# UDAA: An Unsupervised Domain Adaptation Adversarial Learning Framework for Zero-Resource Cross-Domain Named Entity Recognition

**Baofeng Li[2], Jianguo Tang[1,†], Yu Qin[2], Yuelou Xu[2], Yan Lu[2], Kai Wang[1] Lei Li[1,*], Yanquan Zhou[1,*]**
[1]Beijing University of Posts and Telecommunications
[2]China Electric Power Research Institute Company Limited
{jianguotang, wk, leili} @bupt.edu.cn
libaofeng@epri.sgcc.com.cn

## Abstract

The zero-resource cross-domain named entity recognition (NER) task aims to perform NER in a specific domain where labeled data is unavailable. Existing methods primarily focus on transferring NER knowledge from high-resource to zero-resource domains. However, the challenge lies in effectively transferring NER knowledge between domains due to the inherent differences in entity structures across domains. To tackle this challenge, we propose an Unsupervised Domain Adaptation Adversarial (**UDAA**) framework, which combines the masked language model auxiliary task with the domain adaptive adversarial network to mitigate inter-domain differences and efficiently facilitate knowledge transfer. Experimental results on CBS, Twitter, and WNUT2016 three datasets demonstrate the effectiveness of our framework. Notably, we achieved new state-of-the-art performance on the three datasets. Our code will be released.

## 1 Introduction

Named entity recognition (NER) is a fundamental task in natural language processing. However, in real-world scenarios, obtaining the amount of domain-specific labeled data for the NER task is often challenging, as it can be expensive and time-consuming (Ma and Hovy, 2016; Lample et al., 2016; Akbik et al., 2018; Winata et al., 2019). Hence, zero-resource cross-domain NER (Jia et al., 2019; Bari et al., 2020; Zhang et al., 2021; Karouzos et al., 2021; Zheng et al., 2022), which addresses the issue of data scarcity by adapting NER models to specific domains without available labeled data, draws more and more attention.

Based on the mainstream methods, recent researches have focused on the transfer of NER knowledge from high-resource to zero-resource domains. These methods typically involve training a source model on labeled data from the source domain to learn domain invariant features and apply the adapting NER models to the target domain. In the process of transferring knowledge to the target domain, the Masked Language Model(MLM) is currently recognized as an unsupervised adaptive auxiliary task with good performance. AdaptBERT (Han and Eisenstein, 2019) fine-tunes the MLM on unlabeled data to learn both task-special knowledge and domain-invariant knowledge. Furthermore, TOF (Zhang et al., 2021) addresses the zero-resource cross-domain task scenario through MLM capturing enough NER task knowledge. The acquired knowledge serves as a bridge to facilitate the transfer of NER knowledge.

MLM plays a significant role in extracting context and recognizing entity boundaries. However, during knowledge transfer, especially when mapping between two domains, the transfer of knowledge from different tasks and the varying data distributions across domains can introduce noise and hinder model generalization performance. For instance: a) I'm watching a documentary about a **shark** on TV, and b) The stock market experienced a **shark** attack as prices plummeted. In the social media domain, the word **shark** represents sea animals. However, in the news domain, this is used metaphorically, referring to the stock market crash. The word "shark" shares the same affixes and root knowledge across different domains, yet it carries different semantic meanings in each domain. Meanwhile, due to such

[†]Indicates co-first author.
[*]Corresponding authors.

Proceedings of the 23rd China National Conference on Computational Linguistics, pages 1123-1135, Taiyuan, China, July 25 - 28, 2024.
Volume 1: Main Conference Papers
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

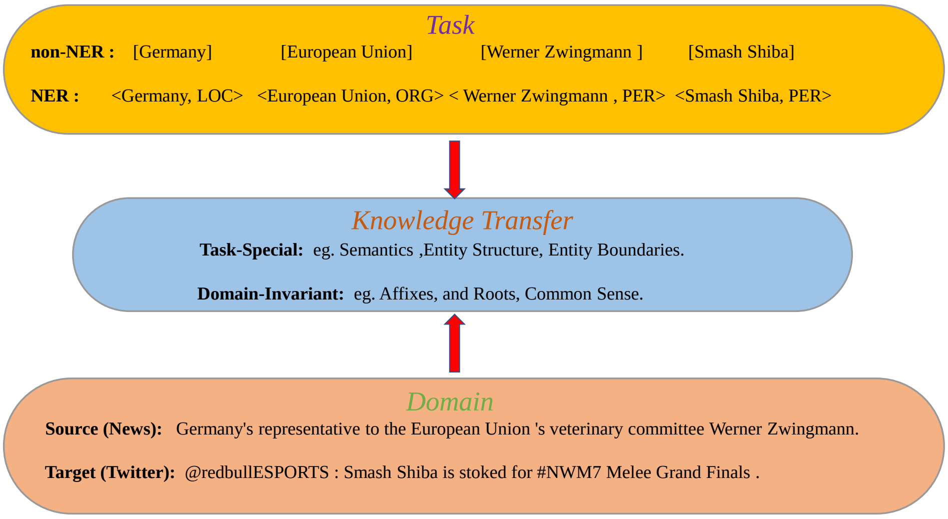
Task
non-NER : [Germany] [European Union] [Werner Zwingmann ] [Smash Shiba]
NER : <Germany, LOC> <European Union, ORG> < Werner Zwingmann , PER> <Smash Shiba, PER>
Knowledge Transfer
Task-Special: eg. Semantics ,Entity Structure, Entity Boundaries.
Domain-Invariant: eg. Affixes, and Roots, Common Sense.
Domain
Source (News): Germany's representative to the European Union 's veterinary committee Werner Zwingmann.
Target (Twitter): @redbullESPORTS : Smash Shiba is stoked for #NWM7 Melee Grand Finals .

Figure 1: The example of knowledge transfer process from two perspectives: domain and task.

semantic biases across different domains, the MLM task may fail to adapt to the specific context and task requirements of the target domain, resulting in a decrease in model performance in the target domain.

Inspired by the idea of (Bari et al., 2020), which adopted the adversarial approach to learning language-invariant knowledge in an unsupervised adaptive way, we propose a framework named Unsupervised Domain Adaptation Adversarial (UDAA). UDAA combines the MLM auxiliary task with the Domain Adaptation Adversarial Network (DAAN) to mitigate inter-domain disparities. By incorporating an adversarial loss, the model becomes more adept at distinguishing domain-invariant knowledge from task-specific knowledge. This encourages the development of resource-agnostic representations, allowing knowledge from high-resource domains to be more effectively aligned with zero-resource domains.

As shown in Figure 1, in our work, the knowledge transfer process is from two perspectives: domain and task.

**Domain.** From the domain perspective, the domain-invariant knowledge can be transferred, such as affixes, roots, and common sense, in both source and target domains. To ensure the consistency between feature representations extracted from the source domain and those from the target domain, we employ the DAAN method to align features across different domains, eliminating distribution discrepancies between domains. This approach captures domain-relevant knowledge, allowing for the learning of more domain-invariant features to ensure the model's generalization performance.

**Task.** From the task perspective, the task-special knowledge can be transferred, such as entity structure and senmantics, in both NER and non-NER task. These task-specific features play a crucial role in the complementary process of NER and non-NER task(Rei, 2017; Peters et al., 2017). For instance, by introducing the MLM task, the model can extract entity and semantic information from the context and identify entity boundaries, thus compensating for missing entity information in domain-specific NER task. In the UDAA model, tasks in different stages act as bridges from high-resource domains to zero-resource domains, facilitating effective knowledge transfer.

In this paper, the UDAA framework is designed to address the challenge of zero-resource cross-domain NER task, which focuses on ensuring the adaptability of unsupervised domains by combining the MLM auxiliary task with DAAN. This integration enables the framework to transfer knowledge from both domain and task perspectives. With enough training on the labeled data from the source domain, the framework generates pseudo-data on the unlabeled corpus to address the data scarcity challenge encountered in the target domain. The training process of the UDAA framework consists of three unified steps: Adversarial Domain Learning (ADL), Adaptive Pseudo-Data Generation(APG), and Domain Adaptive

Prediction(DAP).

To validate the effectiveness and superiority of our proposed approach, we conduct experiments on three datasets for zero-resource cross-domain NER task: CBS (Jia et al., 2019), Twitter (Zhang et al., 2018b), and WNUT2016 (Strauss et al., 2016).

**Our contributions can be summarized as follows:**

- We propose the UDAA framework, which performs the zero-resource cross-domain NER task by unifying training through the steps of Adversarial Domain Learning, Adaptive Pseudo-Data Generation, and Domain Adaptive Prediction.
- We combine the MLM auxiliary task with the DAAN to mitigate inter-domain disparities. By incorporating an adversarial loss, the model becomes more adept at distinguishing task-special knowledge from domain-invariant knowledge. This integration provides a lightweight structure for the UDAA framework.
- Experimental results on the CBS, Twitter, and WNUT2016 datasets demonstrate the effectiveness of our UDAA framework. Meanwhile, we achieve new state-of-the-art performance on the three datasets.

## 2 Related Work

**Cross-domain NER.** Some studies (Devlin et al., 2018; Sun et al., 2020; Liu et al., 2021) achieves remarkable performance in cross-domain NER scenario, yet they require many domain-related labeled data for training when adapting to the target domain. To this end, cross-domain NER algorithms (Yang et al., 2018; Lin and Lu, 2018; Liu et al., 2020b) that alleviate the data scarcity issue and boost the models' generalization ability to target domain have drawn substantial attention recently. To solve this problem, some studies introduce multitask learning (Liu et al., 2020b; Wang et al., 2020) or designing new model architectures (Jia et al., 2019; Liu et al., 2020c; Jia and Zhang, 2020) for improving the NER performance of the target domain by training on data from both source and target domain. However, these methods typically require training with vast amounts of labeled source domain data to achieve satisfactory performance in the target domain. Our work differs in that we neither requires a large amount of domain-related data nor the design of a new model structure. We design a unified training framework and use models of different tasks as the bridge from high-resource to zero-resource domain to achieve knowledge transfer.

**Zero-resource NER.** Some studies (Jia and Zhang, 2020; Pfeiffer et al., 2020) focus on enhancing the architectural design of existing models by incorporating new components to capture specific knowledge, such as entity types and task characteristics. Different from these methods, our approach only modifies the training procedure without changing model structures. Other studies introduce different auxiliary tasks to alleviate data scarcity (Han and Eisenstein, 2019; Phang et al., 2020; Xue et al., 2020a). They are usually based on multi-task learning. Multi-task learning requires balance between the NER task and auxiliary tasks, which needs carefully designed objectives. In addition, LLMs (Ouyang et al., 2022; Zeng et al., 2022a) have good learning and expression capabilities, so they are also used in zero-resource NER task and have achieved some good results. Notably, our work differs in that we design an unsupervised domain adaptation fine-tuning framework to exploit more diverse data and training strategies.

**Adversarial Learning.** Adversarial learning originates from Generative Adversarial Nets(GAN) (Goodfellow et al., 2014). Adversarial learning is a regularization method for improving the generalization of a classifier. It does so by improving model's robustness to adversarial examples, which are created by making small perturbations to the input. Recently, many studies (Bari et al., 2020; Zhao et al., 2022) have tried to apply adversarial learning to NLP tasks. They(Liu et al., 2017) extended adversarial training to the multi-task learning framework for text classification, aiming to alleviate the domain-invariant (shared) and task-special (private) latent feature spaces from interfering with each other. Notably, Our method integrates both the adversarial training in an unified framework to find domain-invariant information for MLM tasks.

## 3 Our Approach

In recent years, researchers have paid more attention to cross-language scenarios, as the data in cross-domain scenarios is more difficult to obtain. Therefore, we adopt TOF framework (Zhang et al., 2021) as our baseline, which was designed for unsupervised domain adaptation in sequence labeling tasks and achieved state-of-the-art performance in previous work.

The UDAA framework is enlightened by the TOF. Our approach consists of the knowledge transfer process and the fine-tuning process. The architecture of UDAA is illustrated in Figure 2.

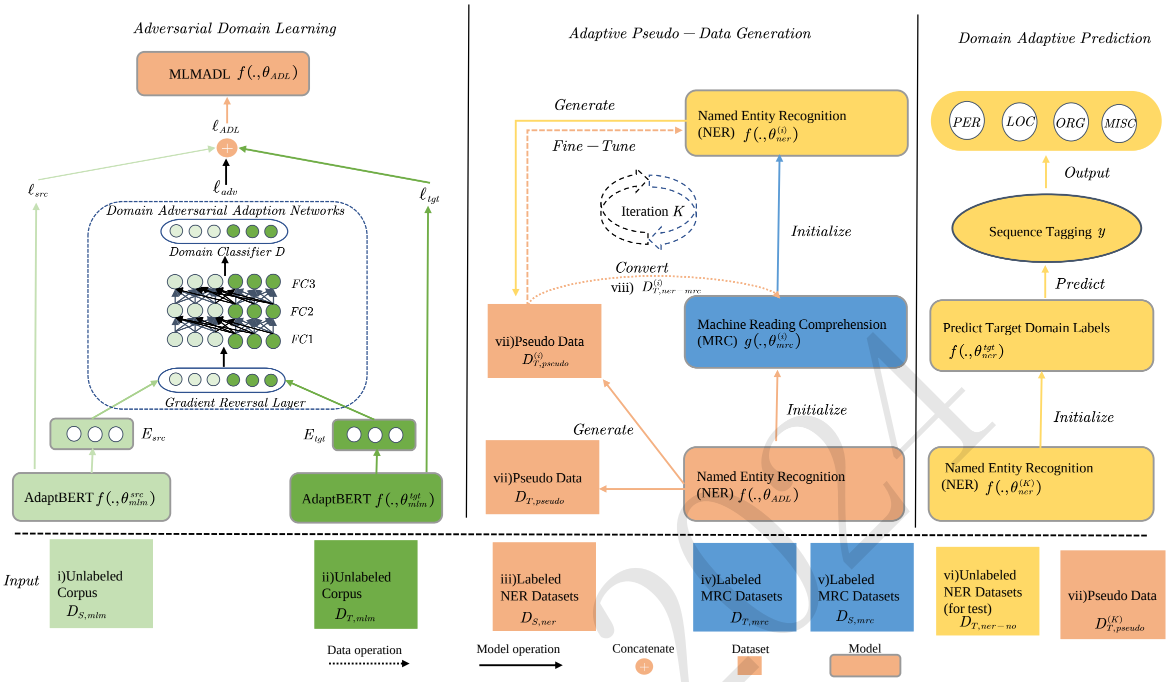
Adversarial Domain Learning
MLMADL $f(.,\theta_{ADL})$
$\ell_{ADL}$
$\ell_{src}$
$\ell_{adv}$
$\ell_{tgt}$
Domain Adversarial Adaption Networks
Domain Classifier D
FC3
FC2
FC1
Gradient Reversal Layer
$E_{src}$
$E_{tgt}$
AdaptBERT $f(.,\theta_{mlm}^{src})$
AdaptBERT $f(.,\theta_{mlm}^{tgt})$
Adaptive Pseudo − Data Generation
Generate
Fine − Tune
Named Entity Recognition (NER) $f(.,\theta_{ner}^{(i)})$
Iteration $K$
Initialize
Convert
viii) $D_{T,ner-mrc}^{(i)}$
vii)Pseudo Data $D_{T,pseudo}^{(i)}$
Machine Reading Comprehension (MRC) $g(.,\theta_{mrc}^{(i)})$
Initialize
Generate
vii)Pseudo Data $D_{T,pseudo}$
Named Entity Recognition (NER) $f(.,\theta_{ADL})$
Domain Adaptive Prediction
PER
LOC
ORG
MISC
Output
Sequence Tagging $y$
Predict
Predict Target Domain Labels $f(.,\theta_{ner}^{tgt})$
Initialize
Named Entity Recognition (NER) $f(.,\theta_{ner}^{(K)})$
Input
i)Unlabeled Corpus $D_{S,mlm}$
ii)Unlabeled Corpus $D_{T,mlm}$
iii)Labeled NER Datasets $D_{S,ner}$
iv)Labeled MRC Datasets $D_{T,mrc}$
v)Labeled MRC Datasets $D_{S,mrc}$
vi)Unlabeled NER Datasets (for test) $D_{T,ner-no}$
vii)Pseudo Data $D_{T,pseudo}^{(K)}$
Data operation
Model operation
Concatenate
Dataset
Model

Figure 2: The overall architecture of the UDAA framework. The dotted arrows represent the knowledge transfer process of the data, and the solid arrows represent the fine-tuning process of the model. At the ADL stage, we combine the AdaptBERT model with the DAAN and incorporating the adversarial loss to distinguish domain-invariant knowledge from task-specific knowledge. At this time, an MLMADL model will be generated as the input of the next APG stage. In the APG stage, the MRC framework (Li et al., 2019) will be used for $K$ rounds of iterations to generate high-quality pseudo-data and the best NER model for prediction in the next DAP stage. The models for tasks at different stages serve as bridges from high-resource domains to zero-resource domains and enable knowledge transfer.

### 3.1 Knowledge Transfer Process

As shown in Figure 2, our framework aims to facilitate knowledge transfer to the target domain, which displays how to transfer not only domain but also task knowledge from various data. Relevant information about the data is listed in Table 1.

We consider six kinds of datasets: i) Unlabeled Corpus $D_{S,mlm}$, ii) Unlabeled Corpus $D_{T,mlm}$, iii) Labeled NER Dataset $D_{S,ner}$, iv) Labeled MRC Dataset $D_{T,mrc}$, v) Labeled MRC Dataset $D_{S,mrc}$, and vi) Unlabeled NER Dataset (for test) $D_{T,ner-no}$, where datasets { i), iii), v)} belong to the source domain and datasets { ii), iv), vi)} belong to the target domain. Additionally, two important intermediate generated data are also included: vii) Pseudo Data $D_{T,pseudo}$, and viii) MRC-Style Data $D_{T,ner-mrc}$, where $D_{T,ner-mrc}$ is converted from $D_{T,pseudo}$ and plays an important role in the knowledge transfer of MRC intermediate task.

Considering the discrepancy between the source and target domains, it is essential to perform fine-tuning tasks on both the source and target data. In addition to the labeled data for the NER task, we also

| Number | Name | Domain | Task |
|---|---|---|---|
| i) | $D_{S,mlm}$ | Source | MLM |
| ii) | $D_{T,mlm}$ | Target | MLM |
| iii) | $D_{S,ner}$ | Source | NER |
| iv) | $D_{T,mrc}$ | Target | MRC |
| v) | $D_{S,mrc}$ | Source | MRC |
| vi) | $D_{T,ner-no}$ | Target | NER |
| vii) | $D_{T,pseudo}$ | Target | NER |
| viii) | $D_{T,ner-mrc}$ | Target | MRC |

Table 1: Various data descriptions in the knowledge transfer process.

utilize the non-NER task. For example, $D_{S,mlm}$ and $D_{T,mlm}$ are used for ADL stage, while $D_{S,mrc}$ and $D_{T,mrc}$ are used for APG stage.

## 3.2 Fine-tuning Process

There are three steps to performing the NER task in the fine-tuning process: a) Adversarial Domain Learning, b) Adaptive Pseudo-Data Generation, and c) Domain Adaptive Prediction.

### 3.2.1 Adversarial Domain Learning

We employ two independent encoders to fine-tune the MLM auxiliary task to capture domain-invariant and task-special representations for source and target domains. Concretely, given two input token sequences $x_s = \{x_i\}_i^N$ and $x_t = \{x_j\}_j^M$, where $N, M$ represents the number of words in source and target domains. We feed it into the feature encoder to obtain contextualized word embeddings $E_{src}$ and $E_{tgt}$:

$$E_{src} = AdaptBERT(x_s) \tag{1}$$

$$E_{tgt} = AdaptBERT(x_t) \tag{2}$$

However, the MLM task may fail to adapt to the specific context and task requirements of the target domain due to semantic biases present across different domains, resulting in a decrease in model performance in the target domain. To address this issue, we integrate DAAN with the MLM task and conduct joint training to mitigate semantic biases between domains and facilitate effective knowledge transfer.

In our work, we use adversarial training to find domain invariant representations . The DAAN exhibit an architecture whose first few feature extraction layers are shared by two classifiers trained simultaneously. DAAN minimizes the domain classification loss with respect to parameters specific to the domain classifier $D$ , while maximizing it with respect to the parameters that are common to both classifiers. This minimax optimization becomes possible via the use of a gradient reversal layer (GRL). By incorporating DAAN, a more powerful model MLMADL is trained, which can capture more domain-invariant knowledge, thereby enhancing the model's generalization performance.

$$L_i(x_d) = W_i \cdot x_d + b_i \tag{3}$$

$$D(E_d) = \sigma([L_2 \cdot (L_1(E_d) \cdot W_3 + b_3)]) \tag{4}$$

where $d \in \{src, tgt\}$ and $L_i$ correspond to the fully connected layer in DAAN. $D(E_d)$ discriminates whether $E$ comes from the source or the target domains. $W_i$ and $b_i$ represent different fully connected layer parameters. The loss function $\ell_{adv}$ for DAAN is calculated as follows:

$$\ell_{adv} = \Sigma_{x_s} log(D(E_{src})) + \Sigma_{x_t} log(1 - D(E_{tgt})) \tag{5}$$

The total loss function $\ell_{ADL}$ for jointly train in the ADL stage is defined as follows:

$$\ell_{ADL} = \ell_{adv} + \ell_{src} + \ell_{tgt} \tag{6}$$

where $\ell_{src}$ represents the source domain loss function and $\ell_{tgt}$ represents the target domain loss function.

By incorporating $\ell_{adv}$, the MLMADL model becomes more adept at distinguishing domain-invariant knowledge from task-specific knowledge. This encourages the development of resource-agnostic representations, allowing knowledge from high-resource domains to be more effectively aligned with zero-resource domains.

### 3.2.2 Adaptive Pseudo-Data Generation

We obtain a trained NER model by using labeled data from the source domain along with the initial parameters of the MLMADL, which is utilized on the unlabeled corpus in the target domain and generates pseudo-data. To iteratively optimize this pseudo-data, we employ the MRC auxiliary task. The MRC task offers several advantages: i) It enhances the NER model's capability to extract spans, thereby capturing semantic information associated with different entity types (Wu et al., 2019). ii) It acts as a bridge between NER and other tasks. iii) It facilitates the transfer of knowledge from diverse tasks into a unified training framework (Liu et al., 2020a).

In the iterative optimization process of the MRC task, the dataset format required differs from that of the NER task dataset. Therefore, dataset conversion is necessary. The dataset is represented using triples $(question, answer, context)$, and to fully leverage contextual information from each sentence, we treat the description of each entity as a query. Each entity in the dataset corresponds to an entity label, and each entity label is associated with a query $q_y = q_1, q_2, ..., q_m$, where $m$ represents the length of the generated query. The training objective of the MRC task at this point is to answer natural language queries given the context to obtain named entity information.

In the MRC task, the design of natural language queries plays a crucial role because it requires skillful construction of queries to encode prior knowledge about entity class labels. Otherwise, named entity information in the text cannot be effectively obtained. We adopt a template-based approach to construct natural language queries for relevant entity type labels.

We employ a span selection strategy (Li et al., 2019) by feeding the context word vectors of each label into two separate linear classification layers. Using binary classifiers to predict the start and end indices of each entity allows the model to determine the boundaries of the entity based on specific queries. The probability formula for computing the $start$ or $end$ indices of each label as entity spans is as follows:

$$p^{start} = softmax(W_{start} \cdot x) \tag{7}$$

$$p^{end} = softmax(W_{end} \cdot x) \tag{8}$$

where $W_{start}$ and $W_{end} \in R^{d_1 \times 2}$ are learnable parameters, and $d_1$ denotes the dimensions of contextualized word embedding.

After $K$ rounds of iterative optimization using the MRC task, we obtain high-quality pseudo-data, which helps alleviate the issue of data scarcity in the target domain. In addition, through the MRC intermediate task, we also obtained the NER model for prediction in the next DAP stage.

### 3.2.3 Domain Adaptive Prediction

In the DAP stage, the NER model inherits parameters from the APG stage and applies it to predict sequence labels in the target domain. During the DAP stage, given the text $w = (w_1, w_2, ...w_T)$ of the dataset $D_{T,ner-no}$, special characters $[CLS]$ and $[SEP]$ are respectively appended at the beginning and end of each sentence to obtain special representations. After encoding by the model $f(\cdot, \theta^{tgt}_{ner})$, a series of contextual feature representations $x$ are returned. These representations $x$ are then inputted into linear classification layers, maximizing the probability of each label $y_t$ using true entity labels. We opt for the cross-entropy loss function, and the probability calculation for entity prediction is as follows:

$$logp(y_t|w_{1:T}) = \beta_{y_t} \cdot x - logy \in Y\Sigma exp(\beta_{y_t} \cdot x) \tag{9}$$

where the contextualized embedding $x$ captures information from the entire sequence $w_{1:T} = (w_1, w_2, ..., w_T)$, and $\beta_{y_t}$ is a vector of weights for each tag $y \in \{PER, LOC, ORG, MISC\}$.

In the DAP stage, the UDAA model utilizes high-quality pseudo-labeled data refined through $K$ rounds of iterative optimization, along with a well-generalized NER model, to make predictions. This process enables the NER task model to effectively adapt to the target domain and achieve domain-adaptive sequence labeling predictions.

## 4 Experiments

### 4.1 Datasets

We take Conll-2003 for English in the news domain as our source domain dataset for the NER task. As for the target domain, we consider three datasets: a) CBS SciTech News(referred to as CBS) (Jia et al., 2019), from the technology news domain, b) Twitter (Zhang et al., 2018b), and c) WNUT2016 (Strauss et al., 2016) both from the social media domain. The unlabeled corpora from these datasets are readily available for the MLM task. In addition, for the iterative optimization process in the MRC task, we select NewsQA (Trischler et al., 2016) and TweetQA (Xiong et al., 2019) as our target domain dataset, while SQuAD (Rajpurkar et al., 2016) is chosen as the source domain dataset.

To ensure one-to-one correspondence between labels across different domains, we chose the “no-type” category from the WNUT2016 dataset, where entities don’t have specific label types and only contain entity types in the BIO format. Additionally, the labels in the CBS and Twitter datasets cover four entity types: PER, LOC, ORG, and MISC. To ensure transferability and adaptability to the target domain, we applied a data preprocessing approach to remove entity label data from the training set in the CBS and Twitter datasets. All datasets in the NER task adhere to the Conll-2003 dataset standard, where data is divided into two columns: the first column contains word segmentation in the input sentence, and the second column contains named entity type labels corresponding to each word.

### 4.2 Hyperparameters

| Module | Task | CBS | Twitter | WNUT2016 |
|---|---|---|---|---|
| ADL | MLM | 2e-5 | 2e-5 | 2e-5 |
| ADL | NER | 5e-5 | 2e-5 | 5e-5 |
| APG | NER | 8e-5 | 1e-6 | 1e-6 |
| APG | MRC | 8e-6 | 3e-5 | 1e-6 |
| DAP | NER | 3e-5 | 5e-6 | 2e-6 |

Table 2: Learning rate for UDAA framework.

In the training procedure, we choose the Adam (Zhang et al., 2018a) optimizer and our experiment runs on the Tesla P100 server, which takes more than 14h. We conduct the hyperparameters search to determine the appropriate learning rate for the $ADL$ module. We explore a range of learning rate values: 5e-6, 8e-6, 1e-5, 2e-5, 3e-5, 5e-5, and 8e-5. We also explore a range of batch size values: 16, 32, and 64, which are limited by server computing power and we choose the batch size of 32. The selection of hyperparameters values is based on the best validation performance.

Furthermore, we fine-tune each MLM model for 3 epochs, MRC for 6 epochs, and NER for 6 epochs. Additionally, we perform one iteration of APG stage and achieve the best performance. The learning rates for the UDAA framework can be found in Table 2. Other hyperparameters are set following the approach described by TOF(Zhang et al., 2021).

### 4.3 Model

We evaluate the following models by P, R, and F1 scores:

- **ChatGPT**(Ouyang et al., 2022) select the GPT-3.5-Turbo model for zero-resource named entity prediction.
- **ChatGLM**(Zeng et al., 2022a) select the ChatGLM2-6B model for zero-resource named entity prediction.

| Model | CBS | | | Twitter | | | WNUT2016 | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F1 | P | R | F1 | P | R | F1 |
| ChatGPT†(Ouyang et al., 2022) | 28.23 | 25.63 | 26.87 | 17.64 | 19.92 | 18.71 | 15.85 | 18.05 | 16.57 |
| ChatGLM†(Zeng et al., 2022a) | 1.74 | 2.11 | 1.91 | 1.58 | 1.88 | 1.71 | 1.29 | 1.33 | 1.31 |
| Cross-LM BiLSTM(Jia et al., 2019) | 68.48 | 79.52 | 73.59 | - | - | - | - | - | - |
| mCell LSTM(Jia and Zhang, 2020) | - | - | 75.19 | - | - | - | - | - | - |
| COFEE-MRC(Xue et al., 2020b) | - | - | - | - | - | 54.56 | - | - | - |
| TOF(Zhang et al., 2021) | - | - | 76.41 | - | - | 67.94 | - | - | 67.86 |
| BiLSTM+CRF*(Liu et al., 2020b) | 78.10 | 59.94 | 67.82 | 65.72 | 58.89 | 62.12 | / | / | / |
| BiLSTM+CRF+DAAN* | 78.48 | 61.29 | 68.82 | 65.88 | 60.74 | 63.21 | / | / | / |
| AdaptBERT*(Han and Eisenstein, 2019) | 69.80 | 81.26 | 75.10 | 66.49 | 66.62 | 66.55 | 65.66 | 62.30 | 64.10 |
| TOF*(Zhang et al., 2021) | 71.11 | 82.58 | 76.42 | 68.20 | 67.61 | 67.90 | 70.56 | 62.74 | 66.42 |
| UDAA($ours$) | 71.76 | 82.23 | **76.64** | 68.94 | 68.67 | **68.81** | 71.17 | 65.53 | **68.24** |
| $w/o$ APG Stage | 71.39 | 81.97 | 76.32 | 68.03 | 67.70 | 67.86 | 69.01 | 65.39 | 67.15 |
| $w/o$ ADL Stage | 71.42 | 81.13 | 75.97 | 66.38 | 67.01 | 66.70 | 67.67 | 64.73 | 66.17 |
| $w/o$ APG Stage & $w/o$ ADL Stage | 68.83 | 79.28 | 73.69 | 65.85 | 65.10 | 65.48 | 61.44 | 63.32 | 62.37 |

Table 3: Results of our UDAA framework compared to previous state-of-the-art methods for the zero-resource cross-domain NER task. "†" indicates the prediction results of the LLMs. "-" indicates results not provided in prior methods. "*" signifies results re-implemented by us. "/" denotes cases where the processed WNUT2016 dataset is incompatible with the BiLSTM model and therefore results are unavailable. Bold text highlights the best experimental outcomes.

| Model | CBS | | | Twitter | | | WNUT2016 | | |
|---|---|---|---|---|---|---|---|---|---|
| | Time | Flops | Size | Time | Flops | Size | Time | Flops | Size |
| **TOF Framework** | | | | | | | | | |
| MRC Enhancing | 19.55h | 3656.9B | 312M | 13.53h | 2419.0B | 312M | 13.56h | 1397.8B | 312M |
| Pseudo Data Enhancing | 0.54h | 347.9B | 103.3M | 0.15h | 337.1B | 103.3M | 0.11h | 282.7B | 103.3M |
| Continual Learning Enhancing | 12.24h | 3086.3B | 206.6M | 12.35h | 2439.5B | 206.6M | 12.01h | 1432.9B | 206.6M |
| **Total** | 31.84h | 7091.1B | 624M | 25.50h | 4858.5B | 624M | 25.6h | 3113.4B | 624M |
| **UDAA Framework** | | | | | | | | | |
| Adversarial Domain Learning | 12.60h | 1043.7B | 103.3M | 1.17h | 872.6B | 103.3M | 1.28h | 726.1B | 103.3M |
| Adaptive Pseudo-Data Generation | 12.24h | 3086.3B | 206.6M | 12.35h | 2439.5B | 206.6M | 12.01h | 1432.9B | 206.6M |
| Domain Adaptive Prediction | 0.54h | 347.9B | 103.3M | 0.15h | 337.1B | 103.3M | 0.11h | 282.7B | 103.3M |
| **Total** | 25.33h | 4477.9B | 415.3M | 13.69h | 3649.2B | 415.3M | 13.71h | 2441.6B | 415.3M |

Table 4: Comparison of UDAA and TOF frameworks in computing resources. The MRC Enhancing, Pseudo Data Enhancing, and Continual Learning Enhancing are mentioned in TOF(Zhang et al., 2021).

- **Cross-LM BiLSTM**(Jia et al., 2019) using a language model as a cross-domain bridge, automatically generate training parameters for the LSTM model to perform zero-resource named entity prediction.

- **mCell LSTM** (Jia and Zhang, 2020) designed a multicell compositional LSTM for cross-domain NER task.

- **COFEE-MRC**(Xue et al., 2020b) designed a neural network model with MRC auxiliary task.

- **BiLSTM+CRF**(Liu et al., 2020b) designed a BiLSTM and CRF paradigm for zero-resource cross-domain NER.

- **BiLSTM+CRF+DAAN.** We built on the work(Liu et al., 2020b), incorporating the DAAN components.

- **AdaptBERT.**(Han and Eisenstein, 2019) adopted the domain-tuning and task-tuning in the cross-domain NER task.

- **TOF**(Zhang et al., 2021) performed MLM and MRC auxiliary tasks. It is the previous best state-of-the-art and we take the TOF as our baseline in the zero-resource cross-domain scenario.

- **UDAA** The framework proposed in our work and details can be found in section 3.

## 5 Results and Analysis

### 5.1 Framework Performance

Table 3 presents the main results of our framework compared to previous methods in the zero-resource cross-domain NER task. We regard re-implemented results of TOF (Zhang et al., 2021) as our baseline. Our framework yields obvious improvements over the baseline (CBS: $0.22$ ↑, Twitter: $0.91$ ↑, and WNUT2016: $1.82$ ↑ ) and achieves new state-of-the-art results on the three datasets.

The two selected LLMs, ChatGPT and ChatGLM, performed poorly. The reasons may be: (a) The instructions used by the LLMs require the model to generate more fine-grained entity tags. In the zero-resource scenario, this more complex instruction design makes it difficult for the models to ensure the accuracy of the output; (b) LLMs are prone to missing entities, mislabeling, and multiple entities; (c) The selected LLMs is limited by computing resources and is not fine-tuning for NER task.

In conclusion, all these results verify the effectiveness and transferability of our framework UDAA in the zero-resource cross-domain NER task.

### 5.2 Effectiveness of DAAN

As shown in Table 3, the integration of DAAN into the BiLSTM model leads to improved F1 scores in the experimental results on the CBS($1.0$ ↑) and Twitter($1.09$ ↑) datasets, which shows the effectiveness of DAAN in enhancing feature extractors. It also indicates that DAAN enables the model to acquire additional entity information and gain the better understanding of the target domain data.

In addition, our model builds upon the baseline of TOF, and we observed enhancements in the F1 score by introducing DAAN while omitting the MRC task during the fine-tuning process. Moreover, DAAN provides a lightweight and effective method, which also accelerates the model's training speed, consequently reducing training time and computational resource costs. We compare the computing resource consumption of the UDAA with TOF, measured by Time(time consumption, $h$ means hour), Flops(floating point operations per second, $B$ means billion), and Size(model parameters). The results shown in Table 4 demonstrate that UDAA consumes fewer computing resources than TOF.

| Method | CBS | Twitter | WNUT2016 |
|---|---|---|---|
| with DAAN | 75.97 | 66.70 | 66.17 |
| without DAAN | 75.66 | 66.04 | 65.14 |

Table 5: Compared the experimental results of NER task with and without the integration of DAAN.

Finally, as shown in Table 5, the integration of DAAN into the MLM task yielded better experimental results on the CBS($0.31$ ↑), Twitter($0.66$ ↑) and WNUT2016($1.03$ ↑) datasets. This further underscores the role of the introduced adversarial loss helps the MLM task distinguish between domain-invariant features and task-specific features, thereby enhancing the model's robustness and generalization performance.

Overall, whether we consider the notable reduction in computational resources as demonstrated in Table 4 or the enhancements in the experimental results as evident in Table 3 and Table 5, it underscores the effectiveness of DAAN. The integration of this structure has proven to be beneficial for the framework.

### 5.3 Ablation Analysis

#### 5.3.1 Component Impact

To assess the influence of various components of UDAA, we examine the effects of two modules: i) ADL, and ii) APG. As shown in Table 3, when we remove the APG stage (CBS($0.32$ ↓), Twitter($0.95$ ↓) and WNUT2016($1.09$ ↓)), implying the absence of high-quality pseudo-data and the MRC task, the model's performance decline is less pronounced compared to when the ADL stage (CBS($0.67$ ↓), Twitter($2.11$ ↓) and WNUT2016($2.07$ ↓)) is absent. This highlights the substantial contribution of the introduced DAAN to the task model in subsequent stages, thus improving its robustness and generalization performance.

Meanwhile, the experimental impact of removing the APG stage alone or removing the ADL stage alone is far lower than removing both simultaneously, indicating the crucial role of unified joint training between the APG and ADL stages for the UDAA model. This may be attributed to the coupling effect of the model eliminating inter-domain differences in the ADL stage and high-quality pseudo-labeled data in the APG stage during the training process of the UDAA model.

In conclusion, the models for tasks at various stages act as bridge from high-resource domains to zero-resource domains, facilitating knowledge transfer. The removal of any component has an impact on the experimental results, proving their positive influence on the UDAA framework.

### 5.3.2 Task Combination

| Components | CBS | Twitter | WNUT2016 |
|---|---|---|---|
| MLMADL | **76.64** | **68.81** | **68.24** |
| NERADL | 76.59 | 68.13 | 67.62 |
| MLMADL & NERADL | 76.03 | 68.18 | 67.30 |

Table 6: Experimental results on tasks combined with DAAN.

We explore the impact of DAAN when integrated with both NER and non-NER task, represented as follows: i) MLMADL, which combines DAAN with the MLM task, ii) NERADL, which combines DAAN with the NER task, and iii) MLMADL & NERADL, which combines DAAN with the MLM task and NER task. Based on the results presented in Table 6, it is evident that the MLMADL component achieves the best results on the three datasets.

The MLMADL & NERADL strategy yielded the lowest experimental results across the three datasets. This may be attributed to the introduction of noisy data during training due to potential differences between tasks after integrating DAAN into both tasks, leading to a decrease in model performance. Therefore, in the training process of the UDAA model, we opted for the strategy of training MLM combined with DAAN. Furthermore, considering the high computational resource demand of the MRC task and its poor experimental results, we did not conduct experiments to construct MRCADL.

### 5.4 Case Study

| | |
|---|---|
| **CBS** | Astronomer [Kobie Van] with the [Johannesburg Planetarium] in [South Africa] told to [CBS News] foreign correspondent Debora [Patta] the reason for ... |
| TOF<br>UDAA | [ Kobie Van $PER$ ] [ Johannesburg Planetarium $LOC$ ] [CBS News ORG ] [Patta PER ]<br>[ Kobie Van $PER$ ] [ Johannesburg Planetarium $ORG$ ] [CBS News ORG ] [Patta PER ] |
| **Twitter** | Caress your phrase tenderly : it will end by smiling at you. [Anatole] [France] #amwriting #writing http://t.co/48WU4phqiO |
| TOF<br>UDAA | [ Anatole $LOC$ ] [ France $LOC$ ]<br>[ Anatole $PER$ ] [ France $LOC$ ] |
| **WNUT2016** | Where it all begins! An exhilarating visit to source of [Arno River] in Alto #[Casentino] http://t.co/EdZIWjVxGc |
| TOF<br>UDAA | [ Casentino $I$ ]<br>[ Casentino $B$ ] |

Table 7: Cases example of three datasets, where the red and blue represent correct and incorrect entities, respectively.

Table 7 presents case examples from three datasets, exhibit the performance comparison between TOF and UDAA. In the CBS dataset, the words “Kobie Van”, “CBS News”, and “Debora Patta” are correctly identified as entities in both TOF and UDAA. However, due to the influence of the word “South Africa” in the following context, TOF erroneously identifies “Johannesburg Planetarium” as the $LOC$ entity, whereas UDAA can better understand the contextual information and correctly identify it as the $ORG$ entity.

In the Twitter and WNUT2016 datasets from the social media domain, "#" is used to highlight specific topics, which constitutes domain-specific knowledge, posing challenges for entity prediction by TOF. However, UDAA can accurately identify and classify entities. For example, in the Twitter dataset, due to the influence of "#" and"France" in the context, TOF erroneously identifies "Anatole" as the $LOC$ entity, while UDAA correctly identifies it as the $PER$ entity. In summary, the examples in Table 7 visually demonstrate the superiority of the UDAA model in distinguishing domain-specific knowledge and context understanding.

## 6 Conclusion

We propose the UDAA framework, which combines the MLM auxiliary task with the DAAN to mitigate inter-domain differences. The models for tasks at different stages serve as bridges from high-resource domains to zero-resource domains and enable knowledge transfer. Through extensive experimental evaluation using the CBS, Twitter, and WNUT2016 datasets, we demonstrate the effectiveness of UDAA in the zero-resource cross-domain NER task. In the future, we plan to improve the optimization of pseudo-data by prompt learning and explore the task where label mismatch scenarios are involved.

## 7 Acknowledgements

This work is supported by the science and technology project of State Grid Corporation of China (Grant No. 5400-202355230A-1-1-ZN).

## References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *International Conference on Computational Linguistics*.

M Saiful Bari, Shafiq Joty, and Prathyusha Jwalapuram. 2020. Zero-resource cross-lingual named entity recognition. In *Proceedings of the aaai conference on artificial intelligence*, volume 34, pages 7415–7423.

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

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Xiaochuang Han and Jacob Eisenstein. 2019. Unsupervised domain adaptation of contextualized embeddings for sequence labeling. *arXiv preprint arXiv:1904.02817*.

Jinpeng Hu, Yaling Shen, Yang Liu, Xiang Wan, and Tsung-Hui Chang. 2022. Hero-gang neural model for named entity recognition. *arXiv preprint arXiv:2205.07177*.

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*, pages 5906–5917.

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*, pages 2464–2474.

Constantinos Karouzos, Georgios Paraskevopoulos, and Alexandros Potamianos. 2021. Udalm: Unsupervised domain adaptation through language modeling. *arXiv preprint arXiv:2104.07078*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2019. A unified mrc framework for named entity recognition. *arXiv preprint arXiv:1910.11476*.

Bill Yuchen Lin and Wei Lu. 2018. Neural adaptation layers for cross-domain named entity recognition. *arXiv preprint arXiv:1810.06368*.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial multi-task learning for text classification. *arXiv preprint arXiv:1704.05742*.

Jian Liu, Yubo Chen, Kang Liu, Wei Bi, and Xiaojiang Liu. 2020a. Event extraction as machine reading comprehension. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1641–1651.

Zihan Liu, Genta Indra Winata, and Pascale Fung. 2020b. Zero-resource cross-domain named entity recognition. *arXiv preprint arXiv:2002.05923*.

Zihan Liu, Genta Indra Winata, Peng Xu, and Pascale Fung. 2020c. Coach: A coarse-to-fine approach for cross-domain slot filling. *arXiv preprint arXiv:2004.11727*.

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*, volume 35, pages 13452–13460.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Matthew E Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. Semi-supervised sequence tagging with bidirectional language models. *arXiv preprint arXiv:1705.00108*.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, and Luke Zettlemoyer. 2018. Deep contextualized word representations.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. Mad-x: An adapter-based framework for multi-task cross-lingual transfer. *arXiv preprint arXiv:2005.00052*.

Jason Phang, Iacer Calixto, Phu Mon Htut, Yada Pruksachatkun, Haokun Liu, Clara Vania, Katharina Kann, and Samuel R Bowman. 2020. English intermediate-task training improves zero-shot cross-lingual transfer too. *arXiv preprint arXiv:2005.13013*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Marek Rei. 2017. Semi-supervised multitask learning for sequence labeling. *arXiv preprint arXiv:1704.07156*.

Erik F Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*.

Benjamin Strauss, Bethany Toma, Alan Ritter, Marie-Catherine De Marneffe, and Wei Xu. 2016. Results of the wnut16 named entity recognition shared task. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pages 138–144.

Tianxiang Sun, Yunfan Shao, Xiaonan Li, Pengfei Liu, Hang Yan, Xipeng Qiu, and Xuanjing Huang. 2020. Learning sparse sharing architectures for multiple tasks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8936–8943.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2016. Newsqa: A machine comprehension dataset. *arXiv preprint arXiv:1611.09830*.

Jing Wang, Mayank Kulkarni, and Daniel Preoţiuc-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*, pages 8476–8488.

Genta Indra Winata, Zhaojiang Lin, Jamin Shin, Zihan Liu, and Pascale Fung. 2019. Hierarchical meta-embeddings for code-switching named entity recognition. *arXiv preprint arXiv:1909.08504*.

Wei Wu, Fei Wang, Arianna Yuan, Fei Wu, and Jiwei Li. 2019. Coreference resolution as query-based span prediction. *arXiv preprint arXiv:1911.01746*.

Wenhan Xiong, Jiawei Wu, Hong Wang, Vivek Kulkarni, Mo Yu, Shiyu Chang, Xiaoxiao Guo, and William Yang Wang. 2019. Tweetqa: A social media focused question answering dataset. *arXiv preprint arXiv:1907.06292*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020a. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

Mengge Xue, Bowen Yu, Zhenyu Zhang, Tingwen Liu, Yue Zhang, and Bin Wang. 2020b. Coarse-to-fine pre-training for named entity recognition. *arXiv preprint arXiv:2010.08210*.

Jie Yang, Shuailong Liang, and Yue Zhang. 2018. Design challenges and misconceptions in neural sequence labeling. *arXiv preprint arXiv:1806.04470*.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2022a. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

Jiali Zeng, Yufan Jiang, Yongjing Yin, Xu Wang, Binghuai Lin, and Yunbo Cao. 2022b. Dualner: A dual-teaching framework for zero-shot cross-lingual named entity recognition. *arXiv preprint arXiv:2211.08104*.

Guodong Zhang, Chaoqi Wang, Bowen Xu, and Roger Grosse. 2018a. Three mechanisms of weight decay regularization. *arXiv preprint arXiv:1810.12281*.

Qi Zhang, Jinlan Fu, Xiaoyu Liu, and Xuanjing Huang. 2018b. Adaptive co-attention network for named entity recognition in tweets. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Ying Zhang, Fandong Meng, Yufeng Chen, Jinan Xu, and Jie Zhou. 2021. Target-oriented fine-tuning for zero-resource named entity recognition. *arXiv preprint arXiv:2107.10523*.

Yichun Zhao, Jintao Du, Gongshen Liu, and Huijia Zhu. 2022. Transadv: A translation-based adversarial learning framework for zero-resource cross-lingual named entity recognition. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 742–749.

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 2022*, pages 2670–2680.