. The execution of the evaluation pipeline was repeated 10 times, except for GPT matcher (3 times) to avoid excessive cost. Table 2 reports the weighted F1 scores for each entity matcher on every corpus test set, and for each matcher the mean weighted by corpus sizes (last column).

Table 2. Weighted F1 scores per entity matcher and corpus test set.

	QUAERO	CASM2	E3C	Mean
UMLS matcher	0.48 ± 0.02	0.31 ± 0.02	**0.61 ± 0.03**	0.41
DrBERT specific	0.57 ± 0.02	0.57 ± 0.0	0.56 ± 0.02	0,57
DrBERT general	0.44 ± 0.02	0.42 ± 0.02	0.43 ± 0.04	0.43
CamemBERT-bio specific	0.42 ± 0.01	0.41 ± 0.02	0.4 ± 0.04	0.41
CamemBERT-bio general	**0.59 ± 0.02**	**0.58 ± 0.01**	0.52 ± 0.04	**0.58**
GPT matcher	0.52 ± 0.03	0.34 ± 0.03	0.55 ± 0.04	0.43

Overall, the general version of CamemBERT-bio achieved the best F1 with a mean over the 3 corpus of 0.58. It also achieved best F1 on the QUAERO and CASM2 corpora with F1 score of 0.59 ± 0.02 and 0.58 ± 0.01, respectively. The UMLS matcher, with its similarity-based fuzzy method, achieved the best performance on the E3C corpus with a F1 score of 0.61 ± 0.03, even though it performs relatively weakly overall with a mean F1 of 0.41. The GPT matcher performed irregularly depending on the corpora, with a mean F1 of 0.43. Table 3 shows detailed F1 scores for the 3 most frequent types of annotation in QUAERO and CASM2, namely Chemical, Disorder and Procedure. BERTbased matcher specifically fine-tuned on only one corpus are discarded for simplicity. CamemBERT-bio performs the best for each type of annotation except for QUAERO Disorders, where the other 3 performed equally.

Table 3. F1 scores per type of annotation on QUAERO and CASM2 corpora. Note that E3C has a unique type of annotation and for this reason, its performance per its unique type is the one reported in Table 2

	QUAERO	CASM2	QUAERO	CASM2	QUAERO	CASM2
UMLS matcher	0.55 ± 0.06	0.35 ± 0.05	**0.58 ± 0.05**	0.32 ± 0.04	0.29 ± 0.06	0.25 ± 0.05
DrBERT general	0.67 ± 0.05	0.43 ± 0.06	**0.58 ± 0.03**	0.4 ± 0.03	0.58 ± 0.04	0.45 ± 0.05
CamemBERT-bio gen.	**0.69 ± 0.04**	**0.66 ± 0.07**	0.55 ± 0.05	**0.61 ± 0.04**	**0.6 ± 0.08**	**0.67 ± 0.03**
GPT matcher	0.62 ± 0.04	0.33 ± 0.05	**0.58 ± 0.03**	0.42 ± 0.04	0.4 ± 0.02	0.24 ± 0.02

4. Discussion

First, we do not observe large differences between performances of BERT-based matchers when they are fine-tuned with larger and more diverse corpora (general setting) instead of single corpora (specific setting). This might be explained by the fact that both DrBERT and CamemBERT-bio models already "saw" large and diverse sets of biomedical texts, what may explain a saturation phenomena. Also, in this experiment, CamemBERTbio has higher performances than DrBERT which is consistent with results from other studies [12]. Here our rather naive reuse of available models may explain the absolute value difference. A shared observation with other authors is the advantage for NER of using masked language (i.e., BERT-based matchers), over Large Language Models (LLM) and prompting. We moderate this observation by the fact that we considered only one LLM, that is not the latest, and a rather simple prompting approach. It would be interesting to see what happens if more and more diverse pre-prompt examples were provided to the LLM. Also we note that the dictionary-based fuzzy matching approaches of the UMLS matcher obtain the best performance with E3C and its single type of annotations (Disorders). We observe a diversity and non consistent trends of results over the three considered corpora, making hard to be conclusive on several aspects (e.g., results on E3C or with QUAERO Disorders inconsistent with the others). This underlines our initial claims about the heterogeneity of clinical texts and the need for tools for flexible evaluation of NLP approaches, even for NLP non-experts. To this aim, the pipelines we share are easy to reproduce on various environments and easy to enrich with new corpora or entity matchers.

References

[1] medkit repository. GitHub; 2024. https://github.com/medkit-lib/medkit.

[2] Névéol A, Grouin C, Leixa J, Rosset S, Zweigenbaum P. The QUAERO French Medical Corpus: A Ressource for Medical Entity Recognition and Normalization. In: Proc of BioTextMining Work; 2014. p. 24-30.

[3] McCray AT, Burgun A, Bodenreider O. Aggregating UMLS semantic types for reducing conceptual complexity. Studies in health technology and informatics. 2001;84(0 1):216.

[4] Grabar N, Dalloux C, Claveau V. CAS: corpus of clinical cases in French. Journal of Biomedical Semantics. 2020;11(1):1-10.

[5] Magnini B, Altuna B, Lavelli A, Speranza M, Zanoli R. The E3C project: European clinical case corpus. Language. 2021;1(L2):L3.

[6] Okazaki N, Tsujii J. Simple and Efficient Algorithm for Approximate Dictionary Matching. In: Proceedings of the 23rd International Conference COLING 2010; 2010. p. 851-9.

[7] Labrak Y, Bazoge A, Dufour R, Rouvier M, Morin E, Daille B, et al. DrBERT: A Robust Pre-trained Model in French for Biomedical and Clinical domains. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics; 2023. p. 16207-21.

[8] Touchent R, Romary L, de La Clergerie E. CamemBERT-bio: a Tasty French Language Model Better for your Health. arXiv preprint arXiv:230615550. 2023.

[9] SpaCy-LLM repository. GitHub; 2024. https://github.com/explosion/spacy-llm.

[10] Ashok D, Lipton ZC. PromptNER: Prompting For Named Entity Recognition. arXiv preprint arXiv:230515444. 2023.

[11] Nakayama H. seqeval: A python framework for sequence labeling evaluation. Software available from https://github com/chakki-works/seqeval. 2018.

[12] Naguib M, Tannier X, Névéol A. Few shot clinical entity recognition in three languages: Masked language models outperform LLM prompting; 2024.