DAMO-NLP at SemEval-2023 Task 2: A Unified Retrieval-augmented System for Multilingual Named Entity Recognition

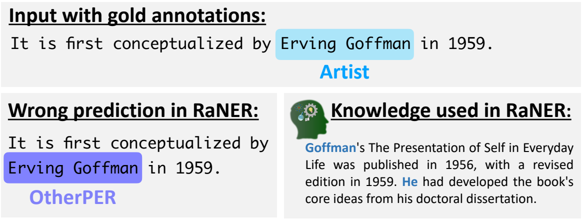
Zeqi Tan[☾†], Shen Huang[☆†], Zixia Jia[✿†], Jiong Cai[✿†], Yinghui Li[♡†], Weiming Lu[☾]
Yueting Zhuang[☾], Kewei Tu[✿], Pengjun Xie[☆], Fei Huang[☆], Yong Jiang[☆*]
[☆]DAMO Academy, Alibaba Group
[☾]College of Computer Science and Technology, Zhejiang University
[✿]School of Information Science and Technology, ShanghaiTech University
[♡]Tsinghua Shenzhen International Graduate School, Tsinghua University
{zqtan,yzhuang,luwm}@zju.edu.cn liyinghu20@mails.tsinghua.edu.cn
{jiazx,caijiong,tukw}@shanghaitech.edu.cn
{pangda,chengchen.xpj,f.huang,yongjiang.jy}@alibaba-inc.com

Abstract

The MultiCoNER II shared task aims to tackle multilingual named entity recognition (NER) in fine-grained and noisy scenarios, and it inherits the semantic ambiguity and low-context setting of the MultiCoNER I task. To cope with these problems, the previous top systems in the MultiCoNER I either incorporate the knowledge bases or gazetteers. However, they still suffer from insufficient knowledge, limited context length, single retrieval strategy. In this paper, our team **DAMO-NLP** proposes a unified retrieval-augmented system (U-RaNER) for fine-grained multilingual NER. We perform error analysis on the previous top systems and reveal that their performance bottleneck lies in insufficient knowledge. Also, we discover that the limited context length causes the retrieval knowledge to be invisible to the model. To enhance the retrieval context, we incorporate the entity-centric Wikidata knowledge base, while utilizing the infusion approach to broaden the contextual scope of the model. Also, we explore various search strategies and refine the quality of retrieval knowledge. Our system[1] wins 9 out of 13 tracks in the MultiCoNER II shared task. Additionally, we compared our system with ChatGPT, one of the large language models which have unlocked strong capabilities on many tasks. The results show that there is still much room for improvement for ChatGPT on the extraction task.

1 Introduction

The MultiCoNER series shared task (Malmasi et al., 2022b; Fetahu et al., 2023b) aims to identify complex named entities (NE), such as titles of creative works, which do not possess the traditional characteristics of named entities, such as persons, locations, etc. It is challenging to identify these ambiguous complex entities based on short contexts (Ashwini and Choi, 2014; Meng et al., 2021; Fetahu et al., 2022). The MultiCoNER I task (Malmasi et al., 2022b) focuses on the problem of semantic ambiguity and low context in multilingual named entity recognition (NER). In addition, the MultiCoNER II task (Fetahu et al., 2023b) this year poses two major new challenges: (1) a fine-grained entity taxonomy with 6 coarse-grained categories (`Location`, `Creative Work`, `Group`, `Person`, `Product` and `Medical`) and 33 fine-grained categories, and (2) simulated errors added to the test set to make the task more realistic and difficult, like the presence of spelling mistakes.



Input with gold annotations:

It is first conceptualized by Erving Goffman in 1959.

Wrong prediction in RaNER:

It is first conceptualized by Erving Goffman in 1959.

Knowledge used in RaNER:

Goffman's The Presentation of Self in Everyday Life was published in 1956, with a revised edition in 1959. He had developed the book's core ideas from his doctoral dissertation.

Figure 1: An example of wrong prediction in RaNER (Wang et al., 2022b) (one of the top systems in the MultiCoNER I task (Malmasi et al., 2022b)) . This case illustrates that the knowledge covered is not sufficient for fine-grained complex NER.

The previous top systems (Wang et al., 2022b; Chen et al., 2022) of the MultiCoNER I task incorporate additional knowledge in pre-trained language models, either a knowledge base or a gazetteer. RaNER (Wang et al., 2022b) builds a multilingual knowledge base based on Wikipedia and the original input sentences are then augmented with retrieved contexts from the knowl-

[*]: project lead. [†]: equal contributions.
This work was done during Zeqi Tan, Zixia Jia, Jiong Cai, and Yinghui Li's internship at DAMO Academy, Alibaba Group.

[1] We will release the dataset, code, and scripts of our system at https://github.com/modelscope/AdaSeq/tree/master/examples/U-RaNER.

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 2014–2028
July 13-14, 2023 ©2023 Association for Computational Linguistics

edge base, allowing the model to access more relevant knowledge. GAIN (Chen et al., 2022) proposes a gazetteer-adapted integration network with a gazetteer built from Wikidata to improve the performance of language models. Although these systems achieve impressive results, they still have some drawbacks. **First**, insufficient knowledge is a common problem. As shown in Figure 1, the knowledge used in RaNER can help the model to identify *Erving Goffman* as a person, but cannot further determine the fine-grained category `Artist`. **Second**, these methods mostly suffer from the limited context length. Wang et al. (2022b) discards stitched text that is longer than 512 after tokenizing, which means that plenty of retrieved context is not visible to the model, leading to resource waste. **Third**, these systems have a single retrieval strategy. Wang et al. (2022b) acquires knowledge by text retrieval, while Chen et al. (2022) accesses knowledge by dictionary matching. This single way of knowledge acquisition will result in the underutilization of knowledge.

To tackle these problems, we propose a unified retrieval-augmented system (U-RaNER) for fine-grained multilingual NER. We use both Wikipedia and Wikidata knowledge bases to build our retrieval module so that more diverse knowledge can be considered. As shown in Figure 1, if we locate the entry for *Erving Goffman* in Wikidata, we can make use of fine-grained entity category information to facilitate predictions. Also, we discover that the retrieval context dropped by the model may also contain useful knowledge. Thus, we explore the infusion approach to make more context visible to the model. In addition, we use multiple retrieval strategies to obtain the most relevant knowledge from two knowledge bases, further improving the model performance.

Our main contributions are as follows:

1. We propose a unified retrieval-augmented system for fine-grained multilingual NER. Our system incorporates more diverse knowledge bases and significantly improves the system performance compared to baseline systems (Section § 4, § 5)
2. We initiated our investigation by identifying the primary bottleneck of the previous top-performing system, which we determined to be insufficient knowledge. Consequently, we focused on exploring both data and model enhancements to improve system performance. (Section § 3)
3. We employ multiple retrieval strategies to obtain entity information from Wikidata, in order to complement the missing entity knowledge. (Section § 4.1)
4. Additionally, we utilize the infusion approach to provide a more extensive contextual view to the model, thus enabling better utilization of the retrieved context (Section § 4.2).
5. Extensive experimental analysis demonstrates the effectiveness of diverse knowledge sources and broader contextual scopes for improving model performance. (Section § 5)

2 Related Work

Named Entity Recognition (NER) (Sundheim, 1995) is a fundamental task in Natural Language Processing. Because of the long-term attention and the rapid development of pre-trained language models, various models (Akbik et al., 2018; Devlin et al., 2019; Yamada et al., 2020; Wang et al., 2020, 2021a) have achieved state-of-the-art results and performance in general NER scenarios and datasets, such as CoNLL 2002 (Sang, 2002), CoNLL 2003 (Tjong Kim Sang and De Meulder, 2003), and OntoNotes 5.0 (Pradhan et al., 2013). Considering that the previous task settings or datasets are monolingual and scenario-constrained, the task of **Multi**lingual **Co**mplex **N**amed **E**ntity **R**ecognition (MultiCoNER) is proposed to promote the NER research to be more oriented to real scenarios (Malmasi et al., 2022b). Our work focuses on this task and we will introduce the related work from the dataset and method of MultiCoNER respectively:

Challenges of MultiCoNER Dataset To address contemporary in the NER field, Malmasi et al. construct MultiCoNER, a large and complex dataset for Multilingual Complex Named Entity Recognition. This 26M token dataset covers 3 domains (including Wiki, question, and search query) and 11 languages (12 languages for SemEval-2023). In particular, aiming at the main challenges of NER research, the MultiCoNER dataset sets 4 key characteristics: (1) **Low Context**: Existing NER methods perform poorly if the context is less informative (Meng et al., 2021), thus, texts in MultiCoNER are low in context to assess the model's performance on the more realistic and difficult setting. (2) **Sufficient Diversity**: MultiCoNER contains an rich variety of entity types, both simple and difficult, which makes it possible to evaluate the model more comprehensively. (3) **Reasonable Distribution**: Considering the non-negligible long-tail distribution problem faced by the previous datasets makes the construction of training data extremely difficult, MultiCoNER ensures that the distribution of its entities is more even and reasonable so that it can be evaluated comprehensively. (4) **High Complexity**: Increasing the complexity of the dataset can effectively improve the quality of the dataset (Fetahu et al., 2021). Therefore, in addition to monolingual subsets, MultiCoNER also distinctively contains a multilingual subset and a code-mixed one, which makes it more challenging. Note that in the dataset version of SemEval-2023, this challenge and setting do not exist.

Progress of MultiCoNER Methods With the MultiCoNER dataset as the core, the SemEval-2022 Task 11 attracts 236 participants, and 55 teams successfully submit their system (Malmasi et al., 2022b). Among them, there are many successful and excellent works worthy of discussion. DAMO-NLP (Wang et al., 2022b) proposes a knowledge-based method that gets multilingual knowledge from Wikipedia to provide informative context for the NER model. And they achieve the previous best overall performance on the MultiCoNER dataset. USTC-NELSLIP (Chen et al., 2022) proposes a gazetteer-adapted integration network to improve the model performance for recognizing complex entities. QTrade AI (Gan et al., 2022) designs kinds of data augmentation strategies for the low-resource mixed-code NER task. *Previous efforts and studies on the MultiCoNER dataset have shown that external data and beneficial knowledge are essential to improve the performance of NER models on it.*

Retrieval-augmented NLP Methods Retrieval-augmented techniques have proven to be highly effective in various natural language processing (NLP) tasks, as evidenced by the exceptional performance achieved in prior studies (Lewis et al., 2020; Khandelwal et al., 2019; Borgeaud et al., 2022). These approaches usually contain two parts: an information retrieval module and a task-specific module. Specifically, in the context of named entity recognition (NER), Wang et al. (2021b) proposes leveraging off-the-shelf search engines like Google to retrieve external information and enhance the contextual representations of tokens in the input text, resulting in improved performance. Furthermore, subsequent research has focused on developing task-specific retrieval systems for domain-specific NER and multi-modal NER tasks, respectively (Zhang et al., 2022b; Wang et al., 2022a). Drawing upon these insights, our proposed system is designed and optimized with guidance from these previous works.

Language	Data Type	P	R	F1	Ratio
BN	Total	90.99	92.60	91.79	1.00
	In-context	92.86	94.66	93.75	0.69
	Out-of-context	88.06	89.39	88.72	0.31
	Δ	4.80	5.27	5.03	-
DE	Total	81.83	83.00	82.41	1.00
	In-context	83.80	88.11	85.90	0.54
	Out-of-context	80.17	78.98	79.57	0.46
	Δ	3.63	9.13	6.33	-
ZH	Total	76.71	78.40	77.54	1.00
	In-context	79.27	83.87	81.50	0.26
	Out-of-context	76.04	77.02	76.53	0.74
	Δ	3.23	6.85	4.97	-

Table 1: The performance and ratio for different types of data on `BN`, `DE` and `ZH`.

3 Data

The MultiCoNER II corpus (Fetahu et al., 2023a) aims to recognize the complex named entities and pose new challenges for current NER systems. To meet these challenges, we first reproduce the results of the top system (Wang et al., 2022b) and perform error analysis on validation sets. We observe that the performance bottleneck of the system lies in the lack of knowledge. Then, we investigate to break this bottleneck from data and model perspectives and improve model robustness.

Following Wang et al. (2022b), we build a multilingual KB based on Wikipedia of the 12 languages to search for the related documents. We download the latest (2022.10.21) version of the Wikipedia dump from Wikimedia[2] and convert it to plain texts. We execute the official system on MultiCoNER II corpus and categorize the results according to whether the annotated entity appears in the retrieval context or not. As shown in Table 1, the F1-measure on different types of test data differs significantly, e.g., 6.33% on `DE` and 4.97% on `ZH`. This indicates that the lack of knowledge about entities in the retrieval context can have a significant impact on the model performance. With this insight, we consider data and model dimensions to compensate for this lack of knowledge.

[2] `https://dumps.wikimedia.org/`

Retrieval Strategy	Query	Retrieval Result
TEXT2TEXT	from 1995 to 2011 deal hudson was the magazine’s publisher.	1. In 1995 Hudson became publisher of the conservative Roman Catholic magazine, Crisis. 2. Hudson is the former publisher and editor of 3. Hudson also hosts the radio show Church and Culture on Ave Maria Radio ...
TEXT2ENT	from 1995 to 2011 deal hudson was the magazine’s publisher.	1. Deal W. Hudson 2. Deal Wyatt Hudson 3. S. Hudson ...
ENT2ENT	[deal hudson]	Type: human Description: Hudson is the former publisher and editor of Crisis Magazine and InsideCatholic.com.

Table 2: Examples of different retrieval strategies related to the input sentence: *"from 1995 to 2011 deal hudson was the magazine’s publisher."* with its corresponding entity *"deal hudson"*.

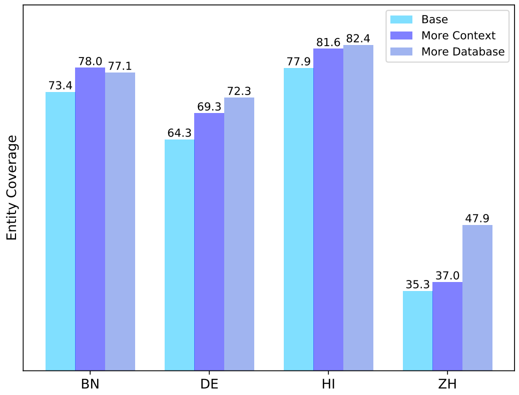


Figure 2: Entity coverage of the retrieval context for the annotated entities within the query sentence.

While Chen et al. (2022) uses Wikidata to build their gazetteer, we explore to enhance our retrieval system with Wikidata. Wikidata is a free and entity-centric knowledge base. Every entity of Wikidata has a page consisting of a label, several aliases, descriptions, and one or more entity types. As shown in Figure 2, Base indicates that only the Wikipedia knowledge base is used, and More Database indicates that we use both Wikipedia and Wikidata knowledge bases. The entity coverage improves on all 4 languages and achieves the maximum gain of 12.6% on ZH. In addition, as More Context shows, expanding the length of the retrieval context also brings more entity knowledge. Thus, we use the infusion approach to make more retrieval context visible to model. More details are described in Section § 4.2.

4 Methodology

Overview As depicted in Fig. 3, U-RaNER is comprised of two parts: a retrieval augmentation module and a NER module. The retrieval augmentation module utilizes multiple retrieval strategies and the NER module adopts a modified transformer structure to utilize the retrieved knowledge. Given an input sentence, U-RaNER retrieves similar texts and entities as external knowledge, which are then utilized in the form of text and vectors to help the NER module obtain improved predictions.

4.1 Retrieval Augmentation Module

In the retrieval augmentation module, we design three different retrieval strategies, namely TEXT2TEXT, TEXT2ENT, and ENT2ENT, which aim to obtain a variety of useful information from different sources to enhance our NER model.

TEXT2TEXT The TEXT2TEXT retrieval strategy is to obtain texts related to input sentences from Wikipedia by the way of sparse retrieval (McDonell, 1977; Robertson and Zaragoza, 2009). Through this form of retrieval, the goal is to obtain additional and useful relevant information as much as possible to alleviate the low-context problem of MultiCoNER. Specifically, we first parse the latest Wikipedia dumps and use ElasticSearch [3] to index them. And finally, we use each sentence in the dataset as the query and use the BM25 retrieval algorithm that comes with ElasticSearch to search in the built index database to obtain the Top-K documents related to the input sentence from Wikipedia, as shown in the first example of Table 2. Note that the TEXT2TEXT strategy is used by Wang et al. (2022b) to win 10 out of 13 tracks when competing in the SemEval-2022 Task 11.

TEXT2ENT The TEXT2ENT retrieval strategy aims to retrieve candidate entities that may be men-

[3] https://github.com/elastic/elasticsearch

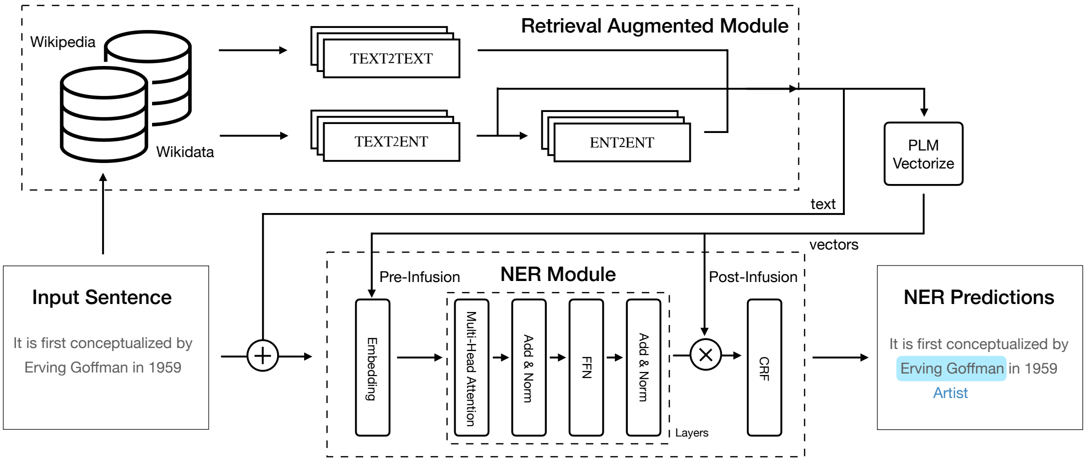


Figure 3: Overall architecture of U-RaNER.

tioned in input sentences, as illustrated in the second example of Table 2. We believe that if the candidate entities that may be mentioned in the sentence can be retrieved in advance, the related knowledge might be helpful to build a stronger entity recognition model. The TEXT2ENT strategy is inspired by the related technologies of dictionary disambiguation (Harige and Buitelaar, 2016) and entity linking (Cao et al., 2021). But dictionary disambiguation can only perform hard matching, and there is no detailed annotation information for entity linking (that is, the corresponding information between span and entity), so these two traditional methods cannot be directly applied to our scene. Therefore, in this part of the specific practice, we tried two different retrieval methods, namely sparse retrieval and dense retrieval. The details of these two retrieval methods are in the Appendix A.4.

ENT2ENT The ENT2ENT retrieval strategy aims to retrieve some entities and their corresponding information from Wikidata. Wikidata integrates billions of structural information between millions of entities, such as the alias of entities and the relationships of entity pairs. And intuitively, such information is beneficial to our NER model.

In the process of ENT2ENT retrieval, we want to find out external entity types which maybe inspire the entity labeling of the input sentence. Concretely, for each given entity, we first retrieve Wikidata to get its relevant Wikidata entities. Next, we gather and utilize the properties of the Wikidata entities from their corresponding Wikidata pages. In particular, we take the “instance of" and “sub-class of" properties as the entity types. For example, as shown in Table 2, with entity “deal hudson” as the query, ENT2ENT strategy will retrieve its type (i.e., “human”) and description text. Finally, all relevant Wikidata entities and their types are as the retrieved augmented data. The detailed procedure for ENT2ENT is in the Appendix A.5.

4.2 Named Entity Recognition Module

BERT-CRF We use *xlm-roberta-large* (XLM-R) (Conneau et al., 2020) as the PLMs for all the tracks. Given an input sentence $\mathbf{x} = x_1, x_2, \ldots, x_n$, transformer-based standard fine-tuning for NER first feeds the input sentence $\mathbf{x}$ into the PLMs to get the token representations $\mathbf{h}$. The token representations $\mathbf{h}$ are fed into a CRF layer to get the conditional probability $p_\theta(\mathbf{y} \mid \mathbf{h})$, and the model is trained by maximizing the conditional probability and minimizing the cross entropy loss: $\mathcal{L} = -\log p_\theta(\mathbf{y} \mid \mathbf{h})$.

RaNER Given the retrieval context $\tilde{\mathbf{x}}$, we define a neural network parameterized by θ that learns from a concatenated input $[\mathbf{x}; \tilde{\mathbf{x}}]$. We feed the input and retrieve the representation $[\mathbf{h}; \tilde{\mathbf{h}}]$:

$$[\mathbf{h}; \tilde{\mathbf{h}}] = [h^{(1)}, \ldots h^{(n)}, \tilde{h}^{(1)}, \ldots \tilde{h}^{(n)}] = \text{embed}([\mathbf{x}; \tilde{\mathbf{x}}]) \quad (1)$$

We then feed $\mathbf{h}$ into the CRF layer and train by minimizing the conditional probability $p_\theta(\mathbf{y} \mid \mathbf{h})$ as mentioned above.

U-RaNER To exploit more retrieval contexts, we first slice $\tilde{\mathbf{x}}$ by model-limited input length as $\tilde{\mathbf{x}} = \tilde{x}_0, \tilde{x}_1, \ldots, \tilde{x}_m$. Then, we keep $\tilde{x}_0$ as the text for concatenation, and feed the rest context list into PLM as $[(\mathbf{x}; \tilde{x}_1), \ldots, (\mathbf{x}; \tilde{x}_m)]$, which is used in Lewis et al. (2020) for better information interaction, and get the token vector list

$[(\mathbf{h}_1; \tilde{h}_1), \ldots, (\mathbf{h}_m; \tilde{h}_m)]$. Afterwards, we consider two infusion (Pre-Infusion and Post-Infusion) approaches using the representation $[\tilde{h}_1, \ldots, \tilde{h}_m]$ and $[\mathbf{h}_1, \ldots, \mathbf{h}_m]$, respectively.

For `Pre-Infusion`, we fetch the token vectors of the corresponding positions of the anchors from the vector list $[\tilde{h}_1, \ldots, \tilde{h}_m]$. Then, we perform the mean operation to obtain the set of anchor vectors $\mathcal{V} \in \mathbb{R}^{p \times d}$, p is the number of anchors, and d is the hidden size. Considering that the word embedding layer in XLM-R has two input modes, including vocabulary index input as well as word embedding input, we first perform the former for $[\mathbf{x}; \tilde{x}_0]$ to obtain the input text embedding E, and later concatenate E and the anchor vectors $\mathcal{V}$ to form the word embedding input. Finally, we get the representation $[\mathbf{h}; \tilde{h}_0; \tilde{h}_v]$. We only use $\mathbf{h}$ to pass the CRF layer.

For `Post-Infusion`, we first feed $[\mathbf{x}; \tilde{x}_0]$ to XLM-R and get the token representation $[\mathbf{h}; \tilde{h}_0]$. For input representation list $[\mathbf{h}; \mathbf{h}_1, \ldots, \mathbf{h}_m]$, we perform the max operation on the token dimension to obtain the final representation $\mathbf{h}_{\text{max}}$. Then, we use $\mathbf{h}_{\text{max}}$ for calculation as in `BERT-CRF`. Notably, we find that the post-infusion method is superior to the pre-infusion method in our preliminary experiments, and the default infusion method in the experimental section is post-infusion.

4.3 Ensemble Module

Given predictions $\{\hat{\boldsymbol{y}}_{\theta_1}, \cdots, \hat{\boldsymbol{y}}_{\theta_m}\}$ from m models with different random seeds, we use majority voting to generate the final prediction $\hat{\boldsymbol{y}}$. Following Yamada et al. (2020); Wang et al. (2022b), the module ranks all spans in the predictions by the number of votes in descending order and selects the spans with more than 50% votes into the final prediction. The spans with more votes are kept if the selected spans have overlaps and the longer spans are kept if the spans have the same votes.

5 Experimental Setup

5.1 Datasets and Evaluation Metrics

We use the official MultiCoNER II dataset (Fetahu et al., 2023a) in all tracks to train our models. The detailed data statistics is in the Appendix A.1 and A.3. The results on the leaderboard are evaluated with the entity-level macro F1 scores, which treat all the labels equally [4].

5.2 Training Strategy

NER Model Training Our final NER models are trained on the combined dataset including both the training and development sets on each track to fully utilize the labeled data. For models trained on the combined dataset, we use the final model checkpoint after training. The detailed system configurations is in the Appendix A.2

Multi-stage Fine-tuning Multi-stage fine-tuning (MSF) aims at transferring the parameters of fine-tuned embeddings in a model at an early stage into other models in the next stage Shi and Lee (2021). The approach stores the checkpoint of fine-tuned XLM-R embeddings at the early stage and uses it as the initialization of XLM-R embeddings for model training at the next stage. Wang et al. (2022b) experimentally demonstrates that MSF can leverage the annotations from all tracks and thus improve performance and accelerate training. In addition, we observe that inconsistent training set sizes on different language tracks can also lead to degradation of model performance. We use increasing batch size and upsampling strategy to address this issue. The details are shown in the Appendix B.1.

5.3 Baselines

In this paper, we compare the proposed U-RaNER with the following baseline models:

- **BERT-CRF**, as introduced in 4.2, is composed of a BERT-like encoder and a CRF decoder . It is widely used for sequence labeling tasks. We use *xlm-roberta-large* (XLM-R) (Conneau et al., 2020) as the pretrained backbone for all the tracks.

- **RaNER**, as introduced in 4.2, improves BERT-CRF by incorporating retrieval contexts as input for better performance. Retrieval augmented methods have proven to be highly effective in the NER task(Wang et al., 2021b; Zhang et al., 2022b; Wang et al., 2022a).

- **RaNER-MSF** (Wang et al., 2022b) achieves the previous best overall performance on the Multi-CoNER I dataset, which exploits multi-stage fine-tuning to leverage the annotations

[4] In comparison, most of the publicly available NER datasets (e.g., CoNLL 2002, 2003 datasets) are evaluated with the entity-level micro F1 scores, which emphasize common labels (Akbik et al., 2018; Devlin et al., 2019; Yamada et al., 2020; Wang et al., 2022b). Except for the results in Table 3, the following results are entity-level micro F1 scores if not otherwise specified.

System	EN	ES	SV	UK	PT	FR	FA	DE	ZH	HI	BN	IT	MULTI	AVG.
BERT-CRF	62.80	65.34	68.68	67.68	64.37	66.05	60.70	69.44	62.02	73.08	71.82	68.15	63.27	66.42
NLPeople	71.81	72.76	75.08	73.41	70.16	72.85	70.76	77.67	65.96	78.50	78.24	73.71	78.38	73.79
USTC-NELSLIP	72.15	74.44	75.47	74.37	71.26	74.25	68.85	78.71	66.57	82.14	80.59	75.70	75.62	74.62
IXA/Cogcomp	72.82	73.81	76.54	75.25	72.28	74.52	69.49	80.35	64.86	79.56	78.95	74.67	78.17	74.71
CAIR-NLP	79.33	83.63	82.88	81.29	80.16	83.08	77.50	74.71	58.43	72.23	69.46	83.78	79.16	77.36
PAI	80.00	71.67	72.38	71.28	81.61	86.17	68.46	**88.09**	74.87	**80.96**	**84.39**	84.88	77.00	78.60
NetEase.AI	-	-	-	-	-	-	-	-	**84.05**	-	-	-	-	-
Ours	**85.53**	**89.78**	**89.57**	**89.02**	**85.97**	**89.59**	**87.93**	84.97	75.98	78.56	81.60	**89.79**	**84.48**	**85.60**

Table 3: Part of the official results on the leaderboard. BERT-CRF is the post-evaluation results of our baseline system (BERT-CRF) on the released test set.

from all tracks and thus improve performance and accelerate training of RaNER.

- **ChatGPT**[5], also known as gpt-3.5-turbo, is the most capable GPT-3.5 (Ouyang et al., 2022) model and optimized for chat. Following (Lai et al., 2023), our prompt structure for ChatGPT consists of a task description, a note for output format, and an input sentence. Despite a Single-turn prompting strategy, we additionally try two enhanced prompting strategies: Multi-turn and Multi-ICL. Multi-turn first performs the task in 6 coarse-grained categories, and later performs finer-grained NER. Multi-ICL constructs demonstrations spliced after the note part by randomly selecting examples from the training set. The detailed prompting procedure for Single-turn, Multi-turn and Multi-ICL is in the Appendix A.6.

6 Results and Analysis

6.1 Main Results

There are 45 teams that participated in the MultiCoNER II shared task. Due to limited space, we only compare our system with the systems from teams NLPeople, USTC-NELSLIP, IXA/Cogcomp, CAIR-NLP, PAI and NetEase.AI[6]. As NetEase.AI solely took part in the Chinese track, which means we only have access to their results for this specific track. In the post-evaluation phase, we evaluate the baseline system without the use of additional knowledge bases to further show the effectiveness of our retrieval-augmented system. The official results and the results of our baseline system are shown in Table 3. Our system performs the best on 9 out of 13 tracks with the average result exceeding the second-place system by the absolute F1-measure of 7.0%. Moreover, our system outperforms our baseline by the 19.18% F1-measure on average, which demonstrates that the retrieval-augmented system based on multiple knowledge bases is extremely helpful in identifying complex entities, leading to significant improvement on model performance.

In addition, we use three prompting strategies to evaluate ChatGPT. Due to the overwhelming number of test sets (millions of levels), the expense of invoking the OpenAI interface is unaffordable. We experiment on the validation set and the results are in Table 4. We observe that ChatGPT's performance on the multilingual NER dataset is quite poor, with an average F1-score of only 14.78% by the best strategy. Even on the coarse-grained level the result is merely 29.70% (Table 5), which is comparable to the result measured on MultiCoNER I (Malmasi et al., 2022b) by Lai et al. (2023).

6.2 Ablation Study

In this section, we perform extensive ablation experiments to show the effectiveness of various settings in our retrieval-augmented system. Following Wang et al. (2022b), we employ the multi-stage fine-tuning (MSF) training strategy. As shown in Table 4, the model performance improves from 87.95% to 89.92%, which illustrates the effectiveness of the multi-stage training. Note that the following five rows in Table 4 all use the MSF training strategy.

For the different knowledge sources, the use of Wikipedia data achieves the gain of 12.61% (RaNER-MSF vs. BERT-CRF), the use of wikidata data achieves the gain of 13.16% (ENT2ENT* vs. BERT-CRF), and using both together achieves the maximum gain of 15.46% (ENT2ENT vs. BERT-CRF). This shows that knowledge is highly

[5] https://openai.com/blog/chatgpt/

[6] Please refer to https://multiconer.github.io/results for more details about the results.

Method	△	†	‡	BN	DE	EN	ES	FA	FR	HI	IT	PT	SV	UK	ZH	AVG.
ChatGPT w/																
`Single-turn`	✗	✗	✗	7.24	10.06	13.36	12.44	10.94	11.05	9.04	16.32	17.27	18.03	10.88	5.02	11.80
`Multi-turn`	✗	✗	✗	8.12	14.57	15.38	15.52	12.75	13.60	9.17	17.81	17.70	20.38	14.25	5.60	13.74
`Multi-ICL`	✗	✗	✗	9.76	14.84	17.65	16.28	14.11	13.95	10.48	18.63	18.84	20.94	15.57	6.34	14.78
BERT-CRF	✗	✗	✗	86.98	76.08	72.61	75.66	69.37	74.44	85.46	80.70	76.54	78.48	76.30	75.11	77.31
RaNER	✗	✓	✗	92.30	84.29	84.32	88.81	87.85	86.77	91.75	91.08	88.45	89.74	88.46	81.55	87.95
RaNER-MSF	✗	✓	✗	93.11	86.81	86.82	90.90	89.52	88.99	93.97	92.42	90.75	91.93	90.93	82.83	89.92
U-RaNER w/																
`TEXT2ENT*`	✗	✗	✓	89.87	85.83	87.54	88.03	86.44	83.86	86.82	91.19	78.92	86.20	84.26	85.62	86.22
`ENT2ENT*`	✗	✗	✓	94.45	88.85	88.11	91.34	89.70	89.96	94.68	91.53	90.15	91.68	88.21	87.02	90.47
`TEXT2TEXT`	✓	✓	✗	94.36	87.79	88.07	92.57	90.91	91.80	94.25	93.60	91.94	93.02	91.40	84.11	91.15
`TEXT2ENT`	✓	✓	✓	94.77	89.48	89.88	93.46	90.80	90.83	94.57	93.83	92.12	93.20	91.12	89.41	91.96
`ENT2ENT`	✓	✓	✓	**94.96**	**90.36**	**90.62**	**93.51**	**91.85**	**92.88**	**95.12**	**94.60**	**92.90**	**94.45**	**91.57**	**90.38**	**92.77**

Table 4: The top bar shows ChatGPT's performance (micro-F1 scores) using three prompting strategies, the former two being zero-shot learning and `Multi-ICL` being few-shot learning. Following the comparison between the top system (Wang et al., 2022b) in the MultiCoNER I and the three variants of our method on the validation set. ⋆ indicates that we merely use the Wikidata knowledge base. △ means we scale the model horizon with the infusion approach. † and ‡ indicate the use of the Wikipedia or Wikidata knowledge base.

	Method	BN	ES	PT	SV	ZH	AVG.
Coarse	ChatGPT	21.26	33.86	35.27	40.11	18.01	29.70
	RaNER	95.92	96.17	96.79	97.55	91.94	95.67
	U-RaNER	97.48	98.30	98.33	98.49	95.55	**97.63**
	Δ	+1.56	+2.13	+1.54	+0.94	+3.61	+1.96
Fine	ChatGPT	9.76	16.28	18.84	20.94	6.34	14.43
	RaNER	93.11	90.90	90.75	91.93	82.83	89.90
	U-RaNER	94.96	93.51	92.90	94.45	90.38	**93.24**
	Δ	+1.85	+2.61	+2.15	+2.52	+7.55	+3.34

Table 5: Comparison of the performance between ChatGPT, RaNER and U-RaNER at coarse-and-fine grained categories.

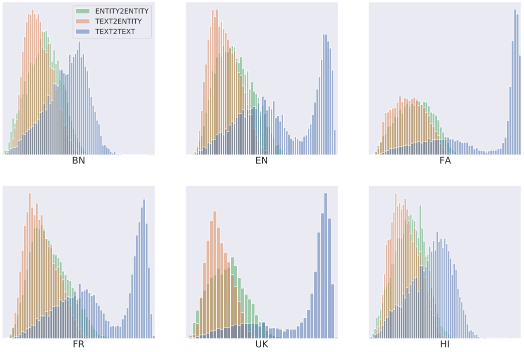


Figure 4: The distribution of the character-level IoU between query and its retrieval result. Each subplot is the histograms of different retrieval strategies on the corresponding dataset, where the x-axis indicates the IoU values ranging from 0 to 1.

useful for system performance and illustrates the complementarity of the two knowledge bases.

For the different knowledge acquisition methods, the `ENT2ENT` approach is superior to the `TEXT2ENT` approach (90.47% vs. 86.22%). In addition, we use the infusion approach to further improve the model performance (`RaNER-MSF` vs. `TEXT2TEXT`), which suggests that guaranteeing knowledge to be visible to model is also important. The default infusion method in our experiments is post-infusion. We also analyze the impact of the two different infusion methods on performance in the Appendix B.2.

6.3 Coarse-and-fine Category Analysis

To illustrate the advantages of U-RaNER on fine-grained NER, we transform the model predictions to the coarse-grained level according to the official topology of fine-grained categories. We use the models of RaNER-MSF and U-RaNER w/ `ENT2ENT` in Table 4 for the analysis. As shown in the Table 5, the improvements in coarse-grained metrics are significantly lower than those of fine-grained metrics, differing by 1.38% (3.91% on the `ZH` track). It suggests that the proposed U-RaNER is better at coping with complex scenarios of fine-grained classification. Besides, the average F1 for ChatGPT at different granularity is significant distinct (29.70% vs. 14.43%), which shows the difficulty in identifying fine-grained complex entities.

6.4 Query Relevance

We define a relevance metric to compute the relevance between the query and retrieval result. The metric calculates the Intersection-over-Union (IoU) between the characters [7] of the query and those

[7] We take repeat characters as different characters.

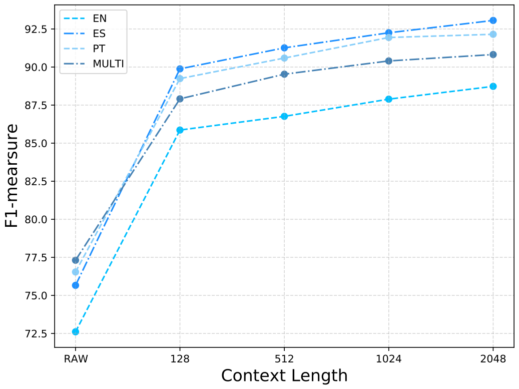


Figure 5: F1-measure with different length of context. `RAW` indicates that no external context is appended.

of the retrieved result. We plot the results on the training set of 6 tracks in Figure 4. It can be observed that the IoU values of `TEXT2TEXT` strategy form a larger cluster than those of `TEXT2ENT` and `ENT2ENT`, which indicates that `TEXT2TEXT` retrieval would focus more on the context instead of merely the entities in the query text. Additionally, we observe that the distributions of `ENT2ENT` have larger medians than those of `TEXT2ENT`. This might due to `ENT2ENT` would retrieve more relevant entities from the Wikidata than `TEXT2ENT`. By employing diverse retrieval techniques, we can leverage data with distinct attributes to improve the effectiveness of the model.

6.5 Context Length Analysis

In this section, we focus on analyzing the impact of different context length on model performance. We conduct a series of experiments on EN, ES, PT and MULTI datasets with the context length ranging from 128 to 2048. We can observe from Figure 5 that the model performance increases as the context length grows. However, when the context list length exceeds 1024, the trend of performance improvement on all four datasets slows down. This indicates that the knowledge capacity in the contexts saturates as the length of the context increases. For better performance, we need to find complementary and highly relevant contextual pieces as additional knowledge sources.

6.6 Error Analysis

We divided the NER task into two stages: mention detection to locate entity spans, and entity typing to classify the spans with pre-defined labels. To further analyze the limitations of our proposed model, we present the experimental results on 12 languages in Table 7 in Appendix. The experimental results reveal that the average F1 score for mention detection is **97.21**, whereas the accuracy for entity typing is **90.35**. These results provide evidence that the bottleneck in fine-grained NER is typing.

More detailed discussion, including the different retrieval methods and case study, is in the Appendix B.3 and B.4.

7 Conclusion

In this paper, we propose a unified retrieval-augmented system (U-RaNER) for the MultiCoNER II shared task, which wins 9 out of 13 tracks in the shared task. We expose that the bottleneck of the previous top system is the lack of knowledge. Accordingly, we use both Wikipedia and Wikidata knowledge bases with three retrieval approaches so that more diverse knowledge can be considered. Also, we explore the infusion approach to make more context visible to the model so as to make the best use of the resources. And the error analysis indicates that the entity typing sub-task is the bottleneck in the current system. In the future, we plan to exploit the knowledge in the large language model such as ChatGPT or LLaMA by self-verification or fine-tuning some adapters, in order to achieve robust generalization performance.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Sandeep Ashwini and Jinho D. Choi. 2014. Targetable named entity recognition in social media. *CoRR*, abs/1408.0782.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2022. Improving language models by retrieving from trillions of tokens. In *International conference on machine learning*, pages 2206–2240. PMLR.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive entity retrieval. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Beiduo Chen, Jun-Yu Ma, Jiajun Qi, Wu Guo, Zhen-Hua Ling, and Quan Liu. 2022. USTC-NELSLIP at SemEval-2022 task 11: Gazetteer-adapted integration network for multilingual complex named entity recognition. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1613–1622, Seattle, United States. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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, 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.

Besnik Fetahu, Anjie Fang, Oleg Rokhlenko, and Shervin Malmasi. 2022. Dynamic gazetteer integration in multilingual models for cross-lingual and cross-domain named entity recognition. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2777–2790, Seattle, United States. Association for Computational Linguistics.

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.

Weichao Gan, Yuanping Lin, Guangbo Yu, Guimin Chen, and Qian Ye. 2022. Qtrade AI at SemEval-2022 task 11: An unified framework for multilingual NER task. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1654–1664, Seattle, United States. Association for Computational Linguistics.

Ravindra Harige and Paul Buitelaar. 2016. Generating a large-scale entity linking dictionary from Wikipedia link structure and article text. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2431–2434, Portorož, Slovenia. European Language Resources Association (ELRA).

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2019. Generalization through memorization: Nearest neighbor language models. *arXiv preprint arXiv:1911.00172*.

Viet Dac Lai, Nghia Trung Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Huu Nguyen. 2023. Chatgpt beyond english: Towards a comprehensive evaluation of large language models in multilingual learning.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

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.

Ken J. McDonell. 1977. An inverted index implementation. *Comput. J.*, 20(2):116–123.

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.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards robust linguistic analysis using OntoNotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 143–152, Sofia, Bulgaria. Association for Computational Linguistics.

Stephen E. Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

Erik F. Tjong Kim Sang. 2002. Introduction to the conll-2002 shared task: Language-independent named entity recognition. In *Proceedings of the 6th Conference on Natural Language Learning, CoNLL 2002, Held in cooperation with COLING 2002, Taipei, Taiwan, 2002*. ACL.

Tianze Shi and Lillian Lee. 2021. TGIF: Tree-graph integrated-format parser for enhanced UD with two-stage generic- to individual-language finetuning. In *Proceedings of the 17th International Conference on Parsing Technologies and the IWPT 2021 Shared Task on Parsing into Enhanced Universal Dependencies (IWPT 2021)*, pages 213–224, Online. Association for Computational Linguistics.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

Beth M. Sundheim. 1995. Named entity task definition, version 2.1. In *Proceedings of the Sixth Message Understanding Conference*, pages 319–332.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Xinyu Wang, Jiong Cai, Yong Jiang, Pengjun Xie, Kewei Tu, and Wei Lu. 2022a. Named entity and relation extraction with multi-modal retrieval. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5925–5936, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2021a. Automated Concatenation of Embeddings for Structured Prediction. In *the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (**ACL-IJCNLP 2021**)*. Association for Computational Linguistics.

Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2021b. Improving named entity recognition by external context retrieving and cooperative learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1800–1812, Online. Association for Computational Linguistics.

Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Huang Zhongqiang, Fei Huang, and Kewei Tu. 2020. More embeddings, better sequence labelers? In *Findings of EMNLP*, Online.

Xinyu Wang, Yongliang Shen, Jiong Cai, Tao Wang, Xiaobin Wang, Pengjun Xie, Fei Huang, Weiming Lu, Yueting Zhuang, Kewei Tu, Wei Lu, and Yong Jiang. 2022b. DAMO-NLP at SemEval-2022 task 11: A knowledge-based system for multilingual named entity recognition. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1457–1468, Seattle, United States. Association for Computational Linguistics.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454, Online. Association for Computational Linguistics.

Wenzheng Zhang, Wenyue Hua, and Karl Stratos. 2022a. Entqa: Entity linking as question answering. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Xin Zhang, Yong Jiang, Xiaobin Wang, Xuming Hu, Yueheng Sun, Pengjun Xie, and Meishan Zhang. 2022b. Domain-specific NER via retrieving correlated samples. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2398–2404, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

A Detailed Experimental Setup

A.1 MultiCoNER II Corpus

The multilingual NER II corpus (MultiCoNER II[8]) aims to recognize the complex named entities, like the titles of creative works which are not simple nouns, and pose challenges for current NER systems. With the same set of tags, the 12 multilingual datasets specifically include: BN-Bangla, DE-German, EN-English, ES-Spanish, FA-Farsi, FR-French, HI-Hindi, IT-Italian, PT-Portuguese, SV-Swedish, UK-Ukrainian and ZH-Chinese. Table 6 shows the detailed dataset statistics.

A.2 System Setup

For fair comparison with prior systems, we use *xlm-roberta-large* (Conneau et al., 2020) as our initial checkpoint. We use the AdamW (Loshchilov

[8] https://multiconer.github.io/dataset

Language	Training	Validataion	Test
BN-Bangla	9,708	507	19,859
DE-German	9,785	512	20,145
EN-English	16,778	871	249,980
ES-Spanish	16,453	854	246,900
FA-Farsi	16,321	855	219,168
FR-French	16,548	857	249,786
HI-Hindi	9,632	514	18,399
IT-Italian	16,579	858	247,881
PT-Portuguese	16,469	854	229,490
SV-Swedish	16,363	856	231,190
UK-Ukrainian	16,429	851	238,296
ZH-Chinese	9,759	506	20,265
MUL-Multilingual	170,824	8,895	358,668

Table 6: Dataset statistics on MultiCoNER II.

and Hutter, 2017) optimizer with a linear warmup-decay learning schedule and a dropout (Srivastava et al., 2014) of 0.1. We set the batch size and learning rate to 16 and 2e-5, and train models over 4 random seeds. According to the dataset sizes, we train the models for 5 epochs and 20 epochs for multilingual and monolingual models respectively. And all our experiments are conducted on a single NVIDIA A100 80GB GPU. For the ensemble module, we train about 4 models for each track.

A.3 Fine-grained Taxonomy

The tagset of MultiCoNER II is a fine-grained tagset including 6 coarse-grained categories and 33 fine-grained categories. The coarse-to-fine mapping of the tags are as follows:

- Location (LOC): Facility, OtherLOC, HumanSettlement, Station;
- Creative Work (CW): VisualWork, MusicalWork, WrittenWork, ArtWork, Software;
- Group (GRP): MusicalGRP, PublicCORP, PrivateCORP, AerospaceManufacturer, SportsGRP, CarManufacturer, ORG;
- Person (PER): Scientist, Artist, Athlete, Politician, Cleric, SportsManager, OtherPER;
- Product (PROD): Clothing, Vehicle, Food, Drink, OtherPROD;
- Medical (MED): Medication/Vaccine, MedicalProcedure, AnatomicalStructure, Symptom, Disease.

The Figure 6 shows the fine-grained taxonomy.

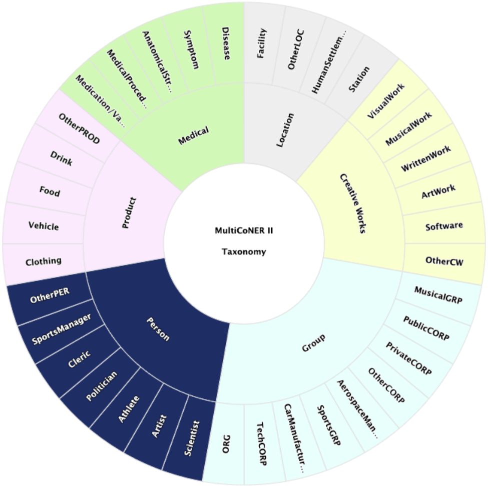


Figure 6: The taxonomy of fine-grained categories on MultiCoNER II from the official webpage.

Language	**F1-entity**	**F1-mention**	**Acc-typing**
BN	92.30	97.33	94.83
DE	84.29	95.00	88.73
EN	84.32	98.15	85.91
ES	88.81	98.13	90.50
FA	87.85	97.21	90.37
FR	86.77	97.34	89.14
HI	91.75	97.15	94.44
IT	91.08	98.53	92.44
PT	88.45	98.45	89.84
SV	89.74	98.60	91.01
UK	88.46	98.33	89.96
ZH	81.55	92.25	87.00
AVG.	87.84	**97.21**	**90.35**

Table 7: Model performance of mention-detection and entity-typing on the 12 multilingual datasets.

A.4 Detailed Procedure for `TEXT2ENT`

For sparse retrieval, we find the relevant entities from Wikidata which contains millions of entities. As in the TEXT2TEXT strategy, we utilize the description and alias information in the Wikidata and index them with ElasticSearch. We use each sentence in the dataset as the query and retrieve the candidate entity with the BM25 algorithm. In order to find candidate entities as much as possible, we apply an iterative retrieval procedure in which we construct a new query by masking the retrieved entities in the query text from the previous retrieval.

For dense retrieval, we utilize the title information and paragraph information [9] from Wikipedia to construct the knowledge base for dense entity

[9] Considering the memory limit of dense retrieval model training, we truncate the paragraph information in wikipedia, and reserve the first 128 tokens for the construction of the knowledge base.

Method	BN	DE	EN	ES	FA	FR	HI	IT	PT	SV	UK	ZH	AVG.
RaNER w/ one stage	91.79	82.41	84.32	87.49	85.69	85.48	90.68	89.51	87.46	88.54	87.82	76.45	86.47
RaNER w/ bs 4	82.02	80.82	85.60	88.46	85.27	87.53	86.56	89.80	87.26	89.77	89.17	68.59	85.07
RaNER w/ bs 128	88.09	83.23	85.87	89.40	85.59	88.18	89.57	91.84	88.97	90.01	88.97	72.11	86.82
RaNER w/ scale up	90.82	86.27	85.86	89.88	86.15	88.70	90.99	91.50	89.24	90.85	88.95	75.71	87.91

Table 8: The model performance with different training strategies.

Method	BN	DE	EN	ES	FA	FR	HI	IT	PT	SV	UK	ZH	AVG.
RaNER	89.81	80.55	79.98	82.99	81.17	81.73	90.57	87.48	83.61	84.43	83.69	77.30	83.61
U-RaNER w/ Pre-infusion	91.35	82.80	83.71	86.73	86.63	85.88	91.07	89.08	87.18	89.16	88.69	80.41	86.89
U-RaNER w/ Post-infusion	91.82	83.24	84.50	86.85	87.64	87.21	91.23	90.36	87.98	90.47	90.02	81.15	87.71

Table 9: The model performance with different infusion approaches.

retrieval, then use the input sentence as the query to retrieve its related Top-K entities in the knowledge base. The dense retrieval model we use is the widely used Bi-Encoder architecture (Zhang et al., 2022a). Different from sparse retrieval, the dense retrieval model is trainable to better perceive the semantic characteristics of the MultiCoNER dataset. Therefore, in practice, we first preprocess the train/dev sets of MultiCoNER into the data format for dense retrieval model training. Specifically, because the train/dev sets provide the golden entity annotation of the sentence, we can fuzzy match the span in the sentence with the entity title in our knowledge base to link each span to a specific entity id. Then we use reconstructed training data to train a dense entity retrieval model with reliable performance, which will be finally applied to the test set to obtain candidate entities for the sentences in the test set.

A.5 Detailed Procedure for ENT2ENT

Suppose that we have already retrieved the boundaries of possible or relative entities of a sentence, we want to encode more knowledge about these entities to benefit the prediction of target entities and their types. A good choice is leveraging Wikidata which integrates billions of structural information between millions of entities, such as the alias of entities and the relationships of entity pairs. Therefore, we adopt the following steps to acquire `ENT2ENT` knowledge to augment the data so as to enhance the entity recognition ability of our model.

1. We preprocess Wikidata to construct two dictionaries of each language in this task. One takes each entity name and each alias string of each entity in Wikidata as keys and the index (called “Qid") of each entity as values. The other takes Qid of each entity as keys and two attributes (called “subclass of" and “sub-instance of") content of each entity as values. It is worth mentioning that the values of the two attributes associated with each entity in Wikidata are themselves entities. Therefore, this method is referred to as **ENT2ENT** retrieval. For the following description, we call the first dictionary *String-to-Qid* and the second dictionary *Qid-to-Types*.

2. For each language, we retrieve argumentation data according to pre-retrieved entities and the knowledge dictionaries from Step1. Concretely, for each retrieved entity, we first extract the corresponding Qid if it can match one key from the *String-to-Qid* dictionary. Next, if the first operation succeeds, we leverage the Qid to query the *Qid-to-Types* dictionary to get the values of “subclass of" and “sub-instance of" as types of the retrieved entity. It is possible that the values of some Qid in the *Qid-to-Types* dictionary of a specific language are NULL. In this situation, we try to get entity types from the *Qid-to-Types* dictionary of English except for processing English itself.

3. If we get the language-specific types or English types of some pre-retrieved entities from Step2, we sequentially splice these pre-retrieved entities and their retrieved types after the original sentence. For those pre-retrieved entities without retrieved types, we only splice the pre-retrieved entities.

A.6 Detailed Procedure for Prompting

Following (Lai et al., 2023), our `Multi-turn` prompt structure for ChatGPT consists of a task

Sentence	Span	Gold Tag	BERT-CRF	RaNER	U-RaNER
pudendal nerve entrapment can occur when the ...	pudendal nerve entrapment	Disease	-	Symptom	Disease
he debuted for gloucestershire in 1887 at the age of ...	gloucestershire	SportsGRP	SportsGRP	HS	SportsGRP
the main event featured thales leites taking on jesse taylor	thales leites jesse taylor	OtherPER OtherPER	Athlete OtherPER	Athlete Athlete	Athlete Athlete

Table 10: Examples of three NER systems. The entity type HS refers to HumanSettlement.

description, a note for output format, and an input sentence. Since the experiments in Lai et al. (2023) indicate that English prompts work better than multilingual ones, we use English prompts for all languages. As shown in Figure 7, the task description part is used to explain the task and list the entity categories, the note part indicates the annotation scheme and output format, and finally we add the input text. In our experiment, {...} is filled by the content in the Appendix A.3.

Multi-turn first performs the task in 6 coarse-grained categories, and later performs finer-grained NER. In our experiment, {...} is filled by the response of ChatGPT and the content from in the Appendix A.3.

Multi-ICL constructs demonstrations spliced after the note part by randomly selecting examples from the training set. xxx is replaced with the selected example. The corresponding prompts can be found in Figure 8.

Task Description: You are working as a named entity recognition expert and your task is to label a given text with named entity labels. Your task is to identify and label any named entities present in the text. The named entity labels that you will be using are 33 categories, as shown below {...}.
Note: Please use BIO annotation schema to complete this task. Please make sure to label each word of the entity with the appropriate prefix ("B" for the first word of the entity, "I" for any non-initial word of the entity). For words which are not part of any named entity, you should return "O". Your output format should be a list of tuples, where each tuple consists of a word from the input text and its corresponding named entity label.
Input: ["from", "1995", "to", "2011", "deal", "hudson", "was", "the", "magazine's", "publisher", "."]
Output:

Figure 7: Input prompt for Single-turn.

B More Analysis

B.1 Multi-stage Fine-tuning

We observe that inconsistent training set sizes on different language tracks will lead to degradation of model performance from 86.47% to 85.07%. We use increasing batch size and scaling up strategy to address this issue. From the Table 8, increasing batch size from 4 to 128 can improve the model performance from 85.07% to 86.82%. Furthermore, scaling up the training data size on BN, DE, HI and ZH can also result in a gain of +1.09%

B.2 Two Infusion Approaches

In the section § 4.2, we propose two infusion methods (Pre-Infusion and Post-Infusion) to make more context visible to the model. Here, we make a quantitative comparison of their effects on model performance. As shown in the Table 9, we observe that the post-infusion method is superior to the pre-infusion method in all language track. We attribute this to the fact that the pre-infusion method only considers the anchor information and ignores other contextual information, while the post-infusion method uses more contextual knowledge and achieves better performance.

B.3 Different Retrieval Methods

To deeply analyze the effectiveness of the two TEXT2ENT retrieval strategies we design, we compare their retrieval performance (i.e., Recall@50) and the enhanced NER performance (i.e., F1) based on their respective retrieval results. From Table 11, we find that the retrieval performance of sparse retrieval does not seem to be worse than dense, and its recall is higher than dense retrieval for both PT and SV languages. In addition, for the BN and DE languages, although their recall results of sparse retrieval are lower than those of dense retrieval, their final performance of NER is higher than that

Task Description: You are working as a named entity recognition expert and your task is to label a given text with named entity labels. Your task is to identify and label any named entities present in the text. The named entity labels that you will be using are PER (person), LOC (location), CW (creative work), GRP (group of people), PROD (product), and MED (medical).
Note: Please use BIO annotation schema to complete this task. Please make sure to label each word of the entity with the appropriate prefix (“B” for the first word of the entity, “I” for any non-initial word of the entity). For words which are not part of any named entity, you should return “O”.
Demonstrations: Optional. [Input: `xxx`, Output: `xxx`].
Input: [“from”, “1995”, “to”, “2011”, “deal”, “hudson”, “was”, “the”, “magazine’s”, “publisher”, “.”]
Output: {...}.
Input: Please complete the above task at a finer granularity based on the fine-grained taxonomy below {...}.
Output:

Figure 8: Input prompt for Multi-turn and Multi-ICL.

of dense retrieval. We think this is mainly due to the different retrieval sources of the two retrieval strategies. Our sparse strategy is retrieved from Wikidata, while the dense strategy is retrieved from Wikipedia. The retrieval quality of Wikipedia is easily disturbed by the existence of entity alias. In addition, because the dense retrieval requires us to train the model, we actually truncate the paragraph information in Wikipedia for model training and retrieval, so the information that can be used for dense retrieval is also limited. However, from the ZH language, we know that the robustness of the dense retrieval strategy for different languages is better than the sparse retrieval strategy. Therefore, when dealing with retrieval in different languages, we can flexibly choose different strategies based on the quality of the retrieval resources in the corresponding language to obtain better performance.

B.4 Case Study

Table 10 provides a closer examination of the predicted results of BERT-CRF, RaNER, and U-RaNER respectively. We selected three cases from the English language dev data to analyze in detail.

In the first case, fine-grained NER necessitates comprehensive information to accurately classify long-tail entity spans. By utilizing knowledge from multiple sources, U-RaNER successfully predicts "pudendal nerve entrapment" in the first case.

Metric	BN	DE	PT	SV	ZH	AVG.
Recall-Sparse	79.32	71.09	98.22	98.20	37.76	76.92
Recall-Dense	93.26	85.18	87.84	89.19	79.80	**85.25**
F1-Baseline	86.98	85.46	76.54	78.48	75.11	80.51
F1-Sparse	89.81	90.57	83.61	84.43	77.30	**85.14**
F1-Dense	88.45	89.83	77.23	80.54	78.00	82.81

Table 11: Comparison of retrieval performance and impact on NER between the sparse and dense `TEXT2ENT` strategies on the dev set.

In the second case, RaNER’s typical ambiguity problem is evident, where the context retrieved from merely Wikipedia source lacks pertinent information about the target entity "gloucestershire" which could refer to either a county or a sports club.

However, in the third case, the retrieval-based systems wrongly predict "theles leites" and "jesse taylor" as "Athlete" due to retrieved knowledge indicating that they are both mixed martial arts fighters. This demonstrates that the use of retrieved information can sometimes be misleading and even harmful.