Ertim at SemEval-2023 Task 2: Fine-tuning of Transformer Language Models and External Knowledge Leveraging for NER in Farsi, English, French and Chinese

Kévin Deturck, Pierre Magistry,
Bénédicte Diot-Parvaz Ahmad, Ilaine Wang, Damien Nouvel
ERTIM Inalco / 2 rue de Lille, 75007 Paris
Hugo Lafayette
Kairntech / 29 Chemin du Vieux Chêne, 38240 Meylan
first.last@{inalco,kairntech}.fr

Abstract

Transformer language models are now a solid baseline for Named Entity Recognition and can be significantly improved by leveraging complementary resources, either by integrating external knowledge or by annotating additional data. In a preliminary step, this work presents experiments on fine-tuning transformer models. Then, a set of experiments has been conducted with a Wikipedia-based reclassification system. Additionally, we conducted a small annotation campaign on the Farsi language to evaluate the impact of additional data. These two methods with complementary resources showed improvements compared to fine-tuning only.

1 Introduction

Entity recognition and linking has now become a standard task for NLP, either as a goal in itself or as a preprocessing step for other goals. As for many other NLP tasks, developing an accurate model for a specific dataset (by language or domain) is still a challenge that heavily relies on resources, either pre-computed in a language model or as an external dataset.

As the NLP research lab of Inalco, Ertim was particularly interested in MultiCoNER II because it gave us a good opportunity to:

- carry out a novel evaluation of HuggingFace models on the complex NER task proposed in MultiCoNER II, through the diversity of more or less endowed languages (Fetahu et al., 2023a)
- experiment the integration of external knowledge as a postprocessing step to improve the classification of named entities
- conduct experiments on NER in Farsi, which coincided with our work for a project (VITAL, see the Acknowledgements section)

We report our official results (Fetahu et al., 2023b) in Table 1. Due to lack of time, some of our experiments could not be submitted as official runs, this paper provides additional information and results about these experiments.

Lang.	Rank	F1 clean	F1 noisy	F1 overall
EN	20/34	61.85	52.78	59.03
FR	9/17	69.73	58.6	66.3
FA	12/14	53.77	-	53.77
ZH	11/22	64.26	44.38	59.45

Table 1: Ertim's rankings for the 4 tracks undertaken

2 Fine-tuning of HuggingFace Language Models

We present experiments based on the fine-tuning of transformer language models for the MultiCoNER II task on specific languages.

2.1 Official runs

In this section, we present our runs based on the fine-tuning of HuggingFace language models that have been officially retained for the MultiCoNER II final rankings. All these runs have been done by using the Spacy framework. We used models respectively pre-trained on our target languages and fine-tuned them by using the MultiCoNER II `train` and `dev` datasets in the corresponding languages.

HuggingFace Model	Prec.	Rec.	F1
roberta-base (Liu et al., 2019)	59.13	59.54	59.03

Table 2: "en" track, fine-grained macro average performance on test dataset

We logically found out two classes of results: the best ones are on highly endowed languages (English and French) whereas the lowest ones are on the less supported languages (Chinese and Farsi).

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 2211–2215

HuggingFace Model	Prec.	Rec.	F1
flaubert-large-cased (Le et al., 2020)	65.03	64.58	63.75
flaubert-base-uncased (Le et al., 2020)	67.44	66.08	66.30

Table 3: "fr" track, fine-grained macro average performance on test dataset

HuggingFace Model	Prec.	Rec.	F1
bert-base-chinese (Devlin et al., 2019)	56.95	52.71	53.14

Table 4: "zh" track, fine-grained macro average performance on test dataset

HuggingFace Model	Prec.	Rec.	F1
bert-base-parsbert-uncased (Farahani et al., 2021)	51.56	58.53	53.77

Table 5: "fa" track, fine-grained macro average performance on test dataset

Comparing French and English, the best performance on French may be explained by the volume and diversity of the corpus used for pre-training the Flaubert models, which are Bert models for French, like `roberta-base` for English.

It is also interesting to note the crucial role of case in the datasets used for training and evaluation. The particularity of the datasets provided for MultiCoNER is that all corpora were uncased, including those of languages with writing systems that distinguish lowercase and uppercase letters. In Table 3, we can see that the `flaubert-base-uncased` model outperforms `flaubert-large-cased`; this is remarkable considering that the uncased version of flaubert has a much smaller training corpus.

2.2 Additional experiments

We conducted additional experiments on the fine-tuning of HuggingFace language models by using the MultiCoNER II `train` and `dev` sets. These experiments were performed independently of the official runs, by using HuggingFace's `transformers` library. It required aligning labels with `offset_mapping` and checking the BIO format. The models used are listed in Table 6. Some of them are multilingual and were tested on a specific language.

Figure 1 reports F1 results by distinguishing languages among line charts and models by different plots in each of them. Most models required more than five epochs to obtain the best F1 results, which are very different among languages. Dedicated language models (Bert for English, CamemBERT for French, ParsBERT for Farsi and `bert-base-chinese`) outperform multilingual ones by about 5pts of F1 score in English, 3pts in French, 10pts in Farsi and 30pts in Chinese.

All (inc. English)	distilbert-base-uncased bert-base-uncased bert-large-uncased xlm-roberta-large
French	camembert-base camembert-large (Martin et al., 2020)
Farsi	bert-base-parsbert-uncased (Farahani et al., 2021)

Table 6: Pre-computed HuggingFace models

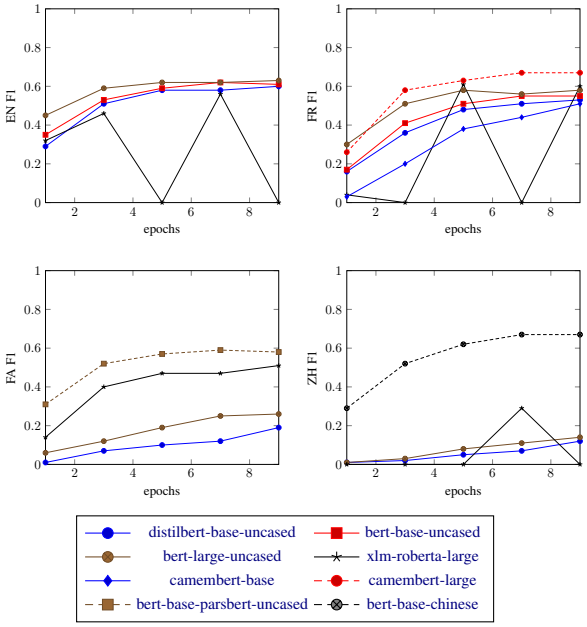

Figure 1: Results per language and epoch

3 Entity Classification Leveraging Wikipedia Articles

3.1 Method

We used as a resource the Wikipedia dump database made publicly available online by the Wikimedia Foundation[1]. We downloaded the dump containing only textual data, for example "wikipedia_zh_all_nopic", and then used it as a

[1] https://dumps.wikimedia.org/other/kiwix/zim/wikipedia/

local database, with the Kiwix[2] tool.

Assuming a mention has been detected, the aim is to create a model of the categories to be predicted from the Wikipedia resources. Our approach consists in training a classifier based on the TF-IDF representation of word bigrams from the first paragraph and the descriptive field headings of Wikipedia pages. This representation is mapped with entity categories in both `train` and `dev` datasets.

The classification system is implemented as a Keras model with two input layers and one output layer. The input layers are dense with `relu` activation of 256 neurons and a dropout of 0.5. The output layer is dense with `softmax` activation. Once we have generated the Wikipedia-based model, we can apply it in order to perform a reclassification of the named entities previously identified by fine-tuned transformer models.

For each named entity detected, we send a query to the Kiwix server to obtain the corresponding Wikipedia page. In cases in which there is no Wikipedia page, we do not modify the previous category. Thus, our approach does not detect new named entities, the goal of this Wikipedia-based system is only to improve a previous categorisation.

If a Wikipedia page is available for the named entity to be reclassified, we generate a representation of the named entity, similar to the ones of the pages used for training the model. Then, we categorise the named entity with the class predicted by the classification model.

3.2 Official Results

Nb. of page per entity	Precision	Recall	F1
1	61.72	58.94	59.45
3	62.09	58.49	59.20

Table 7: Fine-grained macro average performance on Chinese test dataset with our Wikipedia-based reclassification system and the `bert-base-chinese` model

With our Wikipedia-based reclassifier system, we submitted two runs on Chinese using the `bert-base-chinese` transformer model that we previously tested alone. The difference between the two runs is the number of Wikipedia pages included for each entity: for the first run, we only use the first page, for the second run, we use the first three pages in order to bypass the possible disambiguation page.

We found out that the two tests with the reclassification system significantly improve the performance compared to using the `bert-base-chinese` model alone (see section 2.2). Also, taking into account more Wikipedia pages slightly improves the precision, which was the goal since we wanted to avoid the disambiguation page that could lead to a false categorisation. However, we have to counteract the decline in recall induced by this method.

4 Annotation of an Additional Farsi Dataset

4.1 Creation of Additional Farsi Annotations

The Farsi language is a relatively poorly endowed language, especially in comparison with English and French. The goal motivating this part of our work is to evaluate the impact of additional annotated data on the performance of the best model we obtained by using only the MultiCoNER II dataset, i.e. `bert-baseparsbert-uncased` (see section 2.2).

We recruited a Farsi speaker as an annotator. She worked on a news dataset in Farsi including more than 10 news agencies from 2009 to 2021 (Alimoradi, 2021). We asked the annotator to work with all of the 36 annotation categories proposed for the MultiCoNER II task. We used an annotation platform developed by Kairntech (Nibart, 2022).

As we did not have guidelines regarding the categories, we provided the annotator with the MultiCoNER II Farsi `dev` file so that she could see and learn from annotation examples. We also provided the annotator with the annotation guidelines regarding some of the coarse-grain categories from the paper that described the previous MultiCoNER edition (Derczynski et al., 2017; Malmasi et al., 2022a,b).

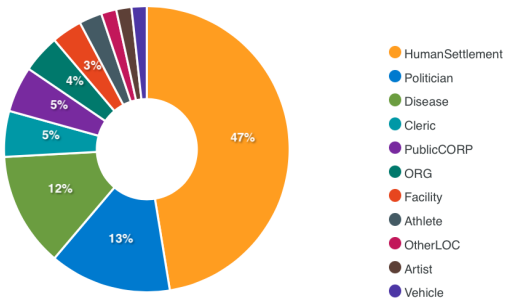

Figure 2: Category distribution from session 1

[2] https://www.kiwix.org

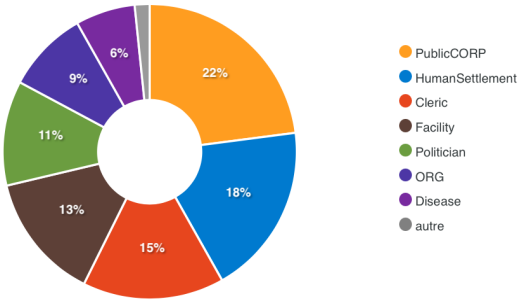

Figure 3: Category distribution from session 2

Two 50-minute annotation sessions were organised. As a result of the first session, the annotator produced 116 annotated entities from 10 documents. As a result of the second session, she produced 122 annotated entities from 9 documents. We present the category distribution in the additional annotations from session 1 in Figure 2 and session 2 in Figure 3.

We note that the categories' distribution between the two sessions are significantly different, which is positive for the categorical diversity of the additional annotations. In addition, only about one third of all proposed MultiCoNER II labels were identified in the new dataset, which leads us to hypothesise that the performance improvements with the additional annotations would be variable across the labels.

4.2 Experimental Results with the Additional Annotations

We carried out a first experiment which consisted in training `bert-base-parsbert-uncased` on the MultiCoNER II `train` set plus the additional annotations, and evaluating on the `dev` set. As we had previously assumed from the unbalanced distribution of categories in the additional annotations, we notice that the results are variable across categories.

We then decided to experiment on a restricted version of the additional annotations in order to filter out categories that performed less well. We have retained 15 categories out of 36: `PublicCorp`, `PrivateCorp`, `Politician`, `Facility`, `HumanSettlement`, `Athlete`, `MedicalProcedure`, `Software`, `AnatomicalStructure`, `Disease`, `ORG`, `OtherPROD`, `Artist`, `CarManufacturer`, and `Clothing`. The corresponding results are presented in Table 8.

HuggingFace model	Precision	Recall	F1
bert-base-parsbert-uncased	59.73	58.99	58.04

Table 8: Fine-grained macro average performance on "fa" test dataset with filtered additional annotations

The filtered additional annotations allowed to improve the F1 score by 4.2pts compared to training `bert-base-parsbert-uncased` only on the MultiCoNER II `train` set (see section 2.2). This indicates that even modest in terms of quantity and categories covered, additional annotations are significantly beneficial to the fine-tuning of a transformer Farsi model for NER. We consider that those improvements may be viewed as few-shot settings that leverage on the generalisation capacities of the underlying large language model.

5 Conclusion

Our work is based on three complementary directions: the fine-tuning of HuggingFace transformer language models, a reclassification system based on Wikipedia, and an effort to provide additional annotations. The first axis allowed us to highlight key results regarding the use of different models on a variety of languages. The second and third axes correspond to the original contributions of our work about integrating complementary data sources.

With our Wikipedia-based reclassification system, we found that using the information available in Wikipedia could correct the classification from a fine-tuned transformer model. We have shown the significant contribution of additional annotations to the fine-tuning of transformer language model, even if these annotations are modest in terms of volume and categorical coverage. We consider sharing as open-source the annotated data, the models produced and our experimental codes.

As a follow-up to this work, it would be interesting to evaluate the evolution of pre-trained model performance according to the quantity and variety of annotations we include. Also, our Wikipedia-based reclassification system has a lot of room for improvement, especially when it comes to handling mentions with no Wikipedia page available and cases of ambiguity. Overall, this work would be all the more interesting and robust when applied to more languages. We consider experimenting further with Spacy as it is a user-friendly wrapper, including Pytorch and HuggingFace components.

Acknowledgements

This work was conducted thanks to the DGA RAPID VITAL funding.

References

Saied Alimoradi. 2021. Persian News Dataset. https://saied71.github.io/RohanAiLab/.

Leon Derczynski, Eric Nichols, Marieke Van Erp, and Nut Limsopatham. 2017. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. pages 4171–4186.

Mehrdad Farahani, Mohammad Gharachorloo, Marzieh Farahani, and Mohammad Manthouri. 2021. Parsbert: Transformer-based Model for Persian Language Understanding. *Neural Processing Letters*, 53:3831–3847.

Besnik Fetahu, Zhiyu Chen, Sudipta Kar, Oleg Rokhlenko, and Shervin Malmasi. 2023a. MultiCoNER v2: a Large Multilingual dataset for Fine-grained and Noisy Named Entity Recognition.

Besnik Fetahu, Sudipta Kar, Zhiyu Chen, Oleg Rokhlenko, and Shervin Malmasi. 2023b. SemEval-2023 Task 2: Fine-grained Multilingual Named Entity Recognition (MultiCoNER 2). In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*. Association for Computational Linguistics.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoît Crabbé, Laurent Besacier, and Didier Schwab. 2020. FlauBERT: Unsupervised Language Model Pre-training for French. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 2479–2490, Marseille, France. European Language Resources Association.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. abs/1907.11692.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022a. MultiCoNER: A Large-scale Multilingual Dataset for Complex Named Entity Recognition. In *Proceedings of the 29th International Conference on Computational Linguistics*, Gyeongju, Republic of Korea.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022b. SemEval-2022 Task 11: Multilingual Complex Named Entity Recognition (MultiCoNER). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a Tasty French Language Model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Vincent Nibart. 2022. Découvrir et enrichir des connaissances à partir de l'analyse de documents grâce à l'intelligence artificielle: l'exemple de la plateforme kairntech. *I2D–Information, données & documents*, pages 38–43.