PAI at SemEval-2022 Task 11: Name Entity Recognition with Contextualized Entity Representations and Robust Loss Functions

Long Ma*, **Xiaorong Jian***,**Xuan Li**
Pingan Life Insurance of China, Ltd
{MALONG633,JIANXIAORONG974,LIXUAN208}@pingan.com.cn

Abstract

This paper describes our system used in the SemEval-2022 Task 11 Multilingual Complex Named Entity Recognition, achieving 3rd for track 1 on the leaderboard. We propose Dictionary-fused BERT, a flexible approach for entity dictionaries integration. The main ideas of our systems are: 1) integrating external knowledge (an entity dictionary) into pre-trained models to obtain contextualized word and entity representations 2) designing a robust loss function leveraging a logit matrix 3) adding an auxiliary task, which is an on-top binary classification to decide whether the token is a mention word or not, makes the main task easier to learn. It is worth noting that our system achieves an F1 of 0.914 in the post-evaluation stage by updating the entity dictionary to the one of Meng et al. (2021), which is higher than the score of 1st on the leaderboard of the evaluation stage.

1 Introduction

Name entity recognition (NER) is a fundamental task in natural language processing. Processing complex and ambiguous Named Entities (NEs) and in low-context situations is a challenging NLP task in practical and open-domain settings, which was recently outlined by Meng et al. (2021). Other work has extended this to multilingual and code-mixed settings (Fetahu et al., 2021) since code-mixed queries, with entities in a different language than the rest of the query, pose a particular challenge in domains like e-commerce (e.g. queries containing movie or product names).

SemEval-2022 task 11 Multilingual Complex Named Entity Recognition (Malmasi et al., 2022b) focuses on dealing with the challenges above: detecting complex entities in short, low-context, and code-mixed settings.

For this task, we propose Dictionary-fused BERT to integrate external dictionaries into the NER model, and it is compatible with emerging entities and user-defined entities, without retraining the model.

2 Related Work

Named Entity Recognition (NER) is a core task in knowledge extraction and is important to various downstream applications such as question answering and dialogue systems. NER is the task of detecting mentions of real-world entities from text and recognizing their types (e.g., locations, persons).

However, the NER task is facing many challenges outlined by Meng et al. (2021), such as short texts like search queries (Wang et al., 2014), emerging entities (Craswell et al., 2020), long-tail entities, complex entities, and entities in code-mixed queries (Fetahu et al., 2021). For example, complex NEs, like the titles of creative works (movie/book/song/software names) are not simple nouns and are harder to recognize. The neural models driven by memorization perform well on "easy" entities (person names) but fail to recognize complex/unseen entities when entities overlap less between train/test sets. A lot of works have been proposed to address the challenges above.

Contextualized Word and Entity Representations There are some works focusing on obtaining good contextual word and entity representations such as KnowBERT (He et al., 2019) and LUKE (Yamada et al., 2020). KnowBERT (He et al., 2019) incorporates knowledge bases into BERT (Devlin et al., 2018) using Knowledge attention and recontextualization, which explores the joint learning of entities and relations. LUKE (Yamada et al., 2020) employs RoBERTa (Delobelle et al., 2020) as the base pre-trained model, trained on a large number of entity-annotated corpora retrieved

* The first two authors contributed equally.

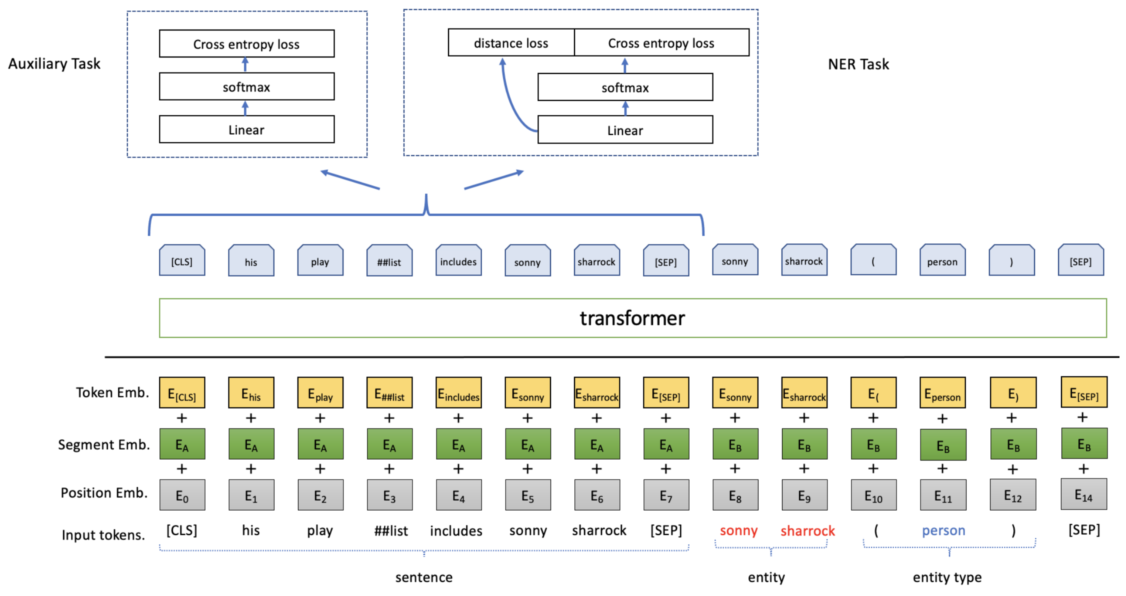

Figure 1: The overall architecture of our proposed system

from Wikipedia. LUKE treats both words and entities as separate tokens and computes the intermediate and output representations of all tokens through the Transformer. Since entities are also used as tokens, LUKE can model the relationship between entities.

Robust Loss Functions Other works like Generalized Cross-Entropy (Zhang and Sabuncu, 2018) and In-trust loss functions (Huang et al., 2021) focus on how to design a robust loss function to solve the NER problem under label noise, which is consistent with complex and ambiguous NER scenarios. Generalized Cross-Entropy (Zhang and Sabuncu, 2018) is actually a new evolutionary form of MAE and CCE, and can be easily applied with the DNN algorithm while yielding good performance in a lot of noisy label scenarios. In-trust loss functions (Huang et al., 2021) combines a CRF loss with a robust Distrust Cross-Entropy term and can effectively alleviate overfitting. What's more, it has been demonstrated that leveraging a logit matrix is an effective way to distinguish noisy samples from difficult samples.

Our proposed model combines the Contextualized-Entity-Representations method and the Robust-Loss-Functions method. Firstly our model incorporates the entity dictionary proposed by LUKE (Yamada et al., 2020) into the BERT-Whole-Word-Masking model and treats entities as separate tokens. Then we design a robust loss function that boosts cross-entropy loss with KL divergence over the logit matrix.

3 Data

We leverage LUKE entity dictionary (Yamada et al., 2020) since our model needs external knowledge. We train and test on MultiCoNER Dataset (Malmasi et al., 2022a).

MultiCoNER.Dataset for the SemEval-2022 Task 11, containing a training set of size 15300, a dev set of size 800, and a test set of size 217818, consisting of 6 entity types: PERSON (PER for short, names of people), LOCATION (LOC, locations/physical facilities), CORPORATION (CORP, corporations and businesses), GROUPS (GRP, all other groups), PRODUCT (PROD, consumer products), and CREATIVE-WORK (CW, movie/song/book/etc. titles). All data are uncased.

LUKE Entity Dictionary consists of 500k entities retrieved from English Wikipedia data.

4 Methodology

We propose Dictionary-fused BERT, which adopts a multi-layer bidirectional transformer (Vaswani et al., 2017) and has the ability to encode not only words of the sentence but also information of matched entities. We experiment with Transformer-

Prerained-model	**dev f1**
BERT_{base}	0.854
BERT_{large}	0.871
RoBERTa_{base}	0.836
RoBERTa_{large}	0.877
DistilBERT_{base}	0.835
BERT-WWM_{large}	**0.883**
LUKE_{base}	0.856
LUKE_{large}	0.878

Table 1: Dev f1 score of different Transformer-based models.

based models which produce contextualized word representations. The overall architecture is shown in Figure 1, the components are detailed below.

4.1 Contextualized Word and Entity Representation

Entity representations are obtained in two steps: entity matching and contextual encoding.

Dictionary Entity Matching We denote the input sentence as $(w_1, w_2..., w_n)$, where w_i is the i-th word and n is the Length. Full string matching is used to match all entities in the input sentence and we choose the longer one while dealing with overlapping matches. $(m_1, m_2, ..., m_L)$ represents the matched entities where $m_j = (w_i, w_{i+1}.., w_{i+k})$ and L is the number of matched entities.

Contextual Encoding The input tokens consist of two segments: words of the sentence and matched entities including the entity itself and entity type. The two parts are separated by a special token [SEP] and different entities are separated by the token $. Then they are fed together into the encoder to get the contextualized representations as LUKE(Yamada et al., 2020) does.

4.2 On-top Binary Classification Task

On-top binary classifier (adding a binary classifier on the top of the encoder) is an auxiliary task to detect entity mention words of the input sentence and helps increase accuracy. For example, the target output of the auxiliary task is represented as $y_{aux} = (0, 0, 0, 1, 0, 0, 1, 1)$ whereas the NER task's is $y_{ner} = (O, O, O, B-CW, O, O, B-PER, I-PER)$ when the input sentence is "adaptation of the manga series by chiho saito".

4.3 Loss Function

As is shown in the Figure 1, the loss is divided into two parts: the cross-entropy loss of the auxiliary task ($L_{CE-auxiliaty-task}$) and the loss of the NER task that consists of the final cross-entropy loss (L_{CE}) and KL divergence loss over the logit matrix (L_{logit}).

4.3.1 Loss over Logit Matrix

The logit matrix is the output of models in the training process, which goes through the softmax layer and then gets into the final loss function. Huang et al. (2021) has demonstrated the use of logit matrix to distinguish noisy samples from difficult samples to prevent overfitting, which we use here to distinguish tag pairs such as (CW, PROD) and (CORP, GRP) pairs that are easily confused with each other by baseline model. These samples are shown below.

* {they have been used experimentally in [gyrobus]{PROD}{CW}}. In this sample, [gyrobus] is a product (PROD) but is mispredicted as a creativework (CW).

* {cockpit concept, developed by [astion martin racing]{GRP}{CORP}}. [astion martin racing] is a group (GRP) but mispredicted as a corporation (CORP).

We denote the logit matrix as $[Z_{ij}]^{N\times M}$ where N is the length of input tokens and M is the number of tags. The loss over logit matrix can be represented as $L_{logit} = -(KLdiv(z_{CW}, z_{PROD}) + KLdiv(z_{GRP}, z_{CORP}))$ where z_{CW}, z_{PROD}, z_{GRP} and z_{CORP} are column vectors corresponding to tag CW, tag PROD, tag GRP and tag CORP in the logit matrix and $KLdiv$ is the Kullback-Leibler divergence loss function. The negative signal $(-)$ indicates that the larger distance between tag pairs the better.

4.3.2 Forming Total Loss

The total loss can be represented as

$$L_{total} = \alpha \cdot L_{CE-auxiliaty-task} + \beta \cdot (L_{CE} + L_{logit})$$

where α and β are two decoupled hyper parameters, α regulates the recall issue while β aims to flexibly explore the robustness of L_{logit}.

	settings	dev f1
baseline	BERT-WWM	0.883
	BERT-WWM$_{entity}$	0.90
ours	BERT-WWM$_{entity}$+auxiliary task	0.904
	BERT-WWM$_{entity}$+auxiliary task+L_{logit}	**0.913**

Table 2: results of ablation experiments

dictionary settings	dev f1	test f1
LUKE (Yamada et al., 2020)'s	0.913	0.784
GEMNET (Meng et al., 2021)'s	0.945	0.914

Table 3: performance of our model with injecting different entity dictionaries

5 Results

In this section, we verify the advantages of our system. On the one hand, we select the pre-trained BERT-Whole-Word-Masking model (BERT-WMM) (Liu et al., 2020) as our baseline model by comparing five different kinds of Transformer-based models. On the other hand, we analyze the effects of the way injecting an entity dictionary, adding an auxiliary task, and the loss function leveraging the logit matrix.

5.1 Baseline Experiments

The baseline performance of BERT (Devlin et al., 2018), BERT-WMM (Liu et al., 2020), DistilBERT (Sanh et al., 2019), RoBERTa (Delobelle et al., 2020) and LUKE(Yamada et al., 2020) is tested by directly fine-tuning the MultiCoNER data(Malmasi et al., 2022a) on the pre-trained LMs. All models are uncased. The results are shown in Table 1. BERT-WWM outperforms the others by 1 to 5 percentage points, so we choose BERT-WWM as our encoder. It may be due to the whole-word-masking mechanism adopted by BERT-WMM. Most hyperparameters are set as before, especially the learning rate is 2e-5, the batch size is 32, max sequence length is 100, max epochs is 20. All the performance data is on the development set.

5.2 Ablation Experiments

Results of ablation experiments are shown in Table 2. Our model Dictionary-fused BERT is represented as BERT-WWM$_{entity}$ + auxiliary task + L_{logit} and achieves an F1 of 0.913 on dev set when integrating LUKE entity dictionary into the model .

Effect of Integrating Entity Dictionaries As Malmasi et al. (2022b) says the MultiCoNER task aims to recognize entities in open-world settings that entities with tag CW and PROD try to mimic, which reminds us of levering external knowledge to deal with such issues. We treat sentence words and matched entities which are obtained by matching the LUKE entity dictionary and the original sentence as separated tokens like Yamada et al. (2020) does. This experiment is called BERT-WWM$_{entity}$ and improves by 1.7% over the no-entity BERT-WWM baseline.

Effect of Auxiliary Task We also explore Multi-Task learning by conducting the experiment of adding a binary classifier on top of the encoder, which predicts whether the token is a mention word or not. The auxiliary task is intended to increase the overall recall of mentions, which indeed brings a tiny improvement by 0.4% over BERT-WWM$_{entity}$.

Effect of L_{logit} As mentioned in the previous chapter, we propose L_{logit} using the logit matrix to distinguish some entities due to they are easily confused with each other. The total loss function boosting with L_{logit} improves by 0.9% . Finally, our model achieves 0.913 F1 on the dev set.

5.3 Evaluation Results

In the stage of evaluation, in order to fully utilize the training and dev sets, we corporate 10-fold cross-validation training trick and ensemble ten results to obtain the final one. Concretely, we divide the origin training and dev sets into 10 mutually exclusive subsets of the same size. Each time the union of 9 subsets is treated as the training set while the remaining subset is treated as the dev set. We do training and predicting 10 times to get 10 models and 10 results of the test set. The final result is a weighted vote of 10 results, where the weight is the F1 score of each label of each model.

We also compare the performance of our model injecting different entity dictionaries on dev and test sets. We incorporate the LUKE entity dictionary into our model in the evaluation stage. In the post-evaluation stage, by updating LUKE(Yamada et al., 2020) entity dictionary to the one of Meng et al. (2021), our system achieves an F1 of 0.914, which is higher than the score of 1st on the leaderboard of the evaluation stage. Results are shown in Table 3.

6 Conclusion

In this paper, we focused on how to integrate external dictionaries into NER models to deal with NER challenges in open-world settings and design a robust loss function to prevent overfitting. We proposed Dictionary-fused BERT, a flexible and simple approach that includes a contextualized entity representation encoder and a robust loss function leveraging a logit matrix. More importantly, our approach can infuse entity dictionaries into any Transformer-based pre-trained models and is compatible with any emerging entities and user-defined entities without retraining the model.

Although we have already succeeded in injecting entity information into Transformer-based models, entity spans are implicitly informed through a special token $. In the future, we will attempt to explore other ways to explicitly encode entity span information.

References

Nick Craswell, Daniel Campos, Bhaskar Mitra, Emine Yilmaz, and Bodo Billerbeck. 2020. Orcas: 18 million clicked query-document pairs for analyzing search.

Pieter Delobelle, Thomas Winters, and Bettina Berendt. 2020. Robbert: a dutch roberta-based language model. *arXiv preprint arXiv:2001.06286*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Besnik Fetahu, Anjie Fang, Oleg Rokhlenko, and Shervin Malmasi. 2021. Gazetteer Enhanced Named Entity Recognition for Code-Mixed Web Queries. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1677–1681.

B. He, D. Zhou, J. Xiao, X. Jiang, Q. Liu, N. J. Yuan, and T. Xu. 2019. Integrating graph contextualized knowledge into pre-trained language models.

Xiusheng Huang, Yubo Chen, Shun Wu, Jun Zhao, Yuantao Xie, and Weijian Sun. 2021. Named entity recognition via noise aware training mechanism with data filter. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4791–4803, Online. Association for Computational Linguistics.

Chao Liu, Cui Zhu, and Wenjun Zhu. 2020. Chinese named entity recognition based on bert with whole word masking. In *Proceedings of the 2020 6th International Conference on Computing and Artificial Intelligence*, pages 311–316.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022a. MultiCoNER: a Large-scale Multilingual dataset for Complex Named Entity Recognition.

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.

Tao Meng, Anjie Fang, Oleg Rokhlenko, and Shervin Malmasi. 2021. GEMNET: Effective gated gazetteer representations for recognizing complex entities in low-context input. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1499–1512.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Kuansan Wang, Evgeniy Gabrilovich, David Carmel, Bo June Hsu, and Ming Wei Chang. 2014. Erd'14: Entity recognition and disambiguation challenge. *ACM SIGIR forum*.

I. Yamada, A. Asai, H. Shindo, H. Takeda, and Y. Matsumoto. 2020. Luke: Deep contextualized entity representations with entity-aware self-attention.

Zhilu Zhang and Mert R. Sabuncu. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels.