A Simple Yet Effective Subsequence-Enhanced Approach for Cross-Domain NER

Jinpeng Hu[1], DanDan Guo[2], Yang Liu[1,*], Zhuo Li[1], Zhihong Chen[1], Xiang Wan[1,3,*], Tsung-Hui Chang[1,2]

[1]Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong, Shenzhen, Guangdong, China
[2]The Chinese University of Hong Kong, Shenzhen
[3]Pazhou Lab, Guangzhou, 510330, China
{jinpenghu, zhuoli3, zhihongchen, yangliu5}@link.cuhk.edu.cn,
{guodandan, changtsunghui}@cuhk.edu.cn, wanxiang@sribd.cn

Abstract

Cross-domain named entity recognition (NER), aiming to address the limitation of labeled resources in the target domain, is a challenging yet important task. Most existing studies alleviate the data discrepancy across different domains at the coarse level via combing NER with language modelings or introducing domain-adaptive pre-training (DAPT). Notably, source and target domains tend to share more fine-grained local information within denser subsequences than global information within the whole sequence, such that subsequence features are easier to transfer, which has not been explored well. Besides, compared to token-level representation, subsequence-level information can help the model distinguish different meanings of the same word in different domains. In this paper, we propose to incorporate subsequence-level features for promoting the cross-domain NER. In detail, we first utilize a pre-trained encoder to extract the global information. Then, we re-express each sentence as a group of subsequences and propose a novel bidirectional memory recurrent unit (BMRU) to capture features from the subsequences. Finally, an adaptive coupling unit (ACU) is proposed to combine global information and subsequence features for predicting entity labels. Experimental results on several benchmark datasets illustrate the effectiveness of our model, which achieves considerable improvements.

Introduction

Named entity recognition (NER) is a fundamental task in text processing, which provides the necessary elements for many downstream tasks, such as relation extraction (Liu et al. 2022b,a), knowledge graph (Fan et al. 2020), summarization (Hu et al. 2022a), etc. Due to the lack of labeled datasets and the expensive cost of human labeling, cross-domain NER has attracted substantial attention over the past years. It aims to adapt the model learned from a source domain with relatively large data to a target domain with limited data.

Recently, many approaches have been proposed for improving the cross-domain NER (Jia, Liang, and Zhang 2019; Liu et al. 2020, 2021b; Jia and Zhang 2020). For example, (Jia, Liang, and Zhang 2019) built a parameter generation network to perform the cross-domain and cross-task knowledge transfer, combining the language modeling and NER. Afterward, (Liu et al. 2021b) introduced a new cross-domain NER dataset containing five diverse domains and provided a domain-related corpus, which can be used to train language models for improving NER in the target domain. Although these efforts can reduce the domain discrepancy, they typically focus on improving the extraction of sentence-level features that belong to the category of coarse-grained level information and pay less attention to exploring the dense fine-grained subsequence information. From the perspective of data sparsity, the coarse-grained level features are more sparse than the fine-grained level ones between two different domains. For example, finding the same sentences in two datasets belonging to distinct domains is almost impossible, while identical subsequences are easier to find. Although these subsequences belong to different domains, they tend to share similar semantic meanings, which is why we believe that subsequence-level features are easier to transfer between different domains. Besides, there are also some studies focusing on single token transfer (Lin and Lu 2018). However, some tokens usually have different meanings when they appear in different domains. For example, the token "nuclear" from the news domain is usually attended to the "powers" and "disarmament" with a larger probability. However, due to limited training data in the target domain in cross-domain NER, the high correlation between "nuclear" and "powers" might still be preserved, which is not the case in the medical domain. Instead, "nuclear" tends to have close relation with "factor" or "transcription". Thus, the neighborhood of a token (e.g., subsequence feature) is also crucial to provide strong evidence to help the model distinguish similarities and differences between different domains.

In this paper, we propose to incorporate subsequence-level features to improve the feature adaptation for cross-domain NER. Specially, we first utilize a pre-trained model to extract coarse-level features from the sequence. Then, we split the sequence as a group of subsequences with the same length as the sequence, and each token in the sequence has a corresponding subsequence. Afterward, we utilize a bidirectional sliding window to extract fine-grained local features from these subsequences. Finally, we propose an adaptive coupling unit (ACU) to integrate fine-grained subsequence features and coarse-grained global information to predict the NER labels. In doing so, for one thing, denser subsequence-level information can be more effectively transferred from the

*Corresponding author.

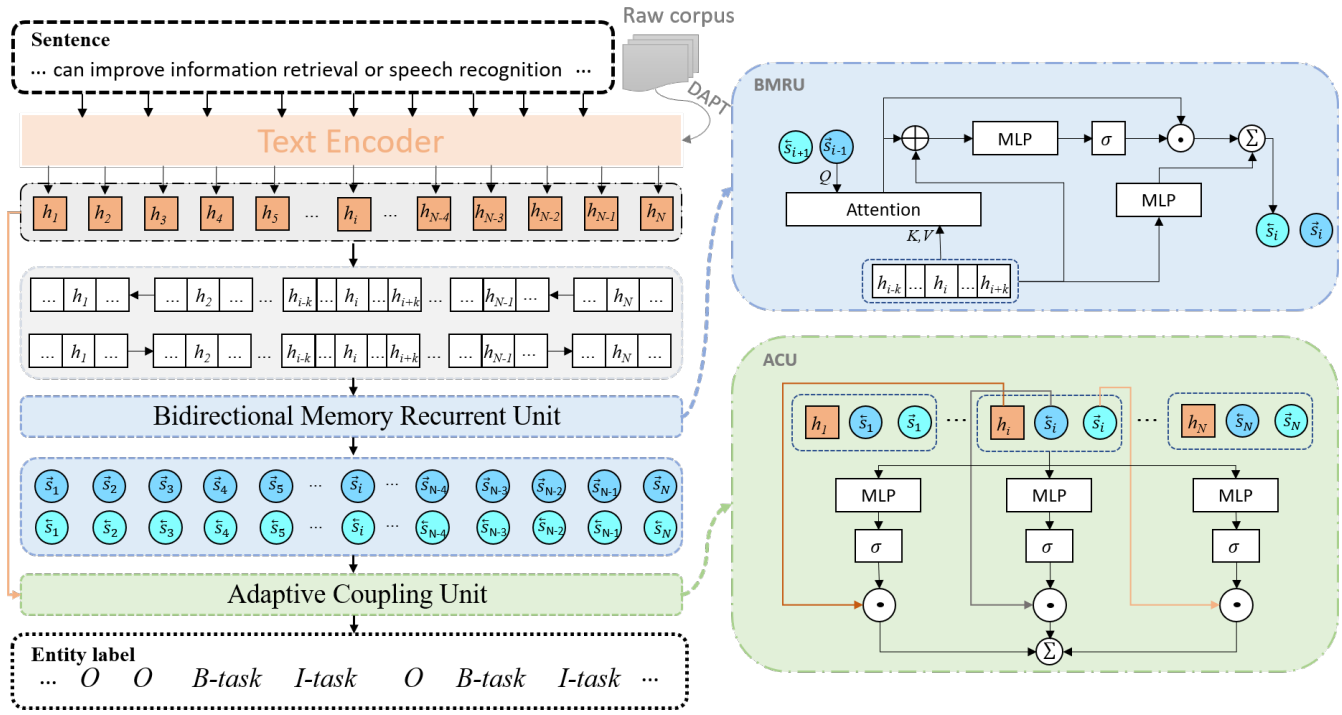

Figure 1: The overall architecture of the proposed model with an example input and output includes text encoder, bidirectional memory recurrent unit (BMRU), and the adaptive coupling unit (ACU). The internal details of the BMRU and ACU are shown in blue and green boxes on the right side of the figure, respectively.

source to the target domain. For another, our model pays more attention to the fine-grained local semantics of each token and thus better distinguishes the meaning of each token between different domains, especially for those tokens that have distinct meanings when they are in different domains. Experiment results on several benchmark datasets illustrate the effectiveness of our proposed model, which outperforms all strong baselines and achieves state-of-the-art performance on most datasets.

Method

We follow the standard sequence labeling paradigm for this task. Specifically, given an input sequence $\mathcal{X}_d = \{x_1, x_2, \cdots, x_n\}$ with n tokens, we annotate its corresponding entity label sequence as $\mathcal{Y}_d = \{y_1, y_2, \cdots, y_n\}$, where $d \in \{src, tgt\}$, src and tgt indicate the source and target domains, respectively. An overview of our proposed model is depicted in Figure 1. It contains three major modules, i.e., the text encoder for extracting the coarse-grained global features, the bidirectional memory recurrent unit (BMRU) for distilling the subsequence-level features, and the adaptive coupling unit (ACU) for combining these features. Below, we will provide more details about these components.

Text Encoder

Pre-trained models (Devlin et al. 2019) have proved their effectiveness in various downstream natural language processing (NLP) tasks, because of their strong ability in feature extraction. Therefore, in our model, we also adopt the pre-trained BERT as our text encoder to extract features from the input sequence:

$$[\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_n] = \text{Encoder}(x_1, x_2, \cdots, x_n), \quad (1)$$

where Encoder refers to a pre-trained encoder, extracting a d-dimensional feature vector $\mathbf{h}_i \in \mathbb{R}^d$ for each token x_i. Owing to the characteristic of canonical point-wise dot-product self-attention in Transformer, these vectors tend to have better coarse-level global information and are limited in fine-grained sensitivity (Li et al. 2019).

Bidirectional Memory Recurrent Unit

As discussed above, capturing the subsequence-level information is inherently important and beneficial for the cross-domain NER, which has not been exploited well. In this section, we show how to extract such features. For each token, we utilize its surrounding tokens to extract fine-grained local features. In detail, we construct a subsequence for each token x_i in $\mathcal{X}_d$ with its localized contextual feature vectors in Eq. (1). Therefore, we can obtain a group of subsequences $\hat{\mathcal{X}}_d$, denoted as

$$\begin{aligned} \hat{\mathbf{h}}_i &= [\mathbf{h}_{i-k}, \cdots, \mathbf{h}_i, \cdots, \mathbf{h}_{i+k}], \\ \hat{\mathcal{X}}_d &= [\hat{\mathbf{h}}_0, \cdots, \hat{\mathbf{h}}_i, \cdots, \hat{\mathbf{h}}_n], \end{aligned} \quad (2)$$

where k is a hyper-parameter to control the grain degree of the local features (i.e., the length of subsequence). Note that we pad k zero vectors at the beginning and end of the $\mathbf{h}$ to keep all subsequences in $\hat{\mathcal{X}}_d$ have the same size. For example, assume we set k equal 3, the subsequence for $\hat{\mathbf{h}}_1$ is $[\mathbf{0}, \mathbf{h}_1, \mathbf{h}_2]$.

Dataset	Train			Val			Test			#Ent.T
	#Sent.	#Ent.	#Ave.E	#Sent.	#Ent.	#Ave.E	#Sent.	#Ent.	#Ave.E	
CoNLL03	15.0k	23.4k	1.56	3.5k	5.9k	1.71	3.7k	5.6k	1.53	4
POLITICS	0.2k	1.3k	6.52	0.5k	3.5k	6.44	0.7k	4.2k	6.47	9
SCIENCE	0.2k	1.1k	5.38	0.5k	2.5k	5.64	0.5k	3.1k	5.69	17
MUSIC	0.1k	0.6k	6.48	0.4k	2.7k	7.05	0.5k	3.3k	7.17	13
LITERATURE	0.1k	0.5k	5.41	0.4k	2.1k	5.25	0.4k	2.3k	5.45	12
AI	0.1k	0.5k	5.32	0.4k	1.5k	4.43	0.4k	1.8k	4.20	14
MOVIE	7.8k	23.0k	2.95	-	-	-	2.0k	5.7k	2.85	14
RESTAURANT	7.7k	15.4k	2.01	-	-	-	1.5k	3.2k	2.13	8

Table 1: The statistics of datasets, including the number of sentences (#Sent.), the number of entities (#Ent.), the averaged entity per sentence (#Ave.E) and the number of entity types (#Ent.T).

Afterward, a BMRU is proposed to take the these subsequences as the input and extract bidirectional sequential features $\overrightarrow{\mathbf{s}}_i \in \mathbb{R}^d$ and $\overleftarrow{\mathbf{s}}_i \in \mathbb{R}^d$ for each token $\hat{\mathbf{h}}_i$. To be more specific, we provide the details of a forward memory recurrent unit (FMRU). We first obtain an overall representation of each $\hat{\mathbf{h}}_i$ and the most straightforward way of doing so is to concatenate each $\mathbf{h}_j$ in $\hat{\mathbf{h}}_i$ by

$$\mathbf{g}_i = \oplus_{\mathbf{h}_j \in \hat{\mathbf{h}}_i} \mathbf{h}_j. \tag{3}$$

The $\mathbf{g}_i$ is then used to generate the current state through $\overrightarrow{\mathbf{c}}_i = \text{MLP}(\mathbf{g}_i) \in \mathbb{R}^d$. In addition, to select the most information-carrying features, we also consider previous output $\overrightarrow{\mathbf{s}}_{i-1}$ of the FMRU with the attention mechanism, stated as

$$\overrightarrow{\mathbf{a}}_i = \text{softmax}(\frac{\overrightarrow{\mathbf{s}}_{i-1} \cdot \hat{\mathbf{h}}_i^\top}{\sqrt{d}}), \tag{4}$$

where "$\cdot$" denotes matrix multiplication, and $\overrightarrow{\mathbf{a}}_i$ is a $2k+1$-dimensional probability vector over $\hat{\mathbf{h}}_i$. Therefore, the forward memory state $\overrightarrow{\mathbf{m}}_i \in \mathbb{R}^d$ is calculated with

$$\overrightarrow{\mathbf{m}}_i = \sum_{\mathbf{h}_j \in \hat{\mathbf{h}}_i} \overrightarrow{\mathbf{a}}_{ij} \mathbf{h}_j. \tag{5}$$

In order to control the flow of the memory state on the current revised input, we utilize a multi-layer perceptron (MLP) to construct a gate from the concatenation of $\overrightarrow{\mathbf{m}}_i$ and $\mathbf{g}_i$, formulated as:

$$\overrightarrow{\mathbf{r}}_i = \sigma(\mathbf{W}_r \cdot [\overrightarrow{\mathbf{m}}_i, \mathbf{g}_i] + \mathbf{b}_r), \tag{6}$$

where $\mathbf{W}_r$ and $\mathbf{b}_r$ are learnable parameters and σ is the *sigmoid* function. Finally, we obtain the forward localized feature of $\hat{\mathbf{h}}_i$ via combining both current state $\mathbf{c}_i$ and memory state $\mathbf{m}_i$:

$$\overrightarrow{\mathbf{s}}_i = \overrightarrow{\mathbf{c}}_i + \overrightarrow{\mathbf{r}}_i \overrightarrow{\mathbf{m}}_i. \tag{7}$$

Meanwhile, we obtain backward feature $\overleftarrow{\mathbf{s}}_i$ through the similar way, which is computed by $\overleftarrow{\mathbf{s}}_{i+1}$ and $\hat{\mathbf{h}}_i$.

Thus, we conclude that subsequence feature extraction is formulated as:

$$[\overrightarrow{\mathbf{s}}_1, \overrightarrow{\mathbf{s}}_2, \cdots, \overrightarrow{\mathbf{s}}_n] = \text{FMRU}(\mathbf{g}_1, \cdots, \mathbf{g}_n), \tag{8}$$

$$[\overleftarrow{\mathbf{s}}_1, \overleftarrow{\mathbf{s}}_2, \cdots, \overleftarrow{\mathbf{s}}_n] = \text{AMRU}(\mathbf{g}_n, \cdots, \mathbf{g}_1), \tag{9}$$

where FMRU and AMRU are the forward and backward memory recurrent unit, respectively. Therefore, $\overrightarrow{\mathbf{s}}_i$ and $\overleftarrow{\mathbf{s}}_i$ can be regarded as subsequence-level features, and they also memorize previous and later subsequence information to some extent, respectively.

Adaptive Coupling Unit

To incorporate both coarse-grained global and fine-grained subsequence features to facilitate the label prediction, we propose an adaptive coupling unit to dynamically combine $\overrightarrow{\mathbf{s}}_i$, $\overleftarrow{\mathbf{s}}_i$ and $\mathbf{h}_i$. In detail, we concatenate these three vectors and build three different gates:

$$\begin{aligned} \mathbf{p}_1 &= \sigma(\mathbf{W}_1[\overrightarrow{\mathbf{s}}_i, \overleftarrow{\mathbf{s}}_i, \mathbf{h}_i] + \mathbf{b}_1), \\ \mathbf{p}_2 &= \sigma(\mathbf{W}_2[\overrightarrow{\mathbf{s}}_i, \overleftarrow{\mathbf{s}}_i, \mathbf{h}_i] + \mathbf{b}_2), \\ \mathbf{p}_3 &= \sigma(\mathbf{W}_3[\overrightarrow{\mathbf{s}}_i, \overleftarrow{\mathbf{s}}_i, \mathbf{h}_i] + \mathbf{b}_3), \end{aligned} \tag{10}$$

where $\mathbf{W}_1$, $\mathbf{W}_2$, $\mathbf{W}_3$, $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$ are learnable parameters and $\mathbf{p}_1$, $\mathbf{p}_2$ and $\mathbf{p}_3$ are three different gates. Each gate can automatically select appropriate information from the corresponding features, and then we add up all these selected features, stated as:

$$\mathbf{u}_i = \mathbf{p}_1 \overrightarrow{\mathbf{s}}_i + \mathbf{p}_2 \overleftarrow{\mathbf{s}}_i + \mathbf{p}_3 \mathbf{h}_i, \tag{11}$$

where coupling vector $\mathbf{u}_i \in \mathbb{R}^d$ is the final representation of token x_i. A trainable fully connected layer is used to align its dimension to the output space by $\mathbf{e}_i = \mathbf{W}_o \cdot \mathbf{u}_i + \mathbf{b}_o$, where $\mathbf{e}_i \in \mathbb{R}^{\mathcal{C}}$ and $\mathcal{C}$ denotes the set of all types of named entity (NE) labels. Then, the softmax function is applied on $\mathbf{e}_i$ to predict the label y_i by:

$$y_i = \arg\max \frac{\exp(e_i^c)}{\sum_{j=1}^{|\mathcal{C}|} \exp(e_i^j)}. \tag{12}$$

Transfer Training Process

We employ a simple parameter initialization method to assist the model in enhancing knowledge transfer. To record the source domain information, we first fine-tune our model on the source domain dataset by $\mathcal{X}_{src}$ and $\mathcal{Y}_{src}$. After that, we initialize a target model with the learned parameters and continue training the model on the target domain with the labeled datasets $\mathcal{X}_{tgt}$ and $\mathcal{Y}_{tgt}$. In doing so, valuable information learned from the source domain can be effectively adapted to the target domains.

Moreover, motivated by (Liu et al. 2021b; Gururangan et al. 2020), we continue pre-training the BERT on the target

Model	Politics	Science	Music	CoNLL03 Literature	AI	Movie	Restaurant
BiLSTM-CRF	56.60	49.97	44.79	43.03	43.56	68.31	78.13
Cross-domain LM	68.44	64.31	63.56	59.59	53.70	-	-
Coach	61.50	52.09	51.66	48.35	45.15	-	-
Flair	69.54	64.71	65.60	61.35	52.48	-	-
Multi-Cell LSTM	70.56	66.42	70.52	66.96	58.28	69.41	78.67
LST-NER	70.44	66.83	72.08	**67.12**	60.32	70.25	78.74
Ours	**71.31**	**68.65**	**72.42**	67.05	**60.89**	**71.19**	**79.20**
Multi-Cell LSTM+DAPT	71.45	67.68	74.19	68.63	61.64	-	-
BERT+DAPT	72.05	68.78	75.71	69.04	62.56	-	-
LST-NER+DAPT	73.25	70.07	76.83	**70.76**	63.28	-	-
Ours (DAPT)	**73.82**	**71.17**	**79.28**	69.22	**63.79**	-	-

Table 2: The performance of existing studies and our proposed models with respect to F1 score. Results are averaged over three runs with different seeds.

domain-related corpus to narrow the background discrepancy between different domains.

Experimental Setting

Dataset

To validate the effectiveness of our proposed model, we employ the following datasets in our experiments. We regard the Conll2003 as the source domain and other datasets as the target domains.

- **Conll2003** (Sang and De Meulder 2003) is a commonly used NER dataset collected from Reuters News, containing person, location, organization, and miscellaneous entity categories.
- **MIT Movie** (Movie) (Liu et al. 2013b) is a movie domain NER dataset, including award, title, opinion, year, origin, genre, director, plot, quote, actor, soundtrack, character, and others.
- **CrossNER** (Liu et al. 2021b) is collected from Wikipedia and split into five diverse domains with Wikipedia's categories, including politics, natural science, music, literature, and artificial intelligence (AI). Besides, it also collects corresponding unlabeled corpora in each domain for continuing pre-training language models.
- **MIT Restaurant** (Restaurant) (Liu et al. 2013a) is a dataset for restaurant review and contains eight types of entities.

We follow the official split for these datasets, with their statistics summarized in Table 1.

Baselines and Evaluation Metric

To explore the advantages of our proposed model, we compare it with the following baselines. **BiLSTM-CRF** (Lample et al. 2016) utilizes a bidirectional LSTM with a sequential conditional random layer above and combines both character-level and word-level representation to enhance NER. **Cross-domain LM** (Jia, Liang, and Zhang 2019) incorporates cross-domain language modeling as a bridge to reduce domain discrepancy and improve knowledge transfer. **Coach** (Liu et al. 2020) proposes a cross-domain slot filling framework, which first learns slot entity patterns and then combines the features for each slot entity and predicts the types of detected entities. **Flair** (Akbik, Blythe, and Vollgraf 2018) leverages the internal states of a character language model to produce contextual string embedding for enhancing NER. **Multi-Cell LSTM** (Jia and Zhang 2020) proposes a multi-cell compositional LSTM structure for cross-domain NER, incorporating BERT representations (Devlin et al. 2019) and entity type information. **BERT+DAPT** (Liu et al. 2021b) uses the domain-related corpus to continue training the language model before NER. **LST-NER** (Zheng, Chen, and Ma 2022) proposes to learn graph structure via matching label graphs from source to target domain for improving cross-domain NER.

For a fair comparison, we follow previous studies to use the F1 score to evaluate model performance.

Implementation Details

In our experiments, we use the standard BIO scheme to label NEs. We utilize a pre-trained language model (i.e., bert-base-cased) as the text encoder to extract features from input sequence[1]. We follow their default model setting: we use 12 layers of self-attention with 768-dimensional embeddings. Besides, the hidden size in BMRU is set to 768 for each direction with its parameters initialized randomly. We use Adam (Kingma and Ba 2015) to optimize all trainable parameters in the model by minimizing the negative log-likelihood, including those in the pre-trained text encoder. k is set to 7 in our experiments. More detailed hyperparameters are reported in Appendix. For transfer training, we first train the model on the source domain data with 2 epochs for Conll2003 and then fine-tune the model to the target domain.

Results and Analyses

Overall Results

To illustrate the effectiveness of our models, we compare our model with existing studies on the same datasets, with all results (i.e., F1 score) reported in Table 2. There are several observations drawn from different aspects. First, when

[1]https://github.com/google-research/bert.

Model	CoNLL03				
	Pol.	Sci.	Mus.	Lit.	AI
BERT‡	68.71	64.94	68.30	63.63	58.88
Subsequence-level feature extractor					
GRU	69.60	67.83	70.03	63.37	60.19
LSTM	69.87	68.64	70.24	66.20	59.75
BMRU	**71.31**	**68.65**	**72.42**	**67.05**	**60.89**
Subsequence-level features Combination					
ADD	70.25	67.23	71.62	65.36	59.79
CONCAT	71.08	68.28	71.96	66.92	60.00
ACU	**71.31**	**68.65**	**72.42**	**67.05**	**60.89**

Table 3: The performance of baseline and our proposed models in terms of F1 score. We compare models using different structures to extract fine-grained local features (e.g., LSTM, BMRU). ADD, CONCAT and ACU represent three methods to combine fine-grained and coarse-grained features. ‡ refers to that the result is directed cited from (Liu et al. 2021b).

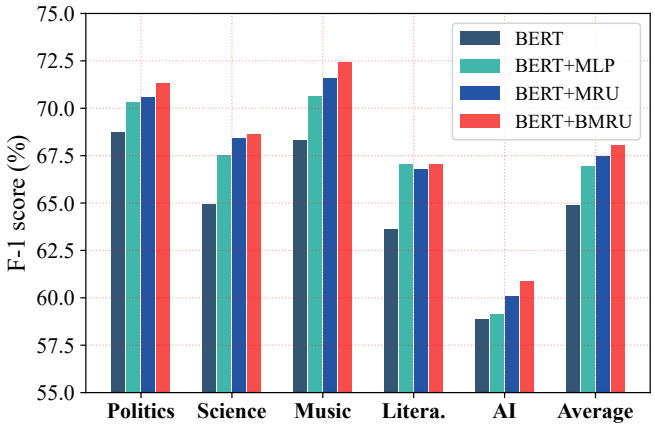

Figure 2: F1 score of models using different structures to capture fine-grained features. MLP represents extracting features using MLP without recurrent structure (memory states) within the revised input sentence; FMRU is a forward memory-based recurrent unit.

comparing our model and those models that incorporate language modeling as an auxiliary task (Jia and Zhang 2020; Jia, Liang, and Zhang 2019), we can observe that our model can achieve better results. This observation indicates that denser subsequence features are effective in cross-domain NER, which is useful to promote feature transfer and can help the model distinguish distinct meanings of tokens in different domains. In addition, language models usually require massive data to train, while the target domains usually have limited training data. Second, the comparison between the former three models (Lample et al. 2016; Jia, Liang, and Zhang 2019; Liu et al. 2020) and the latter three ones (Jia and Zhang (2020); Zheng, Chen, and Ma (2022) and Ours) shows the effectiveness of the pre-trained model in domain transfer, where the former three do not incorporate the BERT, and the latter three utilize the features extracted from BERT. Third, from Table 1, we know that politics, science, music, literature, and AI only have 100 or 200 sentences, and movie and restaurant have more than 7k sentences. Therefore, the former five datasets can be regarded as low-resource cross-domain NER, and the latter two are for rich-resource cross-domain settings. Our model outperforms all baselines in these two different settings on most datasets. This result further illustrates the effectiveness of subsequence features, which can help the model discriminate similarity and discrepancy between source and target text. Fourth, our model confirms its superiority of simplicity when compared to those complicated approaches. For example, LST-NER needs to construct label graphs in source and target spaces and then utilize GCN to extract features from the graph, while our model achieves better results with a rather simpler method.

Moreover, we also compare the models combined with DAPT and the results are also shown in Table 2. We can find that the comparisons between models without DAPT (Jia and Zhang (2020), Liu et al. (2021b), Zheng, Chen, and Ma (2022) and Ours) and those with DAPT (Jia and Zhang (2020)+DAPT, Liu et al. (2021b)+DAPT, Zheng, Chen, and Ma (2022)+DAPT, and Ours+DAPT) illustrates the effectiveness of DAPT, where models with DAPT achieve considerable improvements. This illustrates that continuing pre-training text encoder on a massive domain-related corpus also can further reduce the background discrepancy between different domains and thus bring improvements for these models. Besides, our model can achieve state-of-the-art results among those models on most datasets, further illustrating the validity of subsequence features.

Effect of BMRU and ACU

To further explore the effectiveness of our proposed BMRU and ACU, we conduct experiments on the aforementioned CrossNER datasets. We compare three different approaches to combine subsequence features and coarse-grained global features: direct concatenation, summation, and ACU, respectively. Besides we also compare models using different structures to extract fine-grained features. All the results are reported in Table 3, and there are several observations. First, models utilizing subsequence features extractor obtain better results than their corresponding baseline (i.e., BERT). It indicates the effectiveness of our innovation for incorporating fine-grained subsequence features into the cross-domain NER. The main reason might be that the subsequence feature can enhance the transferability of the feature extractor from source to target domain. It is noticed that BMRU achieves higher F1 scores than LSTM and GRU, confirming that the proposed BRMU can capture better fine-grained features. Second, we observe that ACU achieves superior performance to its competitors (i.e., ADD, CONCAT), suggesting the ACU is effective in automatically balancing different features.

Furthermore, we also conduct some additional experiments to investigate the effect of memory state (i.e., $\overrightarrow{\mathbf{m}}_i$ and $\overleftarrow{\mathbf{m}}_i$) and direction in BMRU. The results are shown in Figure 2, where the BERT+BMRU is our proposed model. By comparing the MLP with FMRU and BMRU, we can see that capturing sequential information between different subsequences (defined in Eq.(2)) can further enhance the subsequence feature extraction, indicating the effectiveness of the memory states.

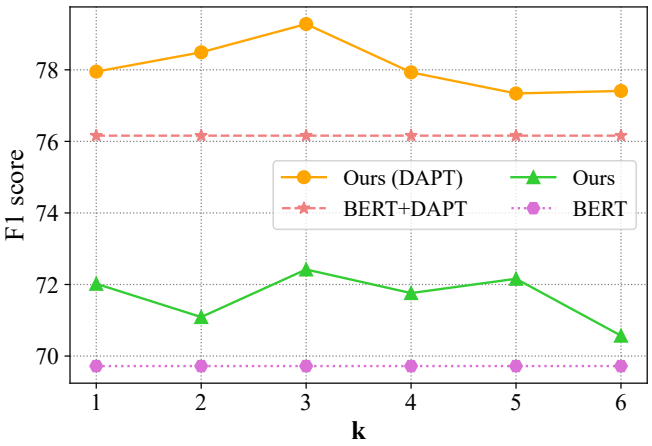

Figure 3: F1 score of models with different k.

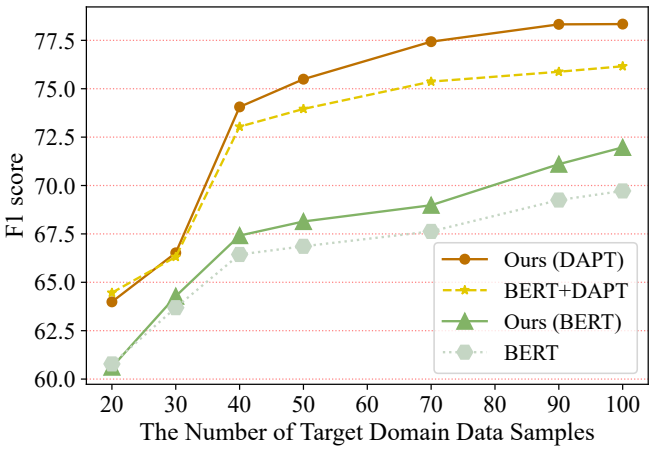

Figure 4: F1 scores of different models varying with the training data size in the target domain.

Moreover, although BMRU may lead to a slight decline in the literature dataset compared to FMRU, BMRU outperforms other baselines on most datasets. This observation illustrates that bidirectional fine-grained local features can provide more valuable information than the single direction in this task.

Effect of k

To explore the impact of the k, we conduct experiments with different k (i.e., $1, 2, 3, 4, 5, 6$) on the music dataset with corresponding lengths of the subsequence $3, 5, 7, 9, 11, 13$. The results are reported in Figure 3. It is found that our proposed models (i.e., Ours and Ours (DAPT)) achieve better performance than corresponding baselines (i.e., BERT and BERT+DAPT), regardless of which k we choose, further demonstrating the effectiveness of incorporating fine-grained subsequence-level features. Besides, we can observe that when k equals 3 (i.e., subsequence length equals 7), our models obtain the best results. This result reveals that features extracted from subsequences with seven words are appropriate to help the model distinguish differences and similarities.

Fine-Grained Comparison

In this section, we further explore the effect of our proposed model in fine-grained comparison, aiming to show how our model achieves better results on the entity type level. The results are reported in Table 4. It is found that our model obtains better results on most entity types, especially for the domain-related entity categories, such as song and album, which achieves significant improvements. An explanation for this observation may be that conll2003 is collected from the Reuters news domain and contains few music-related sentences, so it is difficult to transfer sentence-level information to help music-related entity recognition. In contrast, fine-grained subsequences tend to have more overlaps between source and target domains, and thus they can transfer more valuable features from source to target domains, contributing to significant improvements in these entity types.

Effect of Target Domain Data Size

To explore the impact of the target domain data size, we conduct experiments on different amounts of training data. We utilize the Conll2003 as the source domain and music domain dataset as the target domain, with the number of target domain training data gradually increasing from 20 to 100 samples, where we visualize the results in Figure 4. It is found that our proposed model obtains better performance than their baselines in most groups, further confirming the effectiveness of our innovation in combining coarse-grained global and fine-grained subsequence features for improving NER performance in the target domain. In addition, we can see that with the increase in data size, all these models gradually have better performance, which illustrates the importance of data scale in cross-domain NER.

Case Study

As shown in Table 5, we also give a case study with predicted labels from different models. We can observe that our model can better recognize entities that are close to each other, where both Ours and Ours (DAPT) correctly identify the last four entities, which are almost adjacent in the sentence. This is because our proposed BMRU can extract fine-grained local features from subsequences. Therefore, our model will draw more attention to neighbor words and phrases when predicting the current token, which helps recognize the adjacent entities. For example, the term "four singles" is helpful for models to recognize the adjacent phrase "By the Way" as a song entity, and this knowledge will further benefit other adjacent entity recognition. Moreover, we can find that both BERT and Ours tend to assign all the entities to the same category. However, BERT+DAPT and Ours (DAPT) can recognize these entities into two classes, though BERT+DAPT misclassifies some song entities as album entities(i.e., the second "By the Way" and "The Zephyr Song "). This is because these entities and their background information are mentioned in the domain-related data such that DAPT can improve model inference ability in the target domain.

In-Domain NER

We also investigate the performance of our model on general NER, where we only utilize single domain datasets to train the model, with the results reported in Table 6. The

MODEL	ORG.	MISC.	LOC.	PER.	COU.	ALB.	AWA.	BAN.	SON.	INS.	ART.
BERT	71.11	30.44	78.60	8.71	88.50	61.63	78.47	75.65	25.46	26.53	81.77
OURS	73.01	28.11	77.66	9.32	86.80	67.28	79.32	78.64	43.91	30.52	83.34
BERT+DAPT	77.50	30.92	82.78	13.55	85.98	73.23	81.17	84.85	66.94	42.96	86.89
OURS (DAPT)	77.68	29.41	83.25	12.92	86.79	78.45	82.93	86.06	77.08	44.40	87.30

Table 4: F1 scores of fine-grained comparisons on music datasets. ORG., MISC., LOC., PER., COU., ALB., AWA., BAN., SON., INS. and ART. denote Organization, Miscellaneous, Location, Person, Country, Album, Award, Band, Song, Musical Instrument and Musical Artist, respectively.

Sentence	**By the Way** was released and produced four singles; **By the Way**, **The Zephyr Song**, **Can't Stop** and **Universally Speaking**.				
BERT	By the Way (**Album**)	By the Way (**Album**)	The Zephyr Song (**Album**)	Can't Stop (**Album**)	Universally Speaking (**Album**)
OURS	By the Way (**Song**)	By the Way (**Song**)	The Zephyr Song (**Song**)	Can't Stop (**Song**)	Universally Speaking (**Song**)
DAPT	By the Way (**Album**)	By the Way (**Album**)	The Zephyr Song (**Album**)	Can't Stop (**Song**)	Universally Speaking (**Song**)
OURS (DAPT)	By the Way (**Album**)	By the Way (**Song**)	The Zephyr Song (**Song**)	Can't Stop (**Song**)	Universally Speaking (**Song**)

Table 5: Example on music test set. Entities with and without underlining represent incorrect and correct entities, respectively.

MODEL	POLITICS	SCIENCE	MUSIC	LITERATURE	AI
BERT‡	66.56	63.73	66.59	59.95	50.37
OURS	**67.58**	**65.64**	**67.30**	**61.20**	**51.70**

Table 6: F1 scores on general NER.

comparisons between our proposed model and its baselines demonstrate that fine-grained features also can improve the general NER model. The main reason could be that subsequence information can help the model grasp local structures of tokens, which assists the model in obtaining a better understanding of the text. Besides, when compared between cross-domain and in-domain NER, it is observed that the improvements gained from cross-domain NER over BERT in Table 3 are larger than that of in-domain NER, which reveals that our model is better at cross-domain settings. This might be because our proposed model can distinguish the differences and similarities between different text styles with the help of subsequence feature transfer, while this advantage may not be useful for in-domain NER.

Related Work

Neural networks have played dominant roles in the NER task over the past few years (Huang, Xu, and Yu 2015; Chiu and Nichols 2016; Liu et al. 2021a), which achieves great improvements on this task. Recently, models (Devlin et al. 2019; Luo, Xiao, and Zhao 2020; Yang et al. 2019; Lee et al. 2020; Liu et al. 2019; Yamada et al. 2020; Hu et al. 2022b) based on Transformer (Vaswani et al. 2017) have become the mainstream methods to realize NER since they can provide more effective representation with the help of the multi-head attention mechanism. Luo, Xiao, and Zhao (2020) proposed to utilize two-level hierarchical contextualized representation, including sentence-level and document-level representations, to fuse with token embedding to improve the performance.

Domain adaption in NER has been a popular topic in past decades, and many works focus on this problem Jia, Liang, and Zhang (2019); Zhang, Yue, and Zhuang (2020); Liu et al. (2021b); Chen and Moschitti (2019); Hu et al. (2022c). For example, Lin and Lu (2018) employed a lightweight transfer learning for cross-domain NER, which uses adaptation layers to bridge the gap between the two input spaces. Jia, Liang, and Zhang (2019) combined language modeling and NER tasks in source and target domains via multi-task learning to enhance the model performance. Liu et al. (2021b) utilized the domain-related unlabeled corpus to continue pre-training language modeling and thus improved its domain adaptation ability. Chen and Moschitti (2019) used a neural adapter, which connects the target and the source models, to mitigate the forgetting of the learned knowledge. Most of these approaches construct adapters or combine them with other tasks via multi-task learning to reduce the coarse-level difference between source and target domains. However, these methods pay less attention to leveraging fine-grained features, which play an important role in this task since these features are more accessible and more effective in transferring from source to target domains. Our model provides an alternative solution to effectively combine coarse-grained and fine-grained level features, thus reducing the data discrepancy and robustly improving cross-domain NER.

Conclusion

In this paper, we propose to incorporate denser subsequence-level features for improving cross-domain NER. In detail, for each token, we generate a subsequence constructed by its surrounding tokens, and thus for the input sequence, we can obtain a group of subsequences. Then, we utilize BMRU to extract fine-grained subsequence features from the groups. Finally, we propose an ACU module to fuse coarse-grained global information from the pre-trained encoder and fine-grained sequence features. In doing so, dense subsequence-level features can promote valuable information transferring from the source to the target domain. Experimental results on several benchmark datasets illustrate the effectiveness of our model, which achieves considerable improvements.

Acknowledgments

This work is supported by Chinese Key-Area Research and Development Program of Guangdong Province (2020B0101350001), the Guangdong Provincial Key Laboratory of Big Data Computing, The Chinese University of Hong Kong, Shenzhen and the Shenzhen Fundamental Research Fundunder Grant JCYJ20220818103001002.

References

Akbik, A.; Blythe, D.; and Vollgraf, R. 2018. Contextual String Embeddings for Sequence Labeling. In *Proceedings of the 27th international conference on computational linguistics*, 1638–1649.

Chen, L.; and Moschitti, A. 2019. Transfer Learning for Sequence Labeling Using Source Model and Target Data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 6260–6267.

Chiu, J. P.; and Nichols, E. 2016. Named Entity Recognition with Bidirectional LSTM-CNNs. *Transactions of the Association for Computational Linguistics*, 4: 357–370.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 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)*, 4171–4186.

Fan, R.; Wang, L.; Yan, J.; Song, W.; Zhu, Y.; and Chen, X. 2020. Deep Learning-Based Named Entity Recognition and Knowledge Graph Construction for Geological Hazards. *ISPRS International Journal of Geo-Information*, 9(1): 15.

Gururangan, S.; Marasović, A.; Swayamdipta, S.; Lo, K.; Beltagy, I.; Downey, D.; and Smith, N. A. 2020. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8342–8360.

Hu, J.; Li, Z.; Chen, Z.; Li, Z.; Wan, X.; and Chang, T.-H. 2022a. Graph Enhanced Contrastive Learning for Radiology Findings Summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 4677–4688.

Hu, J.; Shen, Y.; Liu, Y.; Wan, X.; and Chang, T.-H. 2022b. Hero-Gang Neural Model For Named Entity Recognition. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1924–1936. Seattle, United States: Association for Computational Linguistics.

Hu, J.; Zhao, H.; Guo, D.; Wan, X.; and Chang, T.-H. 2022c. A Label-Aware Autoregressive Framework for Cross-Domain NER. In *Findings of the Association for Computational Linguistics: NAACL 2022*, 2222–2232.

Huang, Z.; Xu, W.; and Yu, K. 2015. Bidirectional LSTM-CRF Models for Sequence Tagging. *arXiv preprint arXiv:1508.01991*.

Jia, C.; Liang, X.; and Zhang, Y. 2019. Cross-domain NER using Cross-Domain Language Modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2464–2474.

Jia, C.; and Zhang, Y. 2020. Multi-Cell Compositional LSTM for NER Domain Adaptation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 5906–5917.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *ICLR (Poster)*.

Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; and Dyer, C. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of NAACL-HLT*, 260–270.

Lee, J.; Yoon, W.; Kim, S.; Kim, D.; Kim, S.; So, C. H.; and Kang, J. 2020. BioBERT: a Pre-trained Biomedical Language Representation Model for Biomedical Text Mining. *Bioinformatics*, 36(4): 1234–1240.

Li, S.; Jin, X.; Xuan, Y.; Zhou, X.; Chen, W.; Wang, Y.-X.; and Yan, X. 2019. Enhancing the Locality and Breaking the Memory Bottleneck of Transformer on Time Series Forecasting. *Advances in Neural Information Processing Systems*, 32: 5243–5253.

Lin, B. Y.; and Lu, W. 2018. Neural Adaptation Layers for Cross-Domain Named Entity Recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2012–2022.

Liu, J.; Pasupat, P.; Cyphers, S.; and Glass, J. 2013a. Asgard: A portable architecture for multilingual dialogue systems. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 8386–8390. IEEE.

Liu, J.; Pasupat, P.; Wang, Y.; Cyphers, S.; and Glass, J. 2013b. Query understanding enhanced by hierarchical parsing structures. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, 72–77. IEEE.

Liu, Y.; Hu, J.; Wan, X.; and Chang, T.-H. 2022a. Learn from Relation Information: Towards Prototype Representation Rectification for Few-Shot Relation Extraction. In *Findings of the Association for Computational Linguistics: NAACL 2022*, 1822–1831.

Liu, Y.; Hu, J.; Wan, X.; and Chang, T.-H. 2022b. A Simple yet Effective Relation Information Guided Approach for Few-Shot Relation Extraction. In *Findings of the Association for Computational Linguistics: ACL 2022*, 757–763.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*.

Liu, Y.; Tian, Y.; Chang, T.-H.; Wu, S.; Wan, X.; and Song, Y. 2021a. Exploring Word Segmentation and Medical Concept Recognition for Chinese Medical Texts. In *Proceedings of the 20th Workshop on Biomedical Language Processing*, 213–220.

Liu, Z.; Winata, G. I.; Xu, P.; and Fung, P. 2020. Coach: A Coarse-to-Fine Approach for Cross-Domain Slot Filling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 19–25.

Liu, Z.; Xu, Y.; Yu, T.; Dai, W.; Ji, Z.; Cahyawijaya, S.; Madotto, A.; and Fung, P. 2021b. CrossNER: Evaluating Cross-Domain Named Entity Recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 13452–13460.

Luo, Y.; Xiao, F.; and Zhao, H. 2020. Hierarchical Contextualized Representation for Named Entity Recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 8441–8448.

Sang, E. T. K.; and De Meulder, F. 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*, 142–147.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is All You Need. In *Advances in neural information processing systems*, 5998–6008.

Yamada, I.; Asai, A.; Shindo, H.; Takeda, H.; and Matsumoto, Y. 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)*, 6442–6454.

Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R. R.; and Le, Q. V. 2019. Xlnet: Generalized Autoregressive Pretraining for Language Understanding. *Advances in neural information processing systems*, 32.

Zhang, K.; Yue, D.; and Zhuang, L. 2020. Improving Chinese Clinical Named Entity Recognition Based on BiLSTM-CRF by Cross-Domain Transfer. In *Proceedings of the 2020 4th High Performance Computing and Cluster Technologies Conference & 2020 3rd International Conference on Big Data and Artificial Intelligence*, 251–256.

Zheng, J.; Chen, H.; and Ma, Q. 2022. Cross-domain Named Entity Recognition via Graph Matching. In *Findings of the Association for Computational Linguistics: ACL 2022*, 2670–2680.