Cross-Domain NER under a Divide-and-Transfer Paradigm

XINGHUA ZHANG, Institute of Information Engineering, Chinese Academy of Sciences, School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
BOWEN YU, Institute of Information Engineering, Chinese Academy of Sciences, School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
XIN CONG, Institute of Information Engineering, Chinese Academy of Sciences, School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
TAOYU SU, Institute of Information Engineering, Chinese Academy of Sciences, School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
QUANGANG LI, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
TINGWEN LIU, Institute of Information Engineering, Chinese Academy of Sciences, School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
HONGBO XU, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China

Cross-domain Named Entity Recognition (NER) transfers knowledge learned from a rich-resource source domain to improve the learning in a low-resource target domain. Most existing works are designed based on the sequence labeling framework, defining entity detection and type prediction as a monolithic process. However, they typically ignore the discrepant transferability of these two sub-tasks: the former locating spans corresponding to entities is largely domain-robust, whereas the latter owns distinct entity types across domains. Combining them into an entangled learning problem may contribute to the complexity of domain transfer. In this work, we propose the novel divide-and-transfer paradigm in which different sub-tasks are learned using separate functional modules for respective cross-domain transfer. To demonstrate the effectiveness of divide-and-transfer, we concretely implement two NER frameworks by applying this paradigm with different cross-domain transfer strategies. Experimental results on 10 different domain pairs show the notable superiority of our proposed frameworks. Experimental analyses indicate that significant advantages of the divide-and-transfer paradigm over prior monolithic ones originate from its better performance on low-resource data and a much greater transferability. It gives us a new insight into cross-domain NER. Our code is available on GitHub.[1]

CCS Concepts: • **Computing methodologies → Information extraction**; **Transfer learning;**

[1] https://github.com/AIRobotZhang/Divide-and-Transfer

This work was supported by the National Key Research and Development Program of China (grant 2021YFB3100600) and the Youth Innovation Promotion Association of CAS (grant 2021153).
Authors' addresses: X. Zhang, B. Yu, X. Cong, T. Su, and T. Liu (Corresponding author), Institute of Information Engineering, Chinese Academy of Sciences, School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China; e-mails: zhangxinghua@iie.ac.cn, yubowen@iie.ac.cn, congxin@iie.ac.cn, sutaoyu@iie.ac.cn, liutingwen@iie.ac.cn; Q. Li and H. Xu, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China; e-mails:liquangang@iie.ac.cn, hbxu@iie.ac.cn.

ACM 1046-8188/2024/05-ART137
https://doi.org/10.1145/3655618

Additional Key Words and Phrases: Named entity recognition, cross-domain transfer, information extraction, knowledge acquisition, task decomposition

ACM Reference Format:
Xinghua Zhang, Bowen Yu, Xin Cong, Taoyu Su, Quangang Li, Tingwen Liu, and Hongbo Xu. 2024. Cross-Domain NER under a Divide-and-Transfer Paradigm. *ACM Trans. Inf. Syst.* 42, 5, Article 137 (May 2024), 32 pages. https://doi.org/10.1145/3655618

1 INTRODUCTION

Named Entity Recognition (NER) aims to detect entity spans and classify them into pre-defined categories (e.g., location), which has achieved notable performance using a large amount of high-quality labeled data. Yet performance tends to drop drastically when lacking sufficient annotated data. Cross-domain NER solves this issue by transferring knowledge from the high-resource to low-resource domains, attracting increasing research interest.

End-to-end NER sequence labeling has always been a popular paradigm with compositional tagging schemes (e.g., B-LOC), as shown in Figure 1(a). Prior cross-domain NER methods also follow this framework and introduce corresponding transfer strategies, such as parameter transfer [21, 35] and domain mapping [3, 19]. Note that the sequence labeling framework is monolithic, as it needs to recognize entity span and classify entity category concurrently. So cross-domain NER under this monolithic framework needs to transfer two kinds of coupled information (entity span and type) simultaneously. However, these two information have different cross-domain transfer difficulties: the domain gap of being entity span is small due to the same label space across domains, whereas the entity type set is distinct across domains, which causes a great obstacle for transfer, and this hybrid transfer seems to be challenging. Thus, this article focuses on disentangling the coupled information by *dividing* the NER task into entity detection and type prediction sub-tasks (see Figure 1(b) and (c)), and devises corresponding *transfer* strategies according to the cross-domain barrier in each sub-task for more effective transfer (*divide-and-transfer*). Finally, outputs from these two sub-tasks are integrated as the final result of the original NER task.

Methodologically, we perform the cross-domain transfer in these two sub-tasks, respectively, and propose two instantiated frameworks following by *D*ivide-and-*Trans*fer paradigm, namely *DTrans-SMix* and *DTrans-MPrompt*.

DTrans-*SMix* is our first attempt in which we adopt parameter *S*haring and *Mix*up strategies for cross-domain transfer. Concretely, we use two individual encoders to extract distinct contextual features from entity detection and type prediction sub-tasks separately. The corresponding cross-domain transfer strategies for two sub-tasks are as follows. First, the entity detection sub-task is domain-robust, which has a common label set across domains and seeks to locate entities. For simplicity, we share all model parameters (i.e., *Embedding*, *Encoding*, and *Output* layer) to jointly train between the source and target domain for transfer. Second, the type prediction sub-task aims to classify the located entity spans with pre-defined entity categories. However, category sets are different across domains, leaving classification heads in output layers unshareable, which leads to the obvious domain discrepancy and transfer barrier. To tackle this challenge, an intermediate augmented domain is constructed by a fixed ratio-based mixup on the top of encoder representations between source and target domain, then we send intermediate features into a new classification head to minimize the domain gap.

DTrans-*MPrompt* serves as another instantiation of the proposed divide-and-transfer paradigm in this article, which performs cross-domain transfer with a *M*ulti-view decoding strategy and *Prompt* tuning. Detailed transfer strategies in two sub-tasks are as follows.

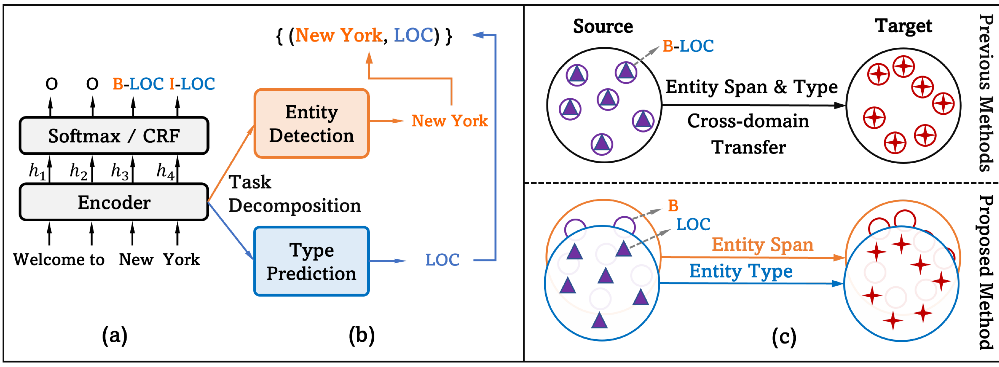

Fig. 1. (a) End-to-end NER sequence labeling framework. (b) Entity detection and type prediction sub-task. (c) Previous methods try to transfer directly without any consideration of coupled information. Our proposed method disentangles the coupled entity span and type information.

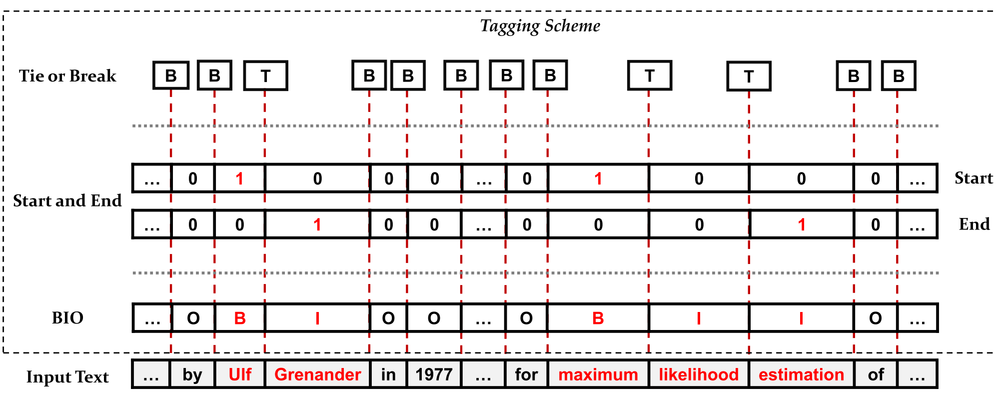

Fig. 2. Various tagging schemes in the entity detection sub-task.

Entity Detection. In *Entity detection* which owns the same label space across domains, we propose a multi-view decoding strategy with three tagging schemes (shown in Figure 2) during training, which can capture more domain-invariant features about what is an entity. These tagging schemes can detect entity spans from various perspectives (e.g., entity boundary, inside): "BIO" considers the information of the whole entity span comprehensively, "Start and End" (SE) exploits the entity boundary explicitly, and "Tie or Break" (TB) models the connection information inside an entity. By following our multi-view decoding strategy, more shared features between the source and target domain can be learned from different views for more effective transfer.

Type Prediction. *Type prediction* is more difficult to transfer than *Entity detection* since the source and target domain have different entity categories (i.e., different label spaces in classification) as discussed previously. Prompt-tuning [2, 33] aims to bridge the gap between pre-training tasks and various downstream tasks in the **natural language processing (NLP)** community. Inspired by this, we adapt the prompt-tuning strategy to the cross-domain transfer for bridging the gap across domains in this sub-task. Concretely, prompt-tuning maintains the word prediction paradigm of the **pre-trained language model (PLM)** to predict a class-related pivot word (or label word) in the PLM vocabulary. Under this paradigm, label words for both the source and target domain are subsets of the PLM vocabulary which allows our model to exploit label correlations across domains, thus reducing the domain gap and type discrepancy. Additionally, the task prediction form is unified for contributing to the transferability between the source and target domain.

In a nutshell, the major contributions of this work are summarized as follows:

— Different from previous monolithic transfer under the sequence labeling framework, we propose the first divide-and-transfer paradigm to disentangle the entity span and type information for more effective transfer in each sub-task, which points out a new research idea for cross-domain NER. The divide-and-transfer paradigm originates from our conference version [65]. This article further clarifies and follows this paradigm to propose a new way of implementing it, developing more effective cross-domain strategies and extending the divide-and-transfer paradigm into diverse scenarios.

— We devise two specific cross-domain NER frameworks—DTrans-SMix and DTrans-MPrompt—by following the proposed divide-and-transfer paradigm. For example, we develop the multi-view decoding strategy for effective knowledge transfer in the *entity detection* sub-task, and first adapt the prompt-tuning to cross-domain transfer for handling the transfer obstacle in the *type prediction* sub-task.

— We evaluate two frameworks on 10 different domain pairs and verify their effectiveness (about average 5.27% and 8.44% absolute F1 score increase), which shows the great potential of our divide-and-transfer paradigm. Further experiments confirm the significant superiority of the proposed paradigm in the extremely low-resource scenario. Using only 10% data in the DTrans-MPrompt framework can achieve the comparable performance as using the full target domain data in previous SOTAs.

This article is a significant extension of our conference version published in SIGIR '22 [65]. The major extensions are as follows:

— We formally summarize the divide-and-transfer paradigm from our previously published work DTrans-SMix [65]. The general paradigm further enriches the cross-domain NER community. Notably, this article extends the divide-and-transfer paradigm to *diverse scenarios*, including low-resource, few-shot, and zero-shot cross-domain scenarios with different label spaces across domains, whereas our previous method DTrans-SMix is limited by the zero-shot scenario.

— Following this paradigm, we devise a novel framework DTrans-MPrompt where cross-domain transfer strategies for entity span and type information are more precise and effective than our original work, taking the performance to a new height—for example, the multi-view decoding strategy instead of simple parameter sharing for entity span transfer, and the unified task prediction form based on prompt-tuning instead of intermediate domain augmentation for entity type transfer.

— We also conduct more in-depth experiments to demonstrate that the proposed divide-and-transfer paradigm shows great generalization ability and can be well extended with tailor-designed transfer strategies in two sub-tasks for cross-domain NER, such as more cross-domain NER baselines and benchmark datasets. Specifically, we further explore the effectiveness of our DTrans-MPrompt proposed in this article under the few-shot and zero-shot cross-domain scenarios, and extend the divide-and-transfer paradigm to cross-domain slot filling, which is also typically a sequence labeling task.

The rest of the article is organized as follows. Section 2 briefly reviews existing research related to our work. We introduce the divide-and-transfer paradigm and newly proposed DTrans-MPrompt in Section 3. Section 4 shows the experimental settings and results to illustrate the effectiveness of the divide-and-transfer paradigm together with further experimental analyses. We conclude our research and present our future work in Section 5.

2 RELATED WORK

2.1 Cross-Domain NER

The end-to-end sequence labeling framework [8, 25] is a popular paradigm that assigns each token a compositional tag (e.g., B-ORG) in NER. Most existing cross-domain NER methods are based on this framework for transfer learning, which can be categorized into *domain mapping* [22, 31, 39, 53, 59, 63] and *parameter transfer* [34, 51, 61, 68]. Domain mapping methods aim to map the features from one domain to another. Jia et al. [19] used the cross-domain **language model (LM)** as a bridge to map from the source to the target domain by designing a novel parameter generation network. Chen et al. [3] studied data augmentation for the cross-domain NER task by projecting data from high-resource domains into low-resource domains. Ma et al. [37] modeled the subword distribution between the source and target domain by solving an optimal transport problem. Zheng et al. [66] built label graphs in both source and target label spaces and performed the graph matching operation for domain mapping. Different from domain mapping approaches, parameter transfer methods tend to share model parameters between the source and target domain. Jia and Zhang [21] proposed a multi-cell compositional LSTM structure on the top of the BERT encoder for multi-task learning, making the cross-domain transfer perform at the entity type level. Liu et al. [35] developed the domain-adaptive pre-training (DAPT) and encoder-shared method for the cross-domain NER task.

Few-shot NER involves learning unseen classes from very few labeled examples, where some few-shot methods also evaluate their cross-domain ability [7, 38, 60]. For instance, Yang and Katiyar [60] proposed NNShot and StructShot with a compositional tagging scheme (e.g., B-LOC) based on the nearest neighbor classifier. These approaches aim to generalize models from very few examples (K-shot).

Most methods are based on a sequence labeling paradigm (e.g., B-LOC), which is a compositional task and requires one model to decide entity span and category simultaneously. Unlike these works, we break down the original NER task into two sub-tasks (*entity detection* and *type prediction*) and verify its superiority in cross-domain transfer.

2.2 Task Decomposition

For some existing NLP and computer vision tasks, decomposing the compositional task into single sub-tasks is very common, which aims to solve issues existing in compositional tasks [10, 48]. For example, in nested NER, Tan et al. [49] proposed a boundary-enhanced neural span classification model that first generated the candidate nested spans and then classified them. They focused on recognizing nested entities by decomposing the NER task. To recognize long entities effectively, Shen et al. [47] divided the NER task and designed a two-stage entity identifier, which first located the long entities by boundary regression, then labeled the span with the corresponding entity categories. In joint extraction of entities and relations, Yu et al. [62] decomposed the joint extraction task into head-entity extraction, and tail-entity and relation extraction to reduce the redundant entity pairs and consider the important inner structure in the process of extracting entities and relations. Wang et al. [52] decomposed the task for learning from the natural language descriptions of entity classes sufficiently. In object detection, Xie et al. [55] divided the task into two stages and proposed an oriented region proposal network for reducing the expensive computation during generating proposals.

We decompose the NER task into two sub-tasks (entity detection and type prediction) for disentangling the hybrid transfer under the monolithic sequence labeling framework. By transferring in each sub-task with reasonable cross-domain strategies, more information can be transferred from the source to the target. To the best of our knowledge, there is currently no specific research for exploring the efficacy of dividing the task in cross-domain NER. Our work mainly inspires a new perspective on cross-domain NER, which is completely different from the existing work introduced earlier.

2.3 Prompt Learning

A series of PLMs make NLP tasks achieve promising performance, such as BERT [8], BART [27], T5 [43], and GPT [42]. PLMs only need to be fine-tuned and show their effectiveness on downstream tasks, like text classification [12], NER [20], and question answering [1]. However, the optimization objective gap between pre-training and fine-tuning limits the utilization of PLM model capabilities on downstream tasks [9, 17, 33]. In this connection, prompt learning is proposed to unleash the knowledge contained in PLMs [2, 5, 11].

Prompt learning formalizes the downstream task as a *cloze-style* objective with a *prompt context* and *verbalizer* similar to those pre-training objectives [9], narrowing the gap between pre-training and fine-tuning. Stemming from GPT-3 [2], which achieves impressive performance on downstream tasks by prompt-tuning, massive prompt learning based methods are arising by focusing on designing hand-crafted prompts. Gao et al. [11] treated the downstream task as a masked language modeling problem given task-specific prompt, where the model directly generated a label word in PLM vocabulary for task prediction. Schick and Schütze [45] utilized natural language patterns to reformulate input sentences into cloze-style phrases to alleviate labor-intensive prompt engineering. Li and Liang [29] proposed prefix-tuning, which kept PLM parameters frozen and instead optimizes a sequence of continuous task-specific vectors as prompts rather than discrete language words. Recently, extensive studies show that **large language models (LLMs)** can be prompted to perform various NLP tasks, given text instruction and some examples of the task as input [41]. The LLMs represented by ChatGPT[2] have attracted widespread attention in both academia and industry [14, 28, 67], profoundly influencing the transformation of research paradigms.

We adapt the prompt technique to cross-domain scenarios which effectively bridges the domain gap by the unified task prediction form, especially for distinct label spaces across domains. LLMs achieve impressive performance on a series of NLP tasks, and their ability to information extraction (e.g., NER) under cross-domain transfer scenarios needs further evaluation and exploration in the future.

3 METHODOLOGY

3.1 Problem Definition

Given a sentence X=<$w_1, w_2, \ldots, w_n$>, w_i is a word (token) and n is the length of the sentence. An entity is a span of X with a category: $\mathbf{e} = \{(w_{start}, w_{start+1}, \ldots, w_{end}), l^e\}$, where $l^e \in C$ is an entity type (category) (e.g., person, location). C is a set of entity types in a specific domain. NER focuses on finding entity $\mathbf{e}$ in the sentence. For cross-domain NER that transfers information from the source domain to the target, there are $N_{\mathcal{S}}$ labeled sentences in the source domain $\mathcal{S}$, and its entity type set is denoted as $C_{\mathcal{S}}$. The target domain $\mathcal{T}$ has $N_{\mathcal{T}}$ labeled sentences. The entity type set in $\mathcal{T}$ is $C_{\mathcal{T}}$.

In this article, we focus on transferring from a high-resource domain ($\mathcal{S}$) to a low-resource domain ($\mathcal{T}$), and there are different entity type spaces between the source and target domain—that is, $N_{\mathcal{T}} \ll N_{\mathcal{S}}$ and $C_{\mathcal{S}} \neq C_{\mathcal{T}}$. The cross-domain experiment in this article is more challenging and meets the real-world cross-domain scenario. Specifically, we also conduct cross-domain experiments under zero-shot scenarios where $N_{\mathcal{T}}$ is zero.

[2]Launched by OpenAI in November 2022 (https://chat.openai.com/chat)

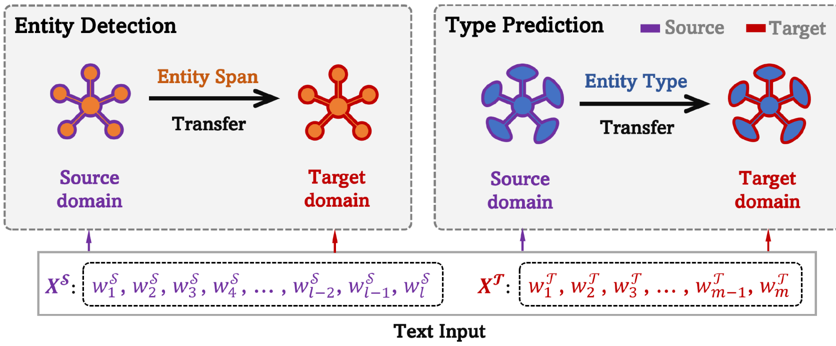

Fig. 3. Overview of the divide-and-transfer paradigm. The NER task is divided into sub-tasks (entity detection and type prediction) using separate functional modules with corresponding cross-domain strategies.

3.2 Divide-and-Transfer Paradigm

Owing to easy implementation and promising performance, the sequence labeling framework has always been a popular paradigm where each token is assigned a compositional label (e.g., B-PER). But this paradigm needs one model to decide the entity span and type concurrently, and transferring these two kinds of information under the monolithic framework is challenging because of their discrepant transferability. Thus, we divide the NER task into two sub-tasks (*Entity Detection*, *Type Prediction*) to disentangle the transferred information for more effective transfer in each sub-task (Figure 3).

After being divided into *Entity Detection* and *Type Prediction* sub-tasks, the most critical step is to devise the corresponding cross-domain transfer strategies in each sub-task for contributing to the transfer from the source to the target domain. As shown in Figure 3, what we need to do is to propose cross-domain strategies and then transfer entity span information from the source to the target in the entity detection sub-task, which assists in detecting entity spans on the target domain. Similarly, the type prediction sub-task also requires tailor-designed modules to improve the prediction of the target entity type. Additionally, the cross-domain barriers are different for the transfer of entity span and type information in two individual sub-tasks. Compared with the hybrid transfer of two kinds of entity information under a monolithic paradigm (e.g., sequence labeling based), there are more possibilities for transfer strategy combinations and greater improvement in divide-and-transfer paradigm based cross-domain NER due to specific sub-task transfer strategies. For example, the divide-and-transfer paradigm based cross-domain NER frameworks *DTrans-SMix* and *DTrans-MPrompt* separately devise the *parameter sharing* and *multi-view decoding* strategies for entity detection, *mixup based intermediate domain augmentation*, and *prompt-tuning* strategies for the type prediction sub-task.

Last but not least, the divide-and-transfer paradigm needs to coordinate the relationship between two sub-tasks to obtain the final NER result (i.e., entity span and type). For instance, DTrans-SMix and DTrans-MPrompt respectively propose the modular interaction mechanism and re-detecting strategy for the explicit interaction of two sub-tasks.

Formally, given a sentence $X^{\mathcal{S}} = < w_1^{\mathcal{S}}, w_2^{\mathcal{S}}, w_3^{\mathcal{S}}, w_4^{\mathcal{S}}, \ldots, w_{l-2}^{\mathcal{S}}, w_{l-1}^{\mathcal{S}}, w_l^{\mathcal{S}} >$ from the source domain and $X^{\mathcal{T}} = < w_1^{\mathcal{T}}, w_2^{\mathcal{T}}, w_3^{\mathcal{T}}, \ldots, w_{m-1}^{\mathcal{T}}, w_m^{\mathcal{T}} >$ from the target domain, the divide-and-transfer paradigm first decomposes the NER task into Entity Detection (**ED**) and Type Prediction (TP) sub-tasks, then devises corresponding cross-domain transfer strategies for each sub-task to transfer source-domain information. Concretely, the divide-and-transfer paradigm needs to learn an entity detection model $f_{\Theta_{ED}}(X^{\mathcal{S}}, X^{\mathcal{T}}; \Phi_{ED})$ and a type prediction model $f_{\Theta_{TP}}(X^{\mathcal{S}}, X^{\mathcal{T}}; \Phi_{TP})$

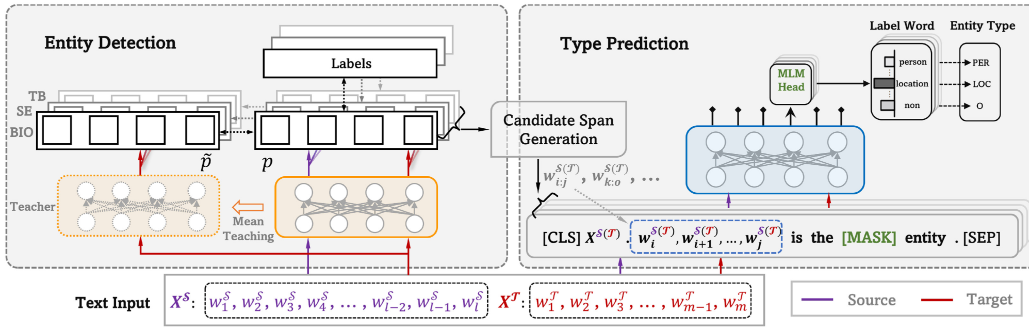

Fig. 4. Overview of DTrans-MPrompt, where the transfer is performed in two sub-tasks (entity detection and type prediction) separately.

based on cross-domain transfer strategies Φ_{ED}, Φ_{TP} with source and target domain data $X^{\mathcal{S}}$, $X^{\mathcal{T}}$ in the corresponding sub-tasks. Specifically, the interaction strategy Υ is developed to associate two sub-tasks for final NER results.

3.3 DTrans-MPrompt Cross-Domain Framework

DTrans-SMix in the conference paper [65] devises *simple parameter sharing* for entity span transfer in the entity detection sub-task and *intermediate domain augmentation* for entity type transfer in the type prediction sub-task. Instead, DTrans-MPrompt is composed of the *multi-view decoding strategy* for the entity detection sub-task and the *prompt-tuning based label space unification* for type prediction. Overall, they both follow the divide-and-transfer paradigm, and DTrans-MPrompt proposed in this article possesses more precise and effective transfer strategies in sub-tasks. Additionally, DTrans-MPrompt can work under zero-shot cross-domain scenarios with different label spaces across domains, which further extends the divide-and-transfer paradigm to diverse scenarios. Full details of *DTrans-SMix* can be found in our previously published version [65]. Next, we detail the *DTrans-MPrompt* framework that abides by the divide-and-transfer paradigm.

3.3.1 Entity Detection Sub-Task. Given a sentence $X = <w_1, w_2, \ldots, w_n>$, this sub-task aims to locate entity spans in the text. We use BERT as the backbone for hidden representations $\mathbf{H} = <\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_n> \in \mathbb{R}^{n \times d}$:

$$\mathbf{H} = \mathrm{BERT}(X), \tag{1}$$

where n is the length of a sentence and d is the dimensions of last hidden layer in BERT.

Multi-View Decoding Strategy. To find more similarities between the source and target domain, we propose a multi-view decoding strategy where three kinds of tagging schemes (i.e., "BIO," "Start and End," "Tie or Break") are utilized from different perspectives. Three kinds of tagging schemes capture features about what an entity is from different perspectives. For example, "Start and End" focuses on the entity boundary, and "Tie or Break" lays emphasis on the entity inside, which contributes to more features transfer from the source domain to the target. Then more domain-invariant features can be captured in different granularities.

"BIO" Tagging Scheme. This scheme detects entity spans with global contextual information at the sentence level. We tag the beginning token of an entity as ``*B*" and other tokens of the same entity as ``*I*." Non-entity is tagged as ``*O*." As shown in Figure 4, we can get the hidden representation $\mathbf{h}_i$ for each token w_i (Equation (1)). Then $\mathbf{h}_i$ is passed into a *Fully Connected Network (FC)* to get the label probability distribution for w_i.

"Start and End" Tagging Scheme. This scheme explicitly models the entity boundary information that effectively perceives the start and end positions of entities. Similarly, hidden representation $\mathbf{h}_i$ is fed to two binary classifiers (FC) to predict the probability of each token w_i being a start or end position. The binary tag (*0/1*) indicates whether w_i corresponds to a *start* or *end* position, as shown in Figure 2.

"Tie or Break" Tagging Scheme. This scheme encodes the connection between adjacent tokens within an entity. As shown in Figure 2, T (Tie) indicates two adjacent tokens belong to the same entity, and B (Break) is for otherwise. Concretely, given the hidden representations $\mathbf{h}_{i-1}$, $\mathbf{h}_i$ of two adjacent tokens, they are added to get the token interaction representation. Then a classifier is constructed to predict the probability of each token pair being tied as follows:

$$p\left(t_k|w_{i-1};w_i\right)=\frac{\exp\left\{\hat{\mathbf{w}}_k^\top(\mathbf{h}_{i-1}+\mathbf{h}_i)+\hat{b}_k\right\}}{\sum_{t_j\in\mathcal{R}}\exp\left\{\hat{\mathbf{w}}_j^\top(\mathbf{h}_{i-1}+\mathbf{h}_i)+\hat{b}_j\right\}}, \tag{2}$$

where $[\hat{\mathbf{w}}_k;\hat{b}_k]$ are parameters specific to the k-th tagging class t_k. $t_k\in\mathcal{R}$ and $\mathcal{R}=\{$"Tie", "Break"$\}$.

The optimization objectives of the preceding three tagging schemes all adopt the cross-entropy loss function, notated as $\mathcal{L}_{\text{BIO}}$, $\mathcal{L}_{\text{SE}}$, and $\mathcal{L}_{\text{TB}}$ (corresponding to "*BIO*," "*S*tart and *E*nd," and "*T*ie or *B*reak").

Mean Teaching. The annotation sparsity on the target domain cannot be underestimated, which tends to cause the overfitting problem. To alleviate this issue, we apply exponential moving average (EMA) [23, 50, 64] to gradually accumulate the parameters θ of the original entity detection model as the teacher model's parameters $\widetilde{\theta}$. The formula is as follows:

$$\widetilde{\theta}_t\leftarrow\alpha\widetilde{\theta}_{t-1}+(1-\alpha)\theta_t, \tag{3}$$

where α denotes the smoothing coefficient and t means the t-th iteration. Before the first iteration, $\widetilde{\theta}_0=\theta_0$, which are initialized with the same parameters (e.g., BERT).

The teacher model can be viewed as the ensemble of original models in different training iterations. As α is generally assigned a value close to 1 (e.g., 0.995), the teacher model is more stable, which prevents the model from overfitting limited target data. Thus, we distill the logit outputs $\widetilde{\mathbf{p}}_i$ of the teacher into original model for robust training ($\mathbf{p}_i$ is the logit outputs of original model):

$$\mathcal{L}_{dis}=\mathbb{E}_i\left[\|\mathbf{p}_i-\widetilde{\mathbf{p}}_i\|^2\right]. \tag{4}$$

3.3.2 Type Prediction Sub-Task. Given a candidate entity span, the type prediction sub-task focuses on classifying it into pre-defined entity categories.

Prompt-Tuning-Based Label Space Unification. As the source and target domain have distinct entity categories, the classification head remains different across domains in the standard fine-tuning process, which causes an obvious gap, whereas prompt-tuning can reformulate the fine-tuning classification task as a PLM task (**masked language model (MLM)** [8]), which predicts the label word in the PLM vocabulary $\mathcal{V}$. Therefore, the label spaces of the source and target domain are both the subsets of PLM vocabulary, which allows our model to exploit label correlations across domains and narrow down the domain gap. Under this paradigm, what is learned is the ability to select label words from the PLM's vocabulary based on context, and no new model parameters are introduced for the target domain, promoting the transfer of this ability from the source domain to the target domain. Additionally, no new parameters are introduced in prompt-tuning, so the model

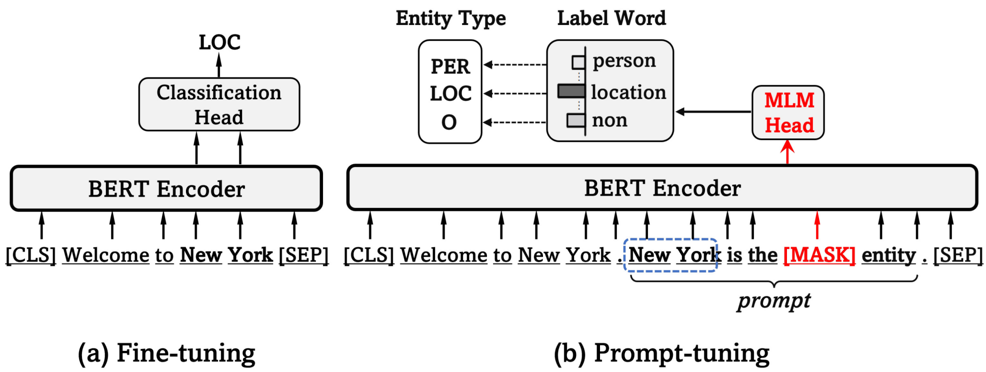

Fig. 5. Fine-tuning vs. Prompt-tuning.

can adapt to the cross-domain task in the annotation sparsity scenario. Then we do not develop extra strategies for low-resource target domains in this sub-task.

As shown in Figure 4, the input of prompt-tuning is organized as $\mathcal{X}$ = "X. $(w_{start}, \ldots, w_{end})$ is the [MASK] entity." and X is the sentence where the entity span $e_s = (w_{start}, \ldots, w_{end})$ is located. For each entity category $l^e \in C$ (e.g., LOC), we define a label word $v^e \in \mathcal{V}$ (e.g., location). Then $\mathcal{X}$ is input into PLM to predict the missing label word at the masked position. Thus, the classification problem is converted into a masked language modeling problem:

$$p(l^e \in C|X; e_s) = p([\text{MASK}] = v^e \in \mathcal{V}|\mathcal{X}). \tag{5}$$

The training objective is a cross-entropy loss, $\mathcal{L}_{\text{PT}}$.

Figure 5 shows the fine-tuning and prompt-tuning paradigms in the *Type Prediction* sub-task. *Fine-tuning* with an extra classification head has been the typical solution for adapting PLM (e.g., BERT) to downstream tasks. As shown in Figure 5(a), given the entity span "New York" and its context "[CLS] Welcome to New York [SEP]," fine-tuning adds a new classification head on the top of BERT encoder for classifying its entity category LOC. *Prompt-tuning* focuses on probing knowledge of PLM with the prompt for downstream tasks. As shown in Figure 5(b), we need to construct the input "[CLS] Welcome to New York . New York is the[MASK] entity . [SEP]" for classifying the category of "New York." Concretely, PLM with its MLM head can compute a probability distribution over the vocabulary at the masked position. The word "location" with the highest probability can be mapped into its corresponding entity category "LOC" (each category corresponds to a word in the PLM vocabulary). Therefore, despite of the source and target domain, prompt-tuning is to predict a label word in the PLM vocabulary, which effectively lessens the domain gap.

Re-Detecting Strategy. In general, the type prediction model (prompt-tuning) can be trained with the ground-truth entity spans according to teacher forcing [24]. However, the entity spans are generated by the entity detection sub-task in the inference phase, which will cause inconsistency between the training and inference phase. Therefore, we add a new label "O" in the type prediction sub-task to filter the false-positive entity spans. In the training phase, we also use the results of the entity detection sub-task, and the entity span is labeled as "O" when it is a false-positive span. But the false-positive entity span may be highly overlapping with the ground-truth one, which will not make it be filtered out during inference. Thus, we add low-threshold filtering as

$$E = \{e_s | \max p([\text{MASK}] = v^e \in \mathcal{V}|\mathcal{X}) \geq \delta\}. \tag{6}$$

That is to say, low-confidence candidate spans are ignored, $\delta \in (0, 1)$.

3.3.3 *Optimization and Inference.* ***Training Phase.*** In each training step of entity detection (ED) and type prediction (TP) sub-tasks, we choose training samples from the source ($\mathcal{S}$) and target ($\mathcal{T}$) domains, respectively. Our training procedure is two-stage, which first trains in the ED sub-task and then the TP sub-task is trained with the outputs of the ED sub-task as input. Their training objectives are as follows:

$$\begin{aligned} \mathcal{L}_{\text{ED}} &= \mathcal{L}_{\text{BIO}}^{\{\mathcal{S},\mathcal{T}\}} + \mathcal{L}_{\text{SE}}^{\{\mathcal{S},\mathcal{T}\}} + \mathcal{L}_{\text{TB}}^{\{\mathcal{S},\mathcal{T}\}} + \mathcal{L}_{dis}^{\mathcal{T}} \\ \mathcal{L}_{\text{TP}} &= \mathcal{L}_{\text{PT}}^{\{\mathcal{S},\mathcal{T}\}}. \end{aligned} \tag{7}$$

Inference Phase. During inference, the entity detection sub-task generates the candidate spans $\mathcal{E}$. Concretely, the entity spans in three tagging schemes ("BIO," "Start and End," and "Tie or Break") are denoted as $\mathcal{E}_{\text{BIO}}$, $\mathcal{E}_{\text{SE}}$, and $\mathcal{E}_{\text{TB}}$, respectively. Then the *generated candidate entity spans* $\mathcal{E}$ can be formed as follows:

❶ $\mathcal{E}_{\text{BIO}}$ or $\mathcal{E}_{\text{SE}}$ or $\mathcal{E}_{\text{TB}}$ (choosing one of them)
❷ $\mathcal{E}_{\text{BIO}} \cup \mathcal{E}_{\text{SE}} \cup \mathcal{E}_{\text{TB}}$ (ensembling of them).

Then the type prediction sub-task gives the entity category l^e of candidate spans. The final results of the NER task are as follows:

$$E_f = \{(e_s, l^e) | l^e \neq \text{``O''}, e_s \in \mathcal{E} \cap E\}, \tag{8}$$

where e_s is the entity span and $l^e = \arg\max p([\text{MASK}] = v^e \in \mathcal{V} | \mathcal{X})$ is its corresponding entity category.

4 EXPERIMENTS

In this section, we conduct extensive experiments and verify the following research questions:

RQ1: How is the efficacy of the divide-and-transfer paradigm?

RQ2: Does the divide-and-transfer paradigm contribute to more gains (effective transfer) from the source domain?

RQ3: How does DTrans-MPrompt perform under zero-shot cross-domain scenarios?

RQ4: How about the effect of the divide-and-transfer paradigm when applied to other sequence labeling tasks similar to NER?

4.1 Experimental Settings

4.1.1 *Datasets.* ***Low-Resource Scenario.*** We evaluate our two frameworks on 10 different domain pairs, consisting of two source domain datasets: CoNLL2003 [44] (*Newswire* domain) and Twitter [36] (*Social Media* domain), and five low-resource target domain datasets (only 100 or 200 labeled sentences) released by Liu et al. [35]: *Politics*, *Natural Science*, *Music*, *Literature*, and *Artificial Intelligence*. The detailed statistics of datasets are reported in Table 1. The text genres and entity categories are completely different between the source and target domains, and the two source domain (Newswire and Social Media) datasets are the most common NER datasets. So the cross-domain setting of this work is more applicable in the real world.

Few-Shot Scenario. Additionally, we conduct extensive experiments under few-shot settings. Similarly, CoNLL2003 [44] (*Newswire* domain) is used as the high-resource source domain. Following the settings in other works [4, 7, 18, 69], we use MIT Movie [32] on the *Review* domain and ATIS [15] on the *Dialogue* domain as the cross-domain few-shot datasets, serving as target domains. Details of these datasets are shown in Table 2. We focus on the few-shot scenario where only few-shot labeled data are available for training on the target domain. As in the work of Cui et al. [7], a fixed number of instances per entity type (e.g., K = 10, 20, 50) are randomly sampled. If

Table 1. Statistics of Cross-Domain NER Datasets

Domain	Dataset	#Train	#Dev	#Test	#Category
Source	CoNLL2003 (Newswire)	14041	–	–	4
	Twitter (Social Media)	4290	–	–	4
Target	Politics	200	541	651	9
	Natural Science	200	450	543	17
	Music	100	380	465	13
	Literature	100	400	416	12
	Artificial Intelligence (AI)	100	350	431	14

Table 2. Statistics of Few-Shot NER Datasets

Dataset	#Train	#Test	#Category	K-shot
MIT Movie (Review)	7.8k	2.0k	12	K = 10, 20, 50 100, 200, 500
ATIS (Dialogue)	5.0k	893	79	K = 10, 20, 50

an entity has a smaller number of instances than the fixed number to sample, they use all of them for training. We use the sampled datasets (K-shot) for our few-shot experiments.

Zero-Shot Scenario. Following Nguyen et al. [39], we utilize the three domains of *Science*, *Literature*, and *Music* released by Liu et al. [35] because they have the largest number of entities. Their original statistics are shown in Table 1. As in the work of Nguyen et al. [39], we use one domain as the source and the rest of the domains serve as targets, forming six cross-domain pairs. It is worth noting that we only use the source domain labeled data for training and target domain data are not available (i.e., training on source labeled training data and directly predicting on target test data). Additionally, the entity type spaces of the source and target domain are different.

Slot Filling Task. Same as NER, slot filling is a classic sequence labeling task, which our divide-and-transfer paradigm can adapt to. We evaluate our paradigm on SNIPS [6], a popular slot filling dataset that contains 39 slot types, 7 domains, and about 2,000 training samples per domain. Following previous cross-domain slot filling studies [34, 58], we use one domain as the source and the remaining six domains as targets each time, for a total of seven cross-domain pairs.

4.1.2 Baselines. Under a ***low-resource scenario***, we compare our DTrans-SMix and DTrans-MPrompt frameworks with the following SOTA methods. *BiLSTM-CRF* [25] combines the source domain and the upsampled target domain data to train the model jointly. *Coach* [34] proposes the coarse-to-fine method with the label description for the data scarcity problem. *LM-NER* [19] bridges the source and target domain using parameter generation networks where language modeling tasks and NER tasks in both source and target domains are integrated. *NNShot* and *StructShot* [60] are two metric-based few-shot NER methods. They exploit a nearest neighbor classifier for few-shot prediction. Compared with NNShot, StructShot develops a Viterbi algorithm during decoding. We extend these two methods to our cross-domain settings by jointly training with the source and target domain data. *MultiCell-LM* [21] develops a multi-cell compositional LSTM structure on top of BERT based on the multi-task transfer learning for learning domain-invariant in the entity level, which models each entity type using a separate cell state. *Template* [7] is a template-based NER method that treats NER as an LM ranking problem in a sequence-to-sequence framework. In the work of Liu et al. [35], *BERT-JF* jointly fine-tunes BERT on both the source and upsampled target domain data. *BERT-PF* first pre-trains BERT on the source domain data, then fine-tunes it to the target domain. *Style-NER* [3] studies the data augmentation in cross-domain NER, which adopts the adversarial transfer idea for projecting the source domain data into the target domain to generate the target data in the labeled data sparsity scenario. *LightNER* [4] is a generative framework [57] with prompt-guided attention that incorporates continuous prompts into the self-attention layer for low-resource NER. We pre-train it on the source domain data and then fine-tune it to the target domain, following the original paper. *EntLM* [38] proposes a template-free approach to prompt NER under few-shot settings. We jointly train it with the source and target domain data for adapting it to the cross-domain settings. *LST-NER* [66] formulates cross-domain NER as a graph matching problem by constructing label graphs in both source and target label spaces to cope with the distinct label sets across domains. Overall, most competitive baselines all model the entangled entity span and type information in a monolithic process fashion.

Under a *few-shot scenario*, we compare our DTrans-SMix and DTrans-MPrompt with the following few-shot NER methods. *Example* [69] is a few-shot NER method inspired by extractive question answering, which first trains model on source domain, then models the correlation between support examples and a query on target domain. *MP-NSP* [18] is a prototype-based method that creates prototypes as the representations for different labels and then predicts via the nearest neighbor criterion. Some baselines from low-resource scenarios like *NNShot* [60], *StructShot* [60], *Template* [7], *LightNER* [4], and *EntLM* [38] can be directly applied to the few-shot scenario.

Under a *zero-shot scenario*, because our previously published DTrans-SMix [65] cannot be adapted to this scenario, we only compare DTrans-MPrompt with the following zero-shot cross-domain NER methods. *LUKE* [56] proposes an entity-aware self-attention mechanism and considers the types of tokens when computing attention scores. To extend LUKE to zero-shot learning, Nguyen et al. [39] propose to learn entity features for each entity label and then compute the dot product with the token hidden representations. *DOZEN* [39] proposes cross-domain zero-shot NER that learns the relations between entities from an existing ontology of knowledge graph across different domains. *DOZEN** [39] is the ablated version of DOZEN without entity analogy modeling of multiple domains. Our DTrans-MPrompt does not use the external knowledge graph.

For a *slot filling task*, we evaluate our divide-and-transfer paradigm by comparing with the following cross-domain slot filling SOTA methods. **Concept Tagger (CT)** [13] proposes to exploit slot descriptions for generalizing to unseen slot types. **Robust Zero-shot Tagger (RZT)** [46] proposes to use both slot descriptions and a few examples of slot values for learning transferable semantic representations across domains. **Coarse-to-fine Approach (Coach)** [34] is a coarse-to-fine slot-filling model that also uses slot descriptions for unseen slot types. **Abundant Information Slot Filling Generator (AISFG)** [58] incorporates domain descriptions, slot descriptions, and examples with context by a generative model with a query template to deal with slot type and example ambiguity issues.

4.1.3 Implementation Details. Our frameworks are based on BERT-base [8] for a fair comparison with previous SOTAs. Other BART-related baselines take BART-base [27] as the backbone. For main results, we tune hyperparameters with Grid-Search according to the results on *dev* sets. The learning rate is 1e-5, maximum training epochs is 30, and the seed of random numbers is set to 0. In DTrans-SMix, for each mini-batch, we sample 16 sentences from the source and target domain datasets, respectively. The fixed mixup ratios (α, β) are set to (0.3, 0.7) by tuning from {(0.1, 0.9),. . . , (0.9, 0.1)}. ξ in entity detection and type prediction sub-task is set to 0.5 and 0, respectively. τ is tuned from {0.1, 1.0, 10} in two sub-tasks and finally is set to 0.1 on all datasets, except *Politics* is 10 in the entity detection sub-task. L is set to 3 by tuning from 0 to 12. We tune μ from {0.2, 0.5, 1.0, 2.0} and set 0.5 on the *Politics* and *AI* datasets, and others are 1.0. λ is set to 0.1. In DTrans-MPrompt, for each mini-batch in two sub-tasks, we sample 32 sentences from the source and target domains, respectively. The EMA α is set to 0.995, and the filtering threshold δ is tuned from {0.5, 0.55, 0.6,…, 0.9}. When generating candidate spans $\mathcal{E}$, we use ❶ (choosing $\mathcal{E}_{\text{BIO}}$). Following prior work, we use the F1 score as the evaluation metric based on exact span matching. We implement our code with PyTorch based on huggingface Transformers [54]. The baseline (except marked with † and LST-NER) results of the first five domain pairs are all from the work of Liu et al. [35]. We report the results of LST-NER [66] from its original paper. As LST-NER [66] has not released the official code, we cannot produce results of the last five domain pairs for it. For other experimental results, we follow the officially released implementation. For efficiency experiments, the specifications of the system used for the time measurements are as follows: (1) the CPU processor is *Intel Xeon Silver 4110 CPU @ 2.10 GHz*, (2) the GPU is *Tesla T4 (16 G)*, (3) the operating system is *CentOS 7*, and (4) the versions of Python and PyTorch respectively are *Python 3.7.4* and *PyTorch 1.8.1*.

4.2 Main Results under a Low-Resource Scenario (RQ1)

Table 3 and Table 4 show the main results of our frameworks compared to competitive baselines. On 10 domain pairs, our frameworks consistently outperform the previous SOTAs with large margins (2.10% ~ 8.61% absolute F1 increase in DTrans-SMix, 6.23% ~ 9.98% increase in DTrans-MPrompt). This demonstrates that the divide-and-transfer paradigm is more effective, which provides a new perspective on cross-domain NER. Most previous cross-domain SOTAs (e.g., MultiCell-LM [21], BERT-JF [35], Style-NER [3], and LST-NER [66]) take the end-to-end sequence labeling framework as the backbone with devised transfer strategies. Our remarkable improvement reflects the limitation that sequence labeling is not ideal in the cross-domain transfer of NER and impairs the efficacy of transfer strategies due to its coupled information transfer. Meanwhile, the main results indicate that the divide-and-transfer paradigm seems to be more suitable as a benchmark transfer framework in NER. Figure 6 shows the learning curves during training, which not only confirms the consistent improvements of divide-and-transfer but also reflects its robust training process and powerful generalization. We also report the training and prediction times for some of the baseline methods and ours in Table 3 and Table 4. Although our methods do not achieve the optimal efficiency for model training, its training time is acceptable (especially the newly proposed DTrans-MPrompt in this article) considering its significant performance improvement comprehensively. It is worth noting that the prediction time of our methods is comparable in comparison with other baselines. That is to say, our method exhibits no efficiency shortcomings during the inference application phase. Additionally, we observe that LightNER [4] has higher training efficiency but lower prediction efficiency because it adopts the parameter-efficient fine-tuning [16, 26, 30] strategy during training and generates the word by word in an auto-regressive manner during prediction. DTrans-SMix requires longer training time due to the construction process of the intermediate augmented domain. Overall, DTrans-MPrompt is more efficient than DTrans-SMix [65].

In Table 3 and Table 4, we observe that no prior baselines can always occupy an absolute advantage on 10 domain pairs, whereas our proposed divide-and-transfer paradigm (both DTrans-SMix and DTrans-MPrompt) can keep the superiority consistently. Our comparison baselines also contain some advanced few-shot or low-resource NER methods (e.g., NNShot [60], StructShot [60], and LightNER [4]). We can see that our divide-and-transfer-based frameworks both significantly outperform them by a large margin. The reason for this is that those few-shot or low-resource baselines learn entity span and type information in a monolithic framework by a compositional

Table 3. F1 Scores on Five Different Domain Pairs That Transfer from the Source Domain Newswire to Five Target Domains, Respectively

	Source Domain	CoNLL2003 (Newswire) →				
	Target Domain	**Politics**	**Natural Science**	**Music**	**Literature**	**AI**
Methods	BiLSTM-CRF [25]	56.60	49.97	44.79	43.03	43.56
	Coach [34]	61.50	52.09	51.66	48.35	45.15
	LM-NER [19]	68.44	64.31	63.56	59.59	53.70
	NNShot [60]†	65.84	64.11	65.72	61.24	56.23
	StructShot [60]†	66.69	65.98	68.62	63.34	57.38
	Template [7]†	65.84	61.95	65.57	63.78	55.01
	BERT-JF [35]	68.85	65.03	67.59	62.57	58.57
	BERT-PF [35]	68.71	64.94	68.30	63.63	58.88
	MultiCell-LM [21]	<u>70.56</u>	66.42	70.52	66.96	58.28
	Style-NER [3]†	68.78	63.95	65.43	60.94	58.73
	LightNER [4]†	69.36	63.47	70.20	64.77	53.96
	Training Time	31.84 *min.*	31.39 *min.*	31.12 *min.*	32.41 *min.*	32.26 *min.*
	Prediction Time	10.49 *s*	9.12 *s*	7.65 *s*	5.86 *s*	6.32 *s*
	EntLM [38]†	69.19	63.93	68.72	63.55	57.48
	Training Time	88.30 *min.*	94.29 *min.*	78.91 *min.*	86.99 *min.*	79.48 *min.*
	Prediction Time	4.26 *s*	4.00 *s*	4.10 *s*	3.86 *s*	3.65 *s*
	LST-NER [66]	70.44	<u>66.83</u>	<u>72.08</u>	<u>67.12</u>	<u>60.32</u>
Ours *Divide-and-Transfer*	DTrans-SMix	76.70	72.35	76.10	69.22	68.93
	Improv.	+6.14	+5.52	+4.02	+2.10	+8.61
	Training Time	127.26 *min.*	126.09 *min.*	127.30 *min.*	125.08 *min.*	126.16 *min.*
	Prediction Time	3.02 *s*	2.64 *s*	2.04 *s*	2.06 *s*	2.15 *s*
	DTrans-MPrompt	**80.54**	**73.06**	**79.54**	**73.51**	**70.13**
	Improv.	+9.98	+6.23	+7.46	+6.39	+9.81
	Training Time	76.82 *min.*	77.91 *min.*	70.77 *min.*	70.23 *min.*	71.32 *min.*
	Prediction Time	4.62 *s*	3.88 *s*	3.29 *s*	3.00 *s*	3.05 *s*

Bold marks the highest number among all methods. <u>Underline</u> indicates the prior SOTA methods. *Italic* number indicates the absolute increase compared with the prior SOTA. † marks produced with official implementation. "*min.*" means minute and "*s*" means second.

tagging scheme (e.g., B-LOC) or generative framework. The source knowledge cannot be transferred into target domains sufficiently owing to different transferability for two kinds of entity information. Another reason may be the different settings between the cross-domain and the few-shot or low-resource scenario where few-shot NER exploits the support set of each entity category for learning general patterns, and low-resource NER focuses on limited target data. Cross-domain NER tends to study cross-domain strategies for transferring knowledge from the high-resource domain to low-resource ones. Compared with DTrans-SMix that adopts parameter sharing and intermediate domain augmentation cross-domain strategies, DTrans-MPrompt gets a better effect on all of the domain pairs. The main reason may be that the prompt-tuning strategy modifies the type prediction into a unified label word prediction task despite different entity categories across domains, which obviously bridges the gap on entity categories between the source and target domain. Meanwhile, the multi-view decoding strategy profits entity span cross-domain transfer by capturing more domain-invariant features.

4.2.1 Parameter Analysis. To dispel concerns over multiple models (more parameters) in our claimed divide-and-transfer paradigm, we show previous SOTA performance under different parameters in Table 5, when respectively transferring from CoNLL2003 and Twitter to five target domains. We can see that our framework DTrans-SMix with 216.6M parameters and DTrans-

Table 4. F1 Scores on Another Five Domain Pairs That Transfer from the Source Domain Social Media to Five Target Domains, Respectively

	Source Domain	Twitter (Social Media) →				
	Target Domain	**Politics**	**Natural Science**	**Music**	**Literature**	**AI**
Methods	BiLSTM-CRF [25]	53.64	47.33	48.85	45.23	44.08
	Coach [34]	55.03	50.22	49.91	44.88	42.98
	LM-NER [19]	66.99	64.23	61.48	59.09	50.46
	NNShot [60]†	69.13	64.59	56.78	53.97	51.02
	StructShot [60]†	71.27	<u>65.24</u>	61.82	58.47	57.30
	Template [7]†	66.70	64.98	64.87	61.42	56.68
	BERT-JF [35]	67.52	64.51	67.74	61.38	57.05
	BERT-PF [35]	68.60	62.23	68.06	61.91	54.72
	MultiCell-LM [21]	66.59	63.79	66.54	59.02	53.82
	Style-NER [3]†	67.33	63.14	67.12	62.06	57.76
	LightNER [4]†	68.49	62.57	66.02	62.40	52.86
	Training Time	8.82 *min.*	7.50 *min.*	7.40 *min.*	7.66 *min.*	7.58 *min.*
	Prediction Time	11.04 *s*	9.22 *s*	7.77 *s*	5.18 *s*	6.64 *s*
	EntLM [38]†	<u>71.34</u>	64.59	<u>68.10</u>	<u>63.77</u>	<u>59.85</u>
	Training Time	29.23 *min.*	25.06 *min.*	25.79 *min.*	23.49 *min.*	21.99 *min.*
	Prediction Time	4.27 *s*	4.06 *s*	4.19 *s*	3.72 *s*	3.58 *s*
Ours *Divide-and-Transfer*	DTrans-SMix	74.62	71.37	74.41	69.67	64.55
	Improv.	+3.28	+6.13	+6.31	+5.90	+4.70
	Training Time	75.74 *min.*	77.76 *min.*	77.05 *min.*	73.74 *min.*	77.86 *min.*
	Prediction Time	3.21 *s*	2.68 *s*	2.46 *s*	2.10 *s*	2.16 *s*
	DTrans-MPrompt	**79.86**	**73.18**	**77.93**	**72.74**	**69.13**
	Improv.	+8.52	+7.94	+9.83	+8.97	+9.28
	Training Time	22.46 *min.*	22.80 *min.*	21.96 *min.*	22.62 *min.*	21.62 *min.*
	Prediction Time	4.63 *s*	3.91 *s*	3.30 *s*	3.00 *s*	3.06 *s*

Bold marks the highest number among all methods. <u>Underline</u> indicates the prior SOTA methods. *Italic* number indicates the absolute increase compared with the prior SOTA. † marks produced with official implementation. "*min.*" means minute and "*s*" means second.

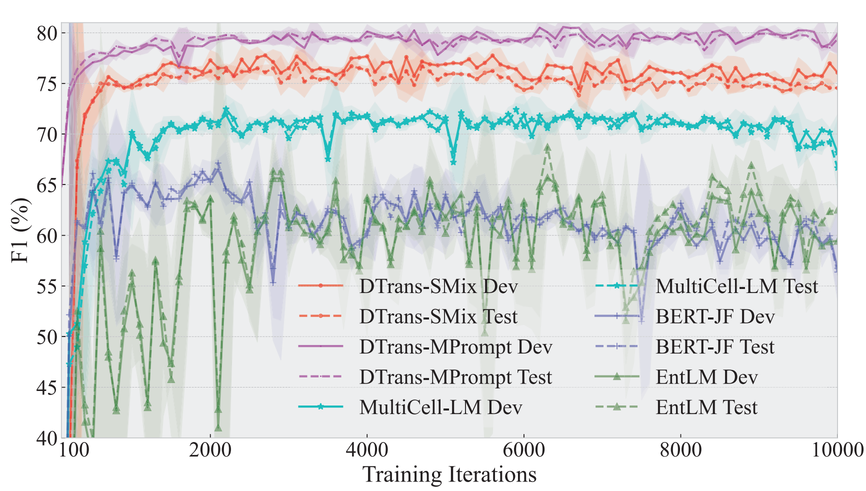

Fig. 6. F1 score vs. Training iterations (CoNLL2003 to the Music domain).

Table 5. Average F1 Score over Five Target Domains (CoNLL2003 or Twitter as the Source Domain) with Different Numbers of Parameters

CoNLL2003 **Method**	F1 (Averaged)	#Parameter	Speed (batches/seconds) ↑
BERT-JF ($BERT_{BASE}$)	64.52	108.9M	15.0 B/s
BERT-JF ($BERT_{LARGE}$)	67.88	334.7M	6.8 B/s
MultiCell-LM ($BERT_{BASE}$)	66.55	119.5M	2.6 B/s
MultiCell-LM ($BERT_{LARGE}$)	67.13	344.7M	2.1 B/s
EntLM ($BERT_{BASE}$)	64.57	108.3M	4.4 B/s
EntLM ($BERT_{LARGE}$)	68.41	333.6M	1.8 B/s
LightNER ($BART_{BASE}$)	64.35	164.6M	1.9 B/s
LightNER ($BART_{LARGE}$)	70.37	469.9M	1.1 B/s
DTrans-SMix ($BERT_{BASE}$)	72.66	216.6M	6.7 B/s
DTrans-MPrompt ($BERT_{BASE}$)	**75.36**	218.4M	4.3 B/s
Twitter **Method**	F1 (Averaged)	#Parameter	Speed (batches/seconds) ↑
BERT-JF ($BERT_{BASE}$)	63.64	108.9M	15.1 B/s
BERT-JF ($BERT_{LARGE}$)	65.56	334.7M	6.5 B/s
MultiCell-LM ($BERT_{BASE}$)	61.95	119.5M	2.7 B/s
MultiCell-LM ($BERT_{LARGE}$)	63.86	344.7M	2.4 B/s
EntLM ($BERT_{BASE}$)	65.53	108.3M	3.9 B/s
EntLM ($BERT_{LARGE}$)	67.63	333.6M	2.0 B/s
LightNER ($BART_{BASE}$)	62.47	164.6M	2.0 B/s
LightNER ($BART_{LARGE}$)	67.61	469.9M	1.2 B/s
DTrans-SMix ($BERT_{BASE}$)	70.92	216.6M	6.2 B/s
DTrans-MPrompt ($BERT_{BASE}$)	**74.57**	218.4M	4.4 B/s

B/s refers to the processed number of batches per second during the test.

MPrompt with 218.4M parameters significantly outperform BERT-JF (334.7M), MultiCell-LM (344.7M) and EntLM (333.6M) with $BERT_{LARGE}$, and LightNER (469.9M) with $BART_{LARGE}$, which indicates that divide-and-transfer does not mainly gain from more parameters but from the disentangled entity information considering distinct transferability of entity span and type. For the running efficiency under the same batch size and experimental environment, our two frameworks are still acceptable. In fact, two sub-tasks in DTrans-SMix can be processed in parallel, which will further accelerate the efficiency. Although *type prediction* in DTrans-MPrompt needs to construct the input for each entity span, the *entity detection* sub-task reduces the number of candidate spans and inputs in type prediction can be processed in batches.

4.2.2 Ablation Studies. We evaluate the influence of each component from our DTrans-MPrompt framework in Table 6. We can observe the following:

(1) In DTrans-MPrompt, without the multi-view decoding strategy (MVD) (i.e., only keeping the "BIO" tagging scheme), the score drops from 76.14% to 74.43% with CoNLL2003 as the source domain and 75.25% to 74.13% with Twitter as the source. The reason is that MVD benefits from the more domain-invariant features in different views.
(2) In DTrans-MPrompt, replacing *Prompt-tuning* with token type classification shows that prompt-tuning respectively contributes to 2.92% and 2.08% increase with CoNLL2003 and Twitter as the source domain, as it lessens the domain gap by the unified label word prediction, which possesses better ability to bridge the domain gap caused by the mismatch between the different domain-specific entity types.

Table 6. Ablation Studies of Our DTrans-MPrompt Framework on dev Sets

Framework	**Dev F1** (CoNLL2003)	**Dev F1** (Twitter)
DTrans-MPrompt with ❶($\mathcal{E}_{\mathrm{BIO}}$) as candidate spans $\mathcal{E}$	**76.14**	**75.25**
w/o Multi-view decoding strategy in ED	74.43	74.13
w/o Prompt-tuning in TP	73.22	72.99
w/o Re-detecting strategy	74.98	74.33
w/o Mean teaching	75.31	74.84
w ❷ as candidate spans $\mathcal{E}$	73.16	73.17
w ❶($\mathcal{E}_{\mathrm{SE}}$) as candidate spans $\mathcal{E}$	74.01	73.39
w ❶($\mathcal{E}_{\mathrm{TB}}$) as candidate spans $\mathcal{E}$	61.08	60.05

Scores are averaged over five target domains (CoNLL2003 or Twitter as the source domain).

Table 7. F1 Score Gain Comparison of the Cross-Domain Transfer Strategy in Two Sub-Tasks between DTrans-SMix and DTrans-MPrompt

Gains		**CoNLL2003 →**					
Δ		**Politics**	**Natural Science**	**Music**	**Literature**	**AI**	Avg
Entity Detection	DTrans-SMix	0.38	0.52	0.75	0.08	0.87	0.52
	DTrans-MPrompt	0.75	1.18	2.37	2.71	1.54	**1.71**
Type Prediction	DTrans-SMix	3.23	0.30	0.43	0.33	1.44	1.15
	DTrans-MPrompt	3.43	3.42	3.02	3.56	1.16	**2.92**
Gains		**Twitter →**					
Δ		**Politics**	**Natural Science**	**Music**	**Literature**	**AI**	Avg
Entity Detection	DTrans-SMix	0.24	0.05	0.24	0.17	0.40	0.22
	DTrans-MPrompt	0.54	0.25	1.20	0.49	1.66	**0.83**
Type Prediction	DTrans-SMix	1.13	1.29	0.66	0.91	1.69	1.14
	DTrans-MPrompt	4.15	2.94	3.14	3.13	4.94	**3.66**

(3) Due to the pipeline structure between two sub-tasks in DTrans-MPrompt, the results are affected by error accumulation. Re-detecting strategy increases the F1 score by filtering the false-positive candidate entity spans to alleviate the accumulative error. Overall, interaction between two sub-tasks under the divide-and-transfer paradigm is necessary, and devising corresponding interaction strategies remains an open problem.
(4) In light of low-resource target domains, the mean teaching strategy in DTrans-MPrompt improves the performance due to its stable optimization and prevents from overfitting the limited target-domain training data.
(5) Otherwise, in DTrans-MPrompt, ensembling the outputs of multiple span decoding strategies [❷] is worse than choosing the "BIO" scheme [❶ ($\mathcal{E}_{\mathrm{BIO}}$)] during generating candidate entity spans because of more false-positive spans and decoding conflicts among three schemes, which bring the great burden for the *type prediction* sub-task. Choosing $\mathcal{E}_{\mathrm{SE}}$ or $\mathcal{E}_{\mathrm{TB}}$ as candidates leads to the poor F1 score because the SE scheme profits the long entity and TB cannot detect single-token entities.

4.2.3 Comparison of the Cross-Domain Transfer Strategy between DTrans-SMix and DTrans-MPrompt. As shown in Table 7, for two divide-and-transfer paradigm based frameworks, we report the F1 score gains before and after adopting corresponding sub-task cross-domain transfer

Table 8. F1 Score Gains of the Target Domain from the Source Domain by Transfer

Without/with Source Domain	CoNLL2003 ⟶ Five *Low-Resource* domains	Twitter ⟶ *High-Resource* BioMedical
MultiCell-LM	↑2.40 (64.15/66.55)	↑1.42 (78.76/80.18)
BERT-JF	↑2.66 (61.86/64.52)	↑1.55 (79.17/80.72)
Style-NER	↑2.13 (61.44/63.57)	↑2.20 (79.60/81.80)
EntLM	↑1.56 (63.01/64.57)	↑1.57 (80.71/82.28)
Divide	↑3.12 (68.26/71.38)	↑2.10 (80.45/82.55)
DTrans-SMix	↑**4.17** (68.49/72.66)	↑**2.60** (80.83/83.43)
DTrans-MPrompt	↑**4.33** (71.03/75.36)	↑**3.69** (80.38/84.07)

↑ means the increase after using the source.

strategies on dev sets with CoNLL2003 and Twitter as the source domains, respectively. We can see that cross-domain transfer strategies of DTrans-MPrompt in entity detection and type prediction sub-tasks both achieve greater gains compared with our previously published DTrans-SMix [65]. The reason is that the *multi-view decoding strategy* and *prompt-tuning-based label space unification* can respectively capture more domain-invariant entity span features and exploit entity type correlations across domains. The cross-domain transfer strategies from DTrans-MPrompt in this article fully tap into the potential of the divide-and-transfer paradigm.

4.3 Discussion

4.3.1 Gains from the Source Domain (RQ2). As shown in Table 8, to show the advantage of our divide-and-transfer paradigm in cross-domain transfer, we compare the average performance gains on five low-resource target domains before and after using the source domain data from CoNLL2003. The blue numbers mean only using the target domain data, and the red ones represent using both the source and target data. We see that DTrans-SMix and DTrans-MPrompt both gain more from the source domain data (4.17% and 4.33% absolute increase) than previous SOTAs based on sequence labeling. This shows the efficacy of divide-and-transfer, which disentangles the coupled information and devises the corresponding transfer strategies in each sub-task. That is to say, more information can be transferred from the source to the target domain under the divide-and-transfer paradigm than sequence labeling, which effectively confirms our motivation and more effective transfer in cross-domain NER. To explore the effectiveness of dividing the NER task and eliminate the interference from transfer strategies, we only jointly train the DTrans-SMix framework with a specific classification head across domains in each sub-task (notated as *Divide* in Table 8), same as BERT-JF [35]. We see that task decomposition with the same transfer strategy still achieves significant gains, which shows that dividing the NER task can unearth the transfer strategies and contribute to the information transfer.

Surprisingly, our two frameworks without using the source domain data even significantly surpass previous SOTAs with the source data in Table 8 (Figure 7 also shows this point). That is because disentangling the information by dividing the NER task benefits the low-resource NER a lot. The NER task decomposition provides a better basis for cross-domain NER. Furthermore, we also perform the cross-domain transfer in the high-resource scenario, where Twitter [36] with 4,290 training sentences is the source domain and BioMedical [40] with 3,033 training sentences is the target (with 1,003 development and 1,906 test sentences). We see that our two frameworks still obtain 2.60% and 3.69% gains, and the advantage of our proposed paradigm without using the source domain over others with the source is reasonably reduced.

Table 9. F1 Scores under Different Settings on Five Target Domains

	Politics	Natural Science	Music	Literature	AI
BERT-DF [35]	66.56	63.73	66.59	59.95	50.37
BERT-JF (CoNLL2003) [35]	68.85	65.03	67.59	62.57	58.57
BERT-JF (Twitter)	67.52	64.51	67.74	61.38	57.05
DTrans-SMix *w/ only target domain*	71.15	70.40	74.10	66.74	60.05
DTrans-SMix (CoNLL2003)	76.70	72.35	76.10	69.22	68.93
DTrans-SMix (Twitter)	74.62	71.37	74.41	69.67	64.55
DTrans-MPrompt *w/ only target domain*	75.24	71.82	75.54	69.24	63.30
DTrans-MPrompt (CoNLL2003)	80.54	73.06	79.54	73.51	70.13
DTrans-MPrompt (Twitter)	79.86	73.18	77.93	72.74	69.13

BERT-DF means requiring no source domain data and directly fine-tuning BERT on the target domain; results are reported from Liu et al. [35]. *BERT-JF* means that BERT-DF requires source domain data by jointly training on both CoNLL2003/Twitter and the target domain data. *DTrans-SMix w/only target domain* and *DTrans-MPrompt w/ only target domain* indicate that we do not use source domain data. *DTrans-SMix (CoNLL2003)* and *DTrans-MPrompt (CoNLL2003)* mean that we use CoNLL2003 as the source domain. *DTrans-SMix (Twitter)* and *DTrans-MPrompt (Twitter)* indicate that we use the Twitter source domain data.

As shown in Table 9, the methods that only require target domain data are significantly worse than those approaches using both source domain and target domain data. Although the pre-trained models (e.g., BERT) have strong transfer learning ability, they are pre-trained on the corpus collected from the general domain, and the corpus is not related to the NER task. Therefore, it has a limited impact on improving the target domain in comparison with using source domain NER data. In fact, Figure 7 also reports the results of our method DTrans-SMix and DTrans-MPrompt without using source domain data (DTrans-SMix w/o Source Domain, DTrans-MPrompt w/o Source Domain). We can see that using the source domain data brings significant improvements in both DTrans-SMix and DTrans-MPrompt, especially in the extremely low-resource scenario, which shows the necessity and effectiveness of cross-domain transfer in the low-resource NER. As shown in Table 9, we report the performance of some method variants (e.g., BERT-DF, DTrans-SMix *w/ only target domain*, and DTrans-MPrompt *w/ only target domain*) which do not use source domain data, and see that these variants still have a significant performance gap compared to those methods using source domain data, such as DTrans-SMix (CoNLL2003), DTrans-SMix (Twitter), DTrans-MPrompt (CoNLL2003), and DTrans-MPrompt (Twitter).

4.3.2 Effect of the Target Domain Data Size. As depicted in Figure 7, we study the performance changes with different numbers of training data from the target domain. We can see the following *as the number of target domain data is reduced*. First, the F1 score drops, which reflects the difficulty of the NER task in the extremely low-resource scenario. Second, gains from using the source domain data become greater whether in DTrans-SMix or in DTrans-MPrompt, which shows the necessity and effectiveness of cross-domain transfer in the low-resource NER. Third, our two frameworks outperform previous SOTAs with large margins, which shows the superiority of the divide-and-transfer paradigm in the scenario of low resources. Additionally, DTrans-MPrompt with only 10% or 25% target domain data (Music or Science) can rival with previous SOTAs of using the full target data. The significant advantage over prior competitive baselines from that our divide-and-transfer paradigm can disentangle the coupled information, then benefit from better performance on low-resource data and more effective transfer based on the tailor-designed transfer strategy in each sub-task.

4.3.3 Error Analysis. As shown in Table 10, we show the performance of two sub-tasks (entity detection and type prediction, ED and TP) in DTrans-SMix and DTrans-MPrompt where the TP

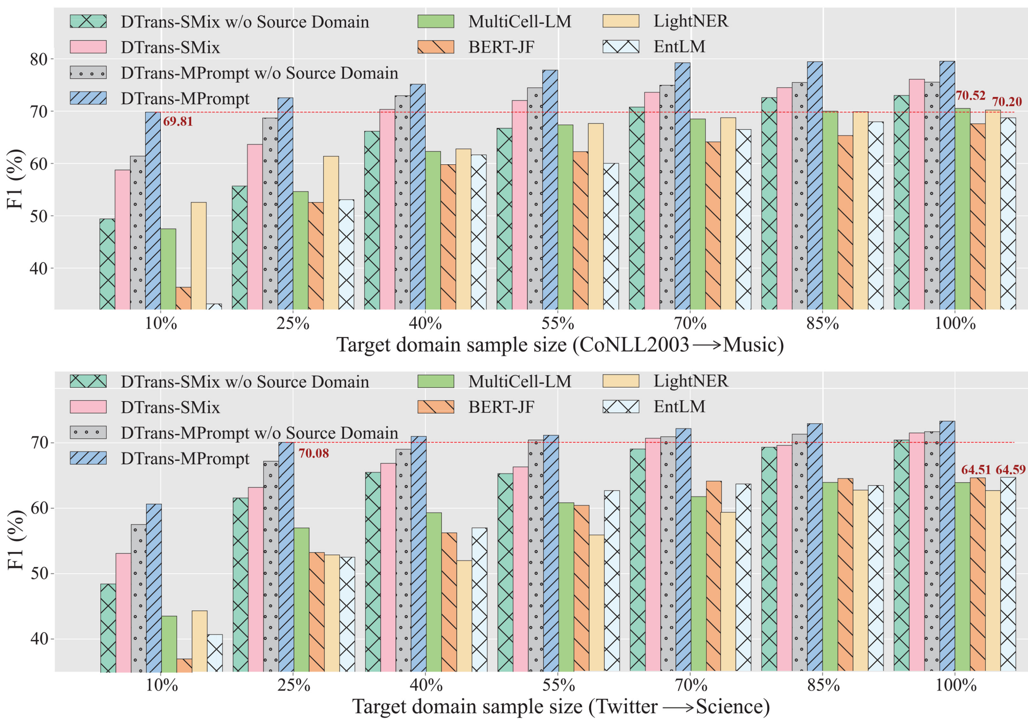

Fig. 7. F1 score vs. data size in Music and Science target domains (averaged over three samplings, with CoNLL2003 and Twitter as the source, respectively).

Table 10. F1 Scores of Two Divided Sub-Tasks and Their Final Combination NER Result, Where CoNLL2003 and Twitter Respectively Serve as the Source Domain and Five Target Domains

DTrans	CoNLL2003 ⟶	Politics	Natural Science	Music	Literature	AI	Average
SMix	Entity Detection	90.16	85.42	90.00	89.41	83.99	87.80
	Type Prediction	81.62	81.05	83.28	75.64	76.82	79.68
	NER Final	76.70	72.35	76.10	69.22	68.93	72.66
MPrompt	Entity Detection	91.06	86.42	91.05	90.33	84.57	88.69
	Type Prediction	88.00	83.91	86.66	80.80	81.04	84.08
	NER Final	80.54	73.06	79.54	73.51	70.13	75.36
DTrans	**Twitter ⟶**	**Politics**	**Natural Science**	**Music**	**Literature**	**AI**	**Average**
SMix	Entity Detection	90.07	86.16	89.58	88.77	81.98	87.31
	Type Prediction	78.60	78.93	81.47	75.89	76.01	78.18
	NER Final	74.62	71.37	74.41	69.67	64.55	70.92
MPrompt	Entity Detection	90.93	86.66	90.48	89.51	83.90	88.30
	Type Prediction	86.77	84.91	85.79	81.07	80.76	83.86
	NER Final	79.86	73.18	77.93	72.74	69.13	74.57

Table 11. F1 Scores on Test Sets When Using Different Prompts in DTrans-MPrompt, and CoNLL2003 and Twitter Respectively Serve as the Source Domain

Prompt	CoNLL2003 → Politics	Natural Science	Music	Literature	AI	Avg
① New York is the [MASK] entity	80.54	73.06	79.54	73.51	70.13	**75.36**
② New York belongs to [MASK] category	81.14	72.05	79.45	72.59	68.41	74.73
③ New York should be tagged as [MASK] category	80.46	72.37	80.38	72.68	68.79	74.94
④ The entity type of New York is [MASK]	79.67	73.04	78.45	71.88	69.06	74.42

Prompt	Twitter → Politics	Natural Science	Music	Literature	AI	Avg
① New York is the [MASK] entity	79.86	73.18	77.93	72.74	69.13	**74.57**
② New York belongs to [MASK] category	80.43	71.87	78.86	72.08	67.08	74.06
③ New York should be tagged as [MASK] category	79.58	73.31	78.55	71.95	67.84	74.25
④ The entity type of New York is [MASK]	80.17	73.08	77.35	72.97	67.95	74.30

Take the entity span "New York" for example.

sub-task uses the ground-truth entity spans as input. We observe that F1 scores of the ED sub-task have achieved an average of 87.80% and TP reaches 79.68% with CoNLL2003 as the source domain, and 87.31% and 78.18% with Twitter as the source domain in DTrans-SMix. Thus, the bottleneck of main results lies in the TP sub-task because of distinct label sets across domains, larger label spaces, and intractable tasks on main datasets compared to ED, hindering the cross-domain and few labeled learning. Additionally, the F1 scores of two sub-tasks in DTrans-MPrompt are comparable on most datasets, whereas their final combinations (the candidate spans generated from ED as TP's input) are obviously lower. Thus, the factor restricting improvements in DTrans-MPrompt mainly originates from error propagation between two sub-tasks, which remains an open problem in this cascaded architecture. We propose a simple re-detecting strategy to alleviate this issue effectively, but it still needs further exploration.

Compared to DTrans-SMix, the prompt-tuning strategy in DTrans-MPrompt unifies the prediction task, benefiting the cross-domain learning of entity typing and then improving the TP sub-task significantly. However, prompt-tuning leads to the pipeline structure in DTrans-MPrompt while DTrans-SMix is parallel between the ED and TP sub-tasks. Therefore, interactive combinations between two sub-tasks in the divide-and-transfer paradigm need more in-depth exploration. This article mainly focuses on exploring the efficacy of divide-and-transfer in cross-domain NER where some components may be simple, but it still outperforms previous SOTAs with large margins (average 5.27% and 8.44% absolute F1 score increase), which highlights the advantage and great potential of this paradigm in the future, providing a new insight into cross-domain NER.

4.3.4 Prompt Analysis. To explore the effect of prompt in *Prompt-Tuning-based Label Space Unification* of DTrans-MPrompt, we utilize different prompts and report their F1 scores in Table 11. We can observe that DTrans-MPrompt is not very sensitive to specific prompts, which shows its stability and robustness. The case also illustrates that *Prompt-Tuning-based Label Space Unification* benefits from the unified entity type space for exploiting label correlations across domains rather than manual prompt engineering. Comprehensively, the first prompt in Table 11 is the most

Table 12. F1 Scores of DTrans-MPrompt under Two Training Paradigms

Source Domain	Training Paradigm	Politics	Natural Science	Music	Literature	AI	Avg
CoNLL2003	*Pretrain-then-fine-tuning*	80.15	**75.81**	**80.26**	73.14	69.96	**75.86**
	Joint training	**80.54**	73.06	79.54	**73.51**	**70.13**	75.36
Twitter	*Pretrain-then-fine-tuning*	79.84	**74.31**	**78.54**	**73.18**	**69.30**	**75.03**
	Joint training	**79.86**	73.18	77.93	72.74	69.13	74.57

Table 13. Cross-Domain F1 Scores on MIT Movie and ATIS under Few-Shot Settings, Where CoNLL2003 Is the Source Domain

Target Domain	**MIT Movie**						**ATIS**		
K-shot	10	20	50	100	200	500	10	20	50
Example. [69]	40.1	39.5	40.2	40.0	40.0	39.5	17.4	19.8	22.2
MP-NSP [18]	36.4	36.8	38.0	38.2	35.4	38.3	71.2	74.8	76.0
NNShot [60]	42.6	52.6	55.5	75.8	79.1	–	89.2	92.8	94.3
StructShot [60]	44.4	57.0	61.8	<u>77.4</u>	<u>79.6</u>	–	<u>89.8</u>	<u>93.1</u>	<u>94.5</u>
Template [7]	42.4	54.2	59.6	65.3	69.6	80.3	77.3	88.9	93.5
LightNER [4]	<u>54.6</u>	<u>65.1</u>	<u>71.0</u>	67.9	76.2	<u>83.0</u>	82.3	88.6	92.2
EntLM [38]	39.5	57.6	69.7	75.7	78.9	82.5	81.3	87.0	92.8
DTrans-SMix (Ours)	61.9	73.8	78.6	80.5	81.9	85.3	93.2	94.9	95.8
DTrans-MPrompt (Ours)	**63.2**	**74.6**	**79.3**	**81.1**	**82.9**	**85.4**	**93.6**	**95.0**	**96.3**

Bold marks the highest number among all methods. <u>Underline</u> indicates the prior SOTA methods. For fair comparison, taking $\text{BERT}_{\text{BASE}}$ or $\text{BART}_{\text{BASE}}$ in competitive baselines with official implementation.

intuitive and effective, and DTrans-MPrompt constructs the first one for each candidate entity span in this article.

4.3.5 Training Paradigm Analysis. For cross-domain NER, there are usually two training paradigms: *pretrain-then-fine-tuning* and *joint training*. *Pretrain-then-fine-tuning* indicates that the model is first trained in the source domain and then fine-tuned in the target domain. *Joint training* means that we train the model in the source and target domains jointly. As shown in Table 12, we report the performance of respectively transferring from CoNLL2003 and Twitter to five target domains under these two training paradigms. We only show the results of DTrans-MPrompt, because DTrans-SMix proposed in our conference version [65] simultaneously requires source and target domain data for constructing the intermediate augmented domain, which can only be trained under the *joint training* paradigm. Comprehensively, training with both source and target domain data jointly may not lead to better results, which is consistent with the conclusion drawn by LST-NER [66] and CrossNER [35]. However, the focus of this work is not on using *pretrain-then-fine-tuning* or *joint training*, but on decoupling the NER task and devising cross-domain strategies. Even in such circumstances, our method still achieves significant improvements.

4.4 Few-Shot Scenario

In this subsection, we evaluate the model performance under the few-shot setting, where few-shot datasets serve as target domains and CoNLL2003 is the source domain. Table 13 reports the few-shot results with K instances for each entity category in target domains. Besides baselines in the main results, we also consider Example [69] (a few-shot NER learning method inspired by extractive question answering) and MP-NSP [18] (a prototype-based method). The hyperparameter settings use the majority of tuned values from the main experiments.

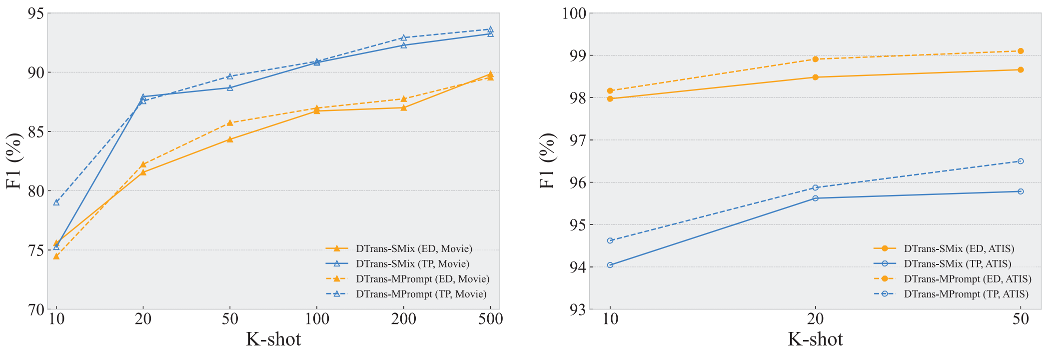

Fig. 8. F1 scores of two divided sub-tasks on K-shot datasets.

We can see that our two frameworks still achieve new SOTAs on all datasets in Table 13, which shows the great generalization of the divide-and-transfer paradigm, especially in an extremely few-shot scenario (e.g., 8.6% increase on MIT Movie under 10-shot). Our advantage over prior baselines becomes increasingly obvious as K decreases, whereas our method achieves limited improvements in relatively high resource scenarios (e.g., K = 200, 500) due to rich target domain data. Overall, the extensive experiments confirm the superiority of the divide-and-transfer paradigm on cross-domain NER because of the entity information disentanglement and distinct cross-domain transfer strategies for entity span and type information.

In comparison with DTrans-SMix, DTrans-MPrompt further improves the performance of the divide-and-transfer paradigm on two few-shot datasets. To explore the reason behind this, Figure 8 gives F1 scores of the entity detection (ED) and type prediction (TP) sub-task on K-shot datasets. We can observe that DTrans-MPrompt achieves consistent advantages over DTrans-SMix on the entity detection (ED) sub-task under different K-shots owing to more shared domain-invariant information by the multi-view decoding strategy. Likewise, DTrans-MPrompt shows its preponderance on the type prediction (TP) sub-task as a result of the unified label space that exploits label correlation across domains and narrows down the domain gap. All in all, DTrans-MPrompt fulfills more precise and concise cross-domain strategies in two sub-tasks, which contributes to more effective transfer.

4.5 Zero-Shot Scenario (RQ3)

In this section, we evaluate the zero-shot transfer ability of the proposed DTrans-MPrompt, compensating for the deficiency that DTrans-SMix [65] cannot deal with zero-shot scenarios. Under zero-shot scenario, the model is only trained on source domain data and directly tested on target domain data. Concretely, our DTrans-MPrompt is trained with loss functions related to source domain data—that is, $\mathcal{L}^{\mathcal{S}}_{\text{BIO}}$, $\mathcal{L}^{\mathcal{S}}_{\text{SE}}$, $\mathcal{L}^{\mathcal{S}}_{\text{TB}}$, and $\mathcal{L}^{\mathcal{S}}_{\text{PT}}$ in Equation (7). As shown in Table 14, our proposed method in this article achieves consistent advantages when transferring between domains of *Science*, *Literature*, and *Music* in pairs, as task decomposition makes the smaller domain gap in entity detection and type prediction sub-tasks. What is more, the *multi-view decoding strategy* can capture the intrinsic entity boundary information in the entity detection sub-task and *prompt-tuning* has stronger domain generalization ability for typing entities.

To go into in-depth understanding of the domain generalization ability on two sub-tasks, and show the performance on seen and unseen entity types[3] between source and target domains, we

[3]Seen entity type means that types appear in both source and target domains, and unseen entity type means that types only appear in the target domain.

Table 14. Macro-F1 Scores of Six Cross-Domain Pairs under Zero-Shot Scenarios

Source	Target	LUKE [56]	DOZEN* [39]	DOZEN [39]	DTrans-MPrompt (Ours)
Science →	*Literature*	34.4	33.8	37.2	**40.6**
	Music	26.1	28.8	28.4	**37.1**
Literature →	*Science*	26.8	31.6	32.7	**34.5**
	Music	32.2	35.5	41.5	**42.4**
Music →	*Science*	22.7	25.3	26.5	**28.9**
	Literature	**49.5**	44.5	48.5	49.1

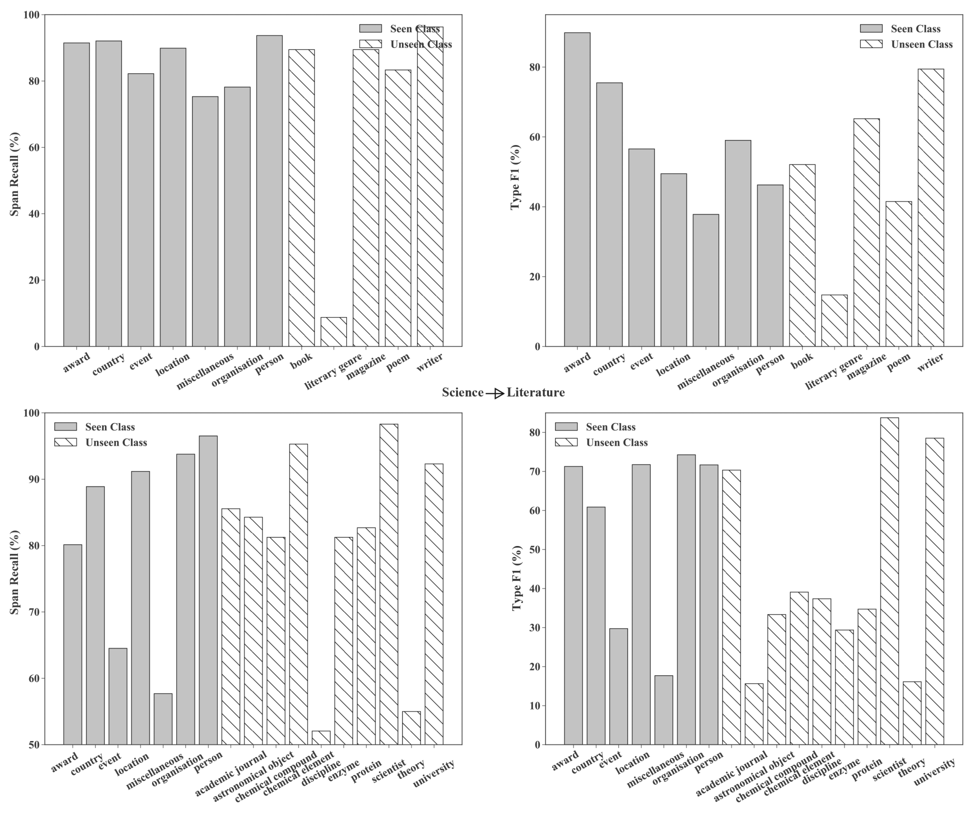

Fig. 9. Performance of entity detection and type prediction sub-tasks over each entity category when transferring between the Science and Literature domains.

calculate the recall of entity span and F1 of entity type over each type in Figures 9, 10, and 11. Because the entity type is unknown in the entity detection sub-task, we cannot get a precision score for each type and then the recall rate is reported for the entity span. We can observe that our DTrans-MPrompt achieves credible performance on both entity span and type for seen and unseen types. Interestingly, the performance differences between seen and unseen types are lower on entity span detection than type prediction. The reason may be that the entity detection sub-task relies on grammar or syntactic information that is more generalized across domains, whereas entity type information is domain-dependent. As shown in Figures 9, 10, and 11, unseen types which

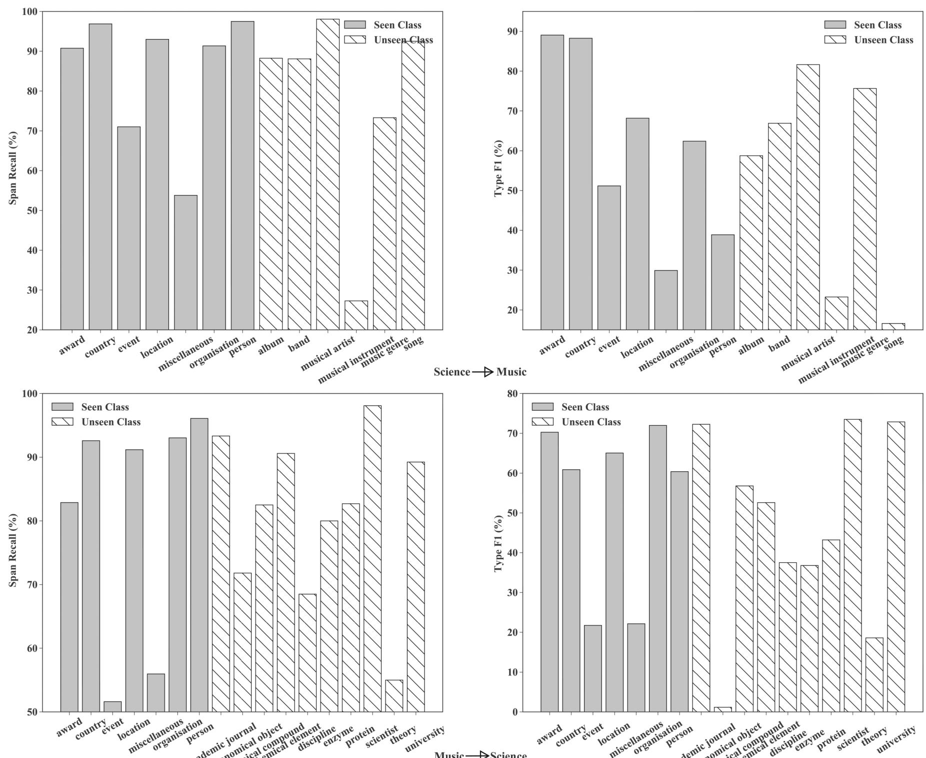

Fig. 10. Performance of entity detection and type prediction sub-tasks over each entity category when transferring between the *Science* and *Music* domains.

do not appear in the source domain can still be correctly classified to some extent. Additionally, regardless of seen and unseen types, performances of entity span detection and type prediction over several types (e.g., "literary genre") are poor due to domain gap and entity type differences. For example, the type "literary genre" generally refers to some abstract words, such as "novel" and "literary criticism," whereas most other types (e.g., book) generally refer to specific words, such as "The Forsyte Saga" and "Aesop's Fables." The type differences lead to inferior performance on entity span detection and type prediction under zero-shot scenarios.

4.6 Slot Filling Task (RQ4)

In this section, we evaluate the model performance on the slot filling task, which is also a classical sequence labeling task, same as NER. As shown in Table 15 and Table 16, there are seven target domains in total, and each domain serves as the target domain for test and the source domain is the left six domains following the setup of other works [34, 58]. Table 15 shows few-shot cross-domain transfer on 20 target-domain samples—that is, both the source domain and 20 target domain labeled samples are available for training. Similarly, Table 16 shows few-shot cross-domain transfer on 50 target domain samples. The proposed method DTrans-MPrompt in this article achieves consistent advantages over previous baselines on average.

DTrans-SMix and DTrans-MPrompt both follow the divide-and-transfer paradigm that performs task decomposition for NER and devises corresponding transfer strategies in each sub-task. Compared with our previously published DTrans-SMix [65], DTrans-MPrompt proposed in this

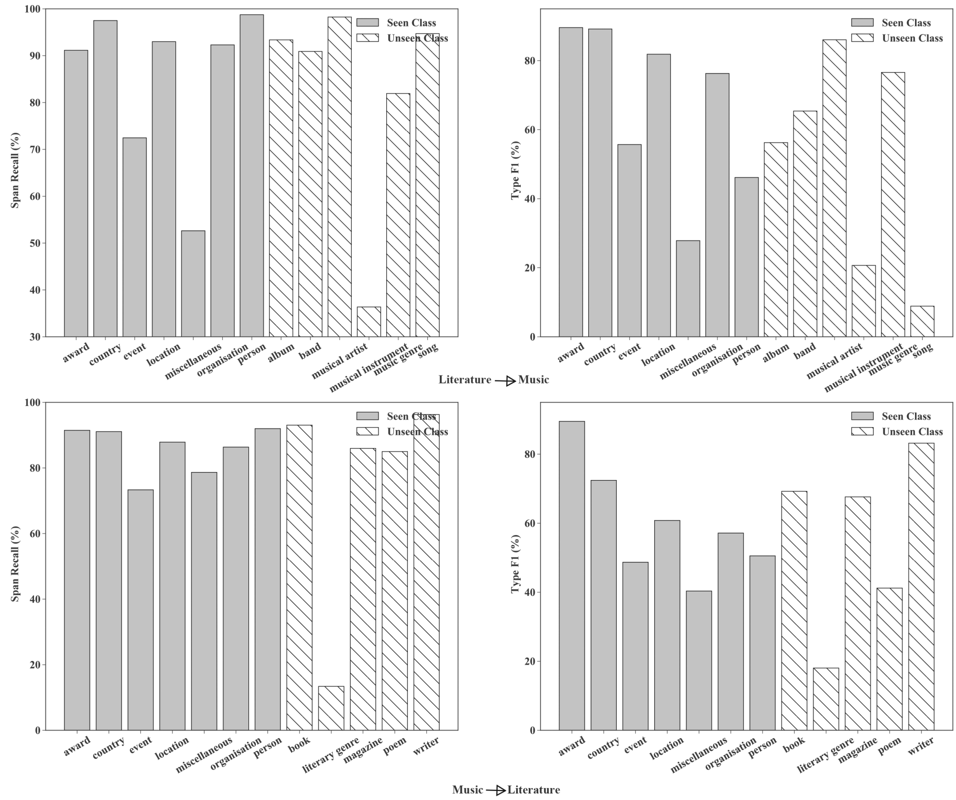

Fig. 11. Performance of entity detection and type prediction sub-tasks over each entity category when transferring between *Literature* and *Music* domains.

Table 15. Cross-Domain F1 Scores on the SNIPS Dataset for Different Target Domains under 20 Few-Shot Samples

Domain \ Method	CT [13]	RZT [46]	Coach [34]	AISFG [58]	DTrans (Ours)	
					SMix	MPrompt
AddToPlaylist	58.36	63.18	62.76	81.64	76.50	**84.54**
BookRestaurant	45.65	50.54	65.97	78.06	81.05	**82.67**
GetWeather	54.22	58.86	67.89	82.68	**88.78**	85.65
PlayMusic	46.35	47.20	54.04	**77.59**	71.81	74.45
RateBook	64.37	63.33	74.68	79.06	77.18	**93.54**
SearchCreativeWork	57.83	63.39	57.19	71.95	73.45	**75.33**
SearchScreeningEvent	48.59	49.18	67.38	73.91	69.72	**88.84**
Average	53.62	56.53	64.27	77.84	76.93	**83.57**

Bold marks the highest number among all methods.

article has further remarkable advantages on the slot filling task, as shown in Table 15 and Table 16. The reason is that slot types are precise and highly similar (e.g., slot type "playlist" and "music item"), and DTrans-MPrompt can probe the knowledge from PLM, exploit label correlation, and reduce the domain gap by the unified prediction form. By comparison, DTrans-SMix is

Table 16. Cross-Domain F1 Scores on the SNIPS Dataset for Different Target Domains under 50 Few-Shot Samples

Domain \ Method	CT [13]	RZT [46]	Coach [34]	AISFG [58]	DTrans (Ours)	
					SMix	MPrompt
AddToPlaylist	68.69	74.89	74.68	83.51	77.93	**88.66**
BookRestaurant	54.22	54.49	74.82	84.60	83.17	**86.11**
GetWeather	63.23	58.87	79.64	83.73	88.74	**89.33**
PlayMusic	54.32	59.20	66.38	78.79	75.31	**80.55**
RateBook	76.45	76.87	84.62	92.85	85.10	**95.32**
SearchCreativeWork	66.38	67.81	64.56	76.00	73.77	**78.30**
SearchScreeningEvent	70.67	74.58	83.85	91.29	83.02	**94.09**
Average	64.85	66.67	75.51	84.39	81.01	**87.48**

Bold marks the highest number among all methods.

based on a classification head for type prediction and cannot capture the label correlation. Additionally, we see that DTrans-SMix shows a significant advantage in the "GetWeather" target domain under few-shot learning with 20 samples. The reason may be that the *GetWeather* domain possesses more common slot types (e.g., "city," "country," "state," and "time range") across domains, and classification head based type prediction also achieves promising performance, and intermediate domain augmentation for type prediction effectively narrows the domain discrepancy. For the "PlayMusic" domain with 20 few-shot samples, AISFG [58] achieves the SOTA because of its tailor-designed template including domain descriptions, slot descriptions, and examples with context. As slot type "album" and "sort" have a similar context pattern—for example, **getting ready** in *play the getting ready by eason chan* is an *album* slot type, whereas **shall we talk** in *play shall we talk by eason chan* is a *sort* slot type—AISFG [58] using the template by incorporating some examples with context may untangle the confusion caused by the similar context pattern between different slot types and then shows the superiority under the few-shot learning with 20 samples. Overall, the proposed DTrans-MPrompt in this article achieves the new SOTA on the cross-domain slot filling task under few-shot learning settings due to the disentanglement of slot boundary and type information with corresponding cross-domain transfer strategies for each slot information. In the future, incorporating the slot descriptions into prompt-tuning may further improve the performance owing to more precise modeling of slot type semantics.

5 CONCLUSION AND FUTURE WORK

This article explored the efficacy of the divide-and-transfer paradigm in cross-domain NER. We divided the NER task into entity detection and type prediction sub-tasks to disentangle the coupled information existing in the sequence labeling framework, then designed the corresponding transfer strategies in each sub-task. Extensive experiments demonstrated its notable effect, which provides a new perspective on cross-domain NER. Additionally, we extended our frameworks to a wider range of application scenarios, such as the target domain with few-shot and zero-shot samples, which confirms the significant advantages of the formally summarized paradigm and instantiated framework in this article. For future work, interactions and result combinations between two sub-tasks need better solutions for unleashing the greater potential of the divide-and-transfer paradigm. The divide-and-transfer paradigm for LLMs (e.g., ChatGPT) also needs further exploration thereafter.

REFERENCES

[1] Ashutosh Baheti, Alan Ritter, and Kevin Small. 2020. Fluent response generation for conversational question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 191–207.

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Hennigan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Eds.). Vol. 33. Curran Associates, 1877–1901.

[3] Shuguang Chen, Gustavo Aguilar, Leonardo Neves, and Thamar Solorio. 2021. Data augmentation for cross-domain named entity recognition. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 5346–5356.

[4] Xiang Chen, Lei Li, Shumin Deng, Chuanqi Tan, Changliang Xu, Fei Huang, Luo Si, Huajun Chen, and Ningyu Zhang. 2022. LightNER: A lightweight tuning paradigm for low-resource NER via pluggable prompting. In *Proceedings of the 29th International Conference on Computational Linguistics*. 2374–2387.

[5] Yulong Chen, Yang Liu, Li Dong, Shuohang Wang, Chenguang Zhu, Michael Zeng, and Yue Zhang. 2022. AdaPrompt: Adaptive model training for prompt-based NLP. In *Findings of the Association for Computational Linguistics: EMNLP 2022*. Association for Computational Linguistics, 6057–6068.

[6] Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, Mael Primet, and Joseph Dureau. 2018. Snips Voice Platform: An embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190* (2018).

[7] Leyang Cui, Yu Wu, Jian Liu, Sen Yang, and Yue Zhang. 2021. Template-based named entity recognition using BART. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Association for Computational Linguistics, 1835–1845.

[8] 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*. 4171–4186.

[9] Ning Ding, Yulin Chen, Xu Han, Guangwei Xu, Xiaobin Wang, Pengjun Xie, Haitao Zheng, Zhiyuan Liu, Juanzi Li, and Hong-Gee Kim. 2022. Prompt-learning for fine-grained entity typing. In *Findings of the Association for Computational Linguistics: EMNLP 2022*. Association for Computational Linguistics, 6888–6901.

[10] Guanting Dong, Zechen Wang, Liwen Wang, Daichi Guo, Dayuan Fu, Yuxiang Wu, Chen Zeng, Xuefeng Li, Tingfeng Hui, Keqing He, Xinyue Cui, Qixiang Gao, and Weiran Xu. 2023. A prototypical semantic decoupling method via joint contrastive learning for few-shot named entity recognition. In *Proceedings of the 2023 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. 1–5.

[11] Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. 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)*. 3816–3830.

[12] Siddhant Garg and Goutham Ramakrishnan. 2020. BAE: BERT-based adversarial examples for text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 6174–6181.

[13] Jacopo Gobbi, Evgeny A. Stepanov, and Giuseppe Riccardi. 2018. Concept tagging for natural language understanding: Two decadelong algorithm development. In *Proceedings of the 5th Italian Conference on Computational Linguistics (CLiC-it '18)*. 224.

[14] Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. How close is ChatGPT to human experts? Comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597* (2023).

[15] Gokhan Tur, Asli Celikyilmaz, Yun-Nung Chen, Jianfeng Gao, Li Deng, and Ye-Yi Wang. 2016. Multi-domain joint semantic frame parsing using bi-directional RNN-LSTM. In *Proceedings of the 17th Annual Conference of the International Speech Communication Association (Interspeech '16)*. 715–719.

[16] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.), Proceedings of Machine Learning Research, Vol. 97. PMLR, 2790–2799.

[17] Shengding Hu, Ning Ding, Huadong Wang, Zhiyuan Liu, Jingang Wang, Juanzi Li, Wei Wu, and Maosong Sun. 2022. Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2225–2240.

[18] Jiaxin Huang, Chunyuan Li, Krishan Subudhi, Damien Jose, Shobana Balakrishnan, Weizhu Chen, Baolin Peng, Jianfeng Gao, and Jiawei Han. 2020. Few-shot named entity recognition: A comprehensive study. *arXiv:2012.14978* (2020).

[19] Chen Jia, Xiaobo Liang, and Yue Zhang. 2019. Cross-domain NER using cross-domain language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2464–2474.

[20] Chen Jia, Yuefeng Shi, Qinrong Yang, and Yue Zhang. 2020. Entity enhanced BERT pre-training for Chinese NER. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 6384–6396.

[21] Chen Jia and Yue Zhang. 2020. Multi-cell compositional LSTM for NER domain adaptation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 5906–5917.

[22] Young-Bum Kim, Karl Stratos, Ruhi Sarikaya, and Minwoo Jeong. 2015. New transfer learning techniques for disparate label sets. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*. 473–482.

[23] Diederik P. Kingma and Jimmy Lei Ba. 2015. ADAM: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

[24] Alex M. Lamb, Anirudh Goyal Alias Parth Goyal, Ying Zhang, Saizheng Zhang, Aaron C. Courville, and Yoshua Bengio. 2016. Professor Forcing: A new algorithm for training recurrent networks. In *Proceedings of the 30th Conference on Neural Information Processing Systems (NIPS)*. 1–9.

[25] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 260–270.

[26] Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt-tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

[27] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 7871–7880.

[28] Bo Li, Gexiang Fang, Yang Yang, Quansen Wang, Wei Ye, Wen Zhao, and Shikun Zhang. 2023. Evaluating ChatGPT's information extraction capabilities: An assessment of performance, explainability, calibration, and faithfulness. *arXiv preprint arXiv:2304.11633* (2023).

[29] Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. 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)*. 4582–4597.

[30] Vladislav Lialin, Vijeta Deshpande, and Anna Rumshisky. 2023. Scaling down to scale up: A guide to parameter-efficient fine-tuning. *arXiv:cs.CL/2303.15647* (2023).

[31] Bill Yuchen Lin and Wei Lu. 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.

[32] Jingjing Liu, Panupong Pasupat, Yining Wang, Scott Cyphers, and Jim Glass. 2013. Query understanding enhanced by hierarchical parsing structures. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*. 72–77.

[33] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021. GPT understands, too. *arXiv:2103.10385* (2021).

[34] Zihan Liu, Genta Indra Winata, Peng Xu, and Pascale Fung. 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.

[35] Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2021. CrossNER: Evaluating cross-domain named entity recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 13452–13460.

[36] Di Lu, Leonardo Neves, Vitor Carvalho, Ning Zhang, and Heng Ji. 2018. Visual attention model for name tagging in multimodal social media. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. 1990–1999.

[37] Ruotian Ma, Yiding Tan, Xin Zhou, Xuanting Chen, Di Liang, Sirui Wang, Wei Wu, and Tao Gui. 2022. Searching for optimal subword tokenization in cross-domain NER. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI)*. 4289–4295.

[38] Ruotian Ma, Xin Zhou, Tao Gui, Yiding Tan, Linyang Li, Qi Zhang, and Xuanjing Huang. 2022. Template-free prompt-tuning for few-shot NER. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 5721–5732.

[39] Hoang-Van Nguyen, Francesco Gelli, and Soujanya Poria. 2021. DOZEN: Cross-domain zero shot named entity recognition with knowledge graph. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. ACM, 1642–1646.

[40] Claire Nédellec, Robert Bossy, Jin-Dong Kim, Jung-Jae Kim, Tomoko Ohta, Sampo Pyysalo, and Pierre Zweigenbaum. 2013. Overview of BioNLP Shared Task 2013. In *Proceedings of the BioNLP Shared Task 2013 Workshop*.

[41] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll 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 F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, Vol. 35. Curran Associates, 27730–27744.

[42] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. Preprint.

[43] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* 21 (2020), 1–67.

[44] 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—Volume 4*. Association for Computational Linguistics, 142–147.

[45] Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. 255–269.

[46] Darsh Shah, Raghav Gupta, Amir Fayazi, and Dilek Hakkani-Tur. 2019. Robust zero-shot cross-domain slot filling with example values. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.5484–5490.

[47] Yongliang Shen, Xinyin Ma, Zeqi Tan, Shuai Zhang, Wen Wang, and Weiming Lu. 2021. Locate and label: A two-stage identifier for nested named entity recognition. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*. 2782–2794.

[48] Liangchen Song, Anpei Chen, Zhong Li, Zhang Chen, Lele Chen, Junsong Yuan, Yi Xu, and Andreas Geiger. 2023. NeRFPlayer: A streamable dynamic scene representation with decomposed neural radiance fields. *IEEE Transactions on Visualization and Computer Graphics* 29, 5 (2023), 2732–2742.

[49] Chuanqi Tan, Wei Qiu, Mosha Chen, Rui Wang, and Fei Huang. 2020. Boundary enhanced neural span classification for nested named entity recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 9016–9023.

[50] Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS)*. 1–10.

[51] Jing Wang, Mayank Kulkarni, and Daniel Preotiuc-Pietro. 2020. Multi-domain named entity recognition with genre-aware and agnostic inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 8476–8488.

[52] Yaqing Wang, Haoda Chu, Chao Zhang, and Jing Gao. 2021. Learning from language description: Low-shot named entity recognition via decomposed framework. In *Findings of the Association for Computational Linguistics: EMNLP 2021*. Association for Computational Linguistics, 1618–1630.

[53] Zhenghui Wang, Yanru Qu, Liheng Chen, Jian Shen, Weinan Zhang, Shaodian Zhang, Yimei Gao, Gen Gu, Ken Chen, and Yong Yu. 2018. Label-aware double transfer learning for cross-specialty medical named entity recognition. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. 1–15.

[54] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. 38–45.

[55] Xingxing Xie, Gong Cheng, Jiabao Wang, Xiwen Yao, and Junwei Han. 2021. Oriented R-CNN for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 3520–3529.

[56] 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)*. 6442–6454.

[57] Hang Yan, Tao Gui, Junqi Dai, Qipeng Guo, Zheng Zhang, and Xipeng Qiu. 2021. A unified generative framework for various NER subtasks. 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)*. 5808–5822.

[58] Yang Yan, Junda Ye, Zhongbao Zhang, and Liwen Wang. 2022. AISFG: Abundant information slot filling generator. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 4180–4187.

[59] Huiyun Yang, Shujian Huang, Xin-Yu Dai, and Jiajun Chen. 2019. Fine-grained knowledge fusion for sequence labeling domain adaptation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 4197–4206.

[60] Yi Yang and Arzoo Katiyar. 2020. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 6365–6375.

[61] Zhilin Yang, Ruslan Salakhutdinov, and William W. Cohen. 2017. Transfer learning for sequence tagging with hierarchical recurrent networks. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*.

[62] Bowen Yu, Zhenyu Zhang, Xiaobo Shu, Yubin Wang, Tingwen Liu, Bin Wang, and Sujian Li. 2020. Joint extraction of entities and relations based on a novel decomposition strategy. In *Proceedings of the 24th European Conference on Artificial Intelligence (ECAI)*.

[63] Tao Zhang, Congying Xia, Philip S. Yu, Zhiwei Liu, and Shu Zhao. 2021. PDALN: Progressive domain adaptation over a pre-trained model for low-resource cross-domain named entity recognition. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 5441–5451.

[64] Xinghua Zhang, Bowen Yu, Tingwen Liu, Zhenyu Zhang, Jiawei Sheng, Xue Mengge, and Hongbo Xu. 2021. Improving distantly-supervised named entity recognition with self-collaborative denoising learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 10746–10757.

[65] Xinghua Zhang, Bowen Yu, Yubin Wang, Tingwen Liu, Taoyu Su, and Hongbo Xu. 2022. Exploring modular task decomposition in cross-domain named entity recognition. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. ACM, 301–311.

[66] Junhao Zheng, Haibin Chen, and Qianli Ma. 2022. Cross-domain named entity recognition via graph matching. In *Findings of the Association for Computational Linguistics: ACL 2002*. Association for Computational Linguistics, 2670–2680.

[67] Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, and Dacheng Tao. 2023. Can ChatGPT understand too? A comparative study on ChatGPT and fine-tuned BERT. *arXiv preprint arXiv:2302.10198* (2023).

[68] Joey Tianyi Zhou, Hao Zhang, Di Jin, Hongyuan Zhu, and et al. 2019. Dual adversarial neural transfer for low-resource named entity recognition. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 3461–3471.

[69] Morteza Ziyadi, Yuting Sun, Abhishek Goswami, Jade Huang, and Weizhu Chen. 2020. Example-based named entity recognition. *arXiv:2008.10570* (2020).

Received 24 May 2023; revised 4 December 2023; accepted 11 March 2024