AdvPicker: Effectively Leveraging Unlabeled Data via Adversarial Discriminator for Cross-Lingual NER

A Preprint

Weile Chen¹, Huiqiang Jiang², Qianhui Wu³, Börje F. Karlsson⁴, and Yi Guan¹
¹Harbin Institute of Technology, Harbin, 15000, China
²Peking University, Beijing, 100871, China
³Department of Automation, Tsinghua University, Beijing, 100084, China
⁴Microsoft Research, Beijing, 100080, China

chen.weile7@gmail.com, jhq@pku.edu.cn, guanyi@hit.edu.cn, wuqianhui@tsinghua.org.cn, borje.karlsson@microsoft.com

ABSTRACT

Neural methods have been shown to achieve high performance in Named Entity Recognition (NER), but rely on costly high-quality labeled data for training, which is not always available across languages. While previous works have shown that unlabeled data in a target language can be used to improve cross-lingual model performance, we propose a novel adversarial approach (AdvPicker) to better leverage such data and further improve results. We design an adversarial learning framework in which an encoder learns entity domain knowledge from labeled source-language data and better shared features are captured via adversarial training - where a discriminator selects less language-dependent target-language data via similarity to the source language. Experimental results on standard benchmark datasets well demonstrate that the proposed method benefits strongly from this data selection process and outperforms existing state-of-the-art methods; without requiring any additional external resources (e.g., gazetteers or via machine translation).

1 Introduction

Named entity recognition (NER) is a fundamental information extraction task, which seeks to identify named entities in text and classify them into pre-defined entity types (such as person, organization, location, etc.) and it is key in various downstream tasks, e.g., question answering [Mollá et al. 2006]. Neural NER models are highly successful for languages with a large amount of quality annotated data. However, most languages don’t have enough labeled data to train a fully supervised model. This motivates research on cross-lingual transfer, which leverages labeled data from a source language (e.g., English) to address the lack of training data problem in a target language. In this paper, following [Wu and Dredze 2019] and [Wu et al. 2020a], we focus on zero-shot cross-lingual NER, where labeled data is not available in the target language.

The state-of-the-art methods for zero-shot cross-lingual NER are mainly divided into three categories: i) feature-based methods [Wu and Dredze 2019], [Wu et al. 2020b], [Pfeiffer et al. 2020], which train a NER model to capture language-independent features of the labeled source-language data and then apply it to the target language; ii) translation-based methods [Mayhew et al. 2017], [Xie et al. 2018], which build pseudo target-language dataset via translating from labeled source-language data and mapping entity labels; and iii) pseudo-labeling methods, which generate pseudo-labeled data for training a target-language NER model via a source-language model [Wu et al. 2020a] or annotation projection [Ni et al. 2017].

However, each method has its own disadvantages. Feature-based methods only learn the knowledge in the source language, but cannot leverage any target-language information. Translation-based methods require high-quality translation resources, which are expensive to obtain. And pseudo-labeled methods assume that all pseudo-labeled data is beneficial for cross-lingual transfer learning, which is not always the case.

¹Code is publicly available at https://aka.ms/AdvPicker
Therefore, here we propose a novel approach – *AdvPicker* – which combines feature-based and pseudo-labeling methods, while not requiring any extra costly resources (e.g., translation models or parallel data). Furthermore, to address the described problems, we enhance the source-language NER model with unlabeled target language data via adversarial training. Unlike other pseudo-labeling methods, we only leverage the language-independent pseudo-labeled data selected by an adversarial discriminator, to alleviate overfitting the model in language-specific features of the source-language.

Specifically, we first train an encoder and a NER classifier on labeled source-language data to learn entity domain knowledge. Meanwhile, a language discriminator and the encoder are trained on a token-level adversarial task which enhances the ability of the encoder to capture shared features. We then apply the encoder and the NER classifier on unlabeled target-language data to generate pseudo-labels and use an adversarial discriminator to select less language-specific data samples. Finally, we utilize knowledge distillation to train a target-language NER model on this selected dataset.

We evaluate our proposed *AdvPicker* over 3 target languages on standard benchmark datasets. Our experimental results show that the proposed method benefits strongly from this data selection process and outperforms existing SOTA methods; without requiring any additional external resources (e.g., gazetteers or machine translation).

Our major contributions are as follows:

- We propose a novel approach to combine feature-based and pseudo-labeling methods via language adversarial learning for cross-lingual NER;
- We adopt an adversarial discriminator to select what language-independent data to leverage in training a cross-lingual NER model to improved performance. To the best of our knowledge, this is the first successful attempt in selecting data by adversarial discriminator for XL-NER;
- Experiments on standard multi-lingual datasets showcase *AdvPicker* achieves new state-of-the-art results in cross-lingual NER.

2 Related Work

2.1 Cross-Lingual NER

Cross-lingual transfer for NER has been widely studied in recent years. Prior works are divided into three categories: feature-based, translation-based, and pseudo-labeling.

Feature-based methods generally use language-independent features to train a NER model in the labeled source-language data, which include word clusters [Täckström et al. 2012], Wikifier features [Tsai et al. 2016], gazetteers [Zirikly and Hagiwara 2015], and aligned word representations [Ni et al. 2017], [Wu and Dredze 2019], etc. Moreover, for language-independent features, adversarial learning was applied on word/char embedding layers [Huang et al. 2019], [Bari et al. 2020] or encoders [Zhou et al. 2019], [Keung et al. 2019]. Translation-based methods generally use pseudo target-language data translated from labeled source-language data. [Ni et al. 2017] proposed to project labels from the source language into the target language by using word alignment information. Most recent methods translate the annotated corpus in the source language to the target language word-by-word [Xie et al. 2018] or phrase-by-phrase [Mayhew et al. 2017] and then copy the labels for each word/phrase to their translations. While [Jain et al. 2019] proposed to translate full sentences in the source language and project entity labels to target-language sentences.

To leverage unlabeled target-language data, pseudo-labeling methods generate the pseudo-labels by annotation projection on comparable corpora [Ni et al. 2017] or via models trained on source-language labeled data [Wu et al. 2020a].

In this paper, we propose *AdvPicker*, an approach that requires no translation and combines feature-based and pseudo-labeling methods. Moreover, we leverage pseudo-labeled data differently from other pseudo-labeling methods. Through adversarial training, we select language-independent pseudo-labeled data for training a new target-language model.

2.2 Language Adversarial Learning

Language-adversarial training [Zhang et al. 2017] was proposed for the unsupervised bilingual lexicon induction task. And it has been applied in inducing language-independent features for cross-lingual tasks in NER [Zhou et al. 2019], [Xie et al. 2018], text classification [Chen et al. 2019a], and sentiment classification [Chen et al. 2018].

[Keung et al. 2019] proposed a multilingual BERT with sentence-level adversarial learning. However, this method does not improve cross-lingual NER performance significantly. To address this limitation, *AdvPicker* uses multilingual BERT
with token-level adversarial training for cross-lingual NER, which induces more language-independent features for each token embedding.

### 2.3 Knowledge Distillation

Knowledge distillation was proposed to compress models [Bucilă et al. 2006] or ensembles of models [Rusu et al. 2016], [Hinton et al. 2015], [Sanh et al. 2019], [Mukherjee and Hassan Awadallah 2020] via transferring knowledge from one or more models (teacher models) to a smaller one (student model). Besides model compression, knowledge distillation has also been applied to various tasks, like cross-modal learning [Hu et al. 2020], machine translation [Weng et al. 2020], and automated machine learning [Kang et al. 2020].

In this paper, we adapt knowledge distillation to leverage unlabeled data in the cross-lingual NER task. This helps the student model learn richer information from easily obtainable data (with pseudo-labels).

### 3 AdvPicker

In this section, we introduce our approach (AdvPicker) which utilizes the adversarial learning approach to select language-independent pseudo-labeled data for training an effective target-language NER model. Figure 1 illustrates the framework of the proposed AdvPicker. Specifically, as shown in Figure 1(a), we train an encoder and a NER classifier on the labeled source-language data. Meanwhile, a language discriminator and the encoder are trained together on source-language samples (blue solid lines on the left side). Encoder and discriminator are trained for the adversarial task (on the right side). In this way, we can select the most language-independent pseudo-labeled data samples. Finally, we utilize knowledge distillation to train a target-language NER model on this selected dataset.

In the following section, we describe the language-independent data selection process, including the token-level adversarial training, data selection by the discriminator, and knowledge distillation on select language-independent data.

#### 3.1 Token-level Adversarial Training for Cross-Lingual NER

To avoid the model overfitting on language-specific features of the source-language, we propose the token-level adversarial learning (TLADV) framework, which is shown in Figure 1(a).

Following [Keung et al. 2019], we formulate adversarial cross-lingual NER as a multi-task problem: i) NER and ii) binary language classification (i.e source vs. target language). For the NER task, we train the encoder and classification...
layer on NER annotated text in the source language. The encoder learns to capture the NER features of the input sentences and then the classification layer tries to predict the entity labels for each word based on their feature vectors.

For the language classification task, we train a language discriminator and an encoder on the labeled source-language dataset and unlabeled target-language data. The language discriminator is added to classify whether an embedding generated by the encoder is associated to the source or the target language. The encoder tries to produce language-independent embeddings that are difficult for the language discriminator to classify correctly. We define the encoder, the language discriminator, and their objectives as follows:

**Encoder** Given an input sentence \( x = [x_i]_{1 \leq i \leq N} \) with \( N \) words, we feed it into encoder \( E \) to obtain feature vectors \( h = [h_i]_{1 \leq i \leq N} \) for all words:

\[
h = E(x)
\]

where \( E \) is the feature encoder which generates language-independent feature vectors \( h \) for each sentence \( x \). Following Keung et al. [2019], we use multilingual BERT as the feature encoder here and denote the encoder as mBERT-TLADV.

**NER Classifier** We feed \( h \) into the NER classifier which is a linear classification layer with the \( \text{softmax} \) activation function to predict the entity label of token \( x \).

\[
P_\theta(Y_{\text{NER}}) = \text{softmax}(W_{\text{NER}}h + b_{\text{NER}})
\]

where \( P_\theta(Y_{\text{NER}}) \in \mathbb{R}^{|C|} \) is the probability distribution of entity labels for token \( x \) and \( C \) is the entity label set. \( W_{\text{NER}} \in \mathbb{R}^{d_e \times |C|} \) and \( b_{\text{NER}} \in \mathbb{R}^{|C|} \) denote the to-be-learned parameters with \( d_e \) being the dimension of vector \( h \).

**Language Discriminator** The language discriminator is comprised of two linear transformations and a \( \text{ReLU} \) function for classifying token embedding. The \( \text{sigmoid} \) function is used to predict the probability of whether \( h \) belongs to the source language.

\[
P_\theta(Y_{\text{DIS}}) = \sigma(W_{\text{DIS1}} \text{ReLU}(W_{\text{DIS2}}h))
\]

where \( W_{\text{DIS1}} \in \mathbb{R}^{d_d \times d_e} \) and \( W_{\text{DIS2}} \in \mathbb{R}^{d_e \times d_d} \), with \( d_d \) being the hidden dimension of discriminator and \( d_e \) the language classification task label size. \( \sigma \) is the \( \text{sigmoid} \) function to obtain the language probability of each word.

For language-adversarial training, we have 3 loss functions: the encoder loss \( L_E \), the language discriminator loss \( L_{\text{DIS}} \), and the NER task loss \( L_{\text{NER}} \).

**Algorithm 1** Pseudocode for token-level adversarial training on zero-shot cross-lingual NER.

```
Require: training set \( D_E = D_{\text{DIS}} = \{(x, y^{\text{DIS}})\} \) and \( D_{\text{NER}} = \{(x, y^{\text{NER}})\} \), batch size \( bs = 1 \) for clarity, batch number \( B \), the output probability of model \( P_\theta(Y^m) \), the hidden vector of model \( h \).
1: Initialize hidden vector of \( W^*, b^* \);
2: for \( i = 1, \ldots, B \) do
3: for model \( m \in \{\text{NER, E, DIS}\} \) do
4: Sample batch \( (x, y) \in D_m \);
5: # Calculate model hidden vector;
6: \( h = E(x) \), refer to Eq. (1)
7: if model \( m \) equal to model NER then
8: Calculate \( P_\theta(Y_{\text{NER}}) \), Eq. (2)
9: else
10: Calculate \( P_\theta(Y_{\text{DIS}}) \), Eq. (3)
11: end if
12: Calculate model loss \( L_m \) using \( x, y \) and \( P_\theta(Y^m) \), Eq. (4)
13: Update parameters of embeddings w.r.t. the gradients using \( \nabla L_m \) with parameters \( W^m, b^m \) and the encoder \( E \) (if \( m \) in \( \{\text{NER, E}\} \)).
14: end for
15: end for
```

Note that we don’t add these three loss functions together for backward propagation. Parameters of different components in adversarial learning are alternatively updated based on the corresponding loss function, similarly to Keung et al. [2019]. Specifically, for the NER task, the parameters of the encoder and the NER classifier are updated based on
For the adversarial task, the parameters of the encoder are updated based on $L^E$, while the parameters of the discriminator are updated based on $L^{DIS}$. Algorithm 1 shows the pseudocode for the adversarial training process.

$$L^E = -\frac{1}{N}\sum_{i \in [1,N]} \log P_\theta(Y^{DIS}_i = y_i^{DIS})$$
$$+ \log P_\theta(Y^{DIS}_i = y_i^{DIS})$$

$$L^{DIS} = -\frac{1}{N}\sum_{i \in [1,N]} \log P_\theta(Y^{DIS}_i = y_i^{DIS})$$
$$+ \log P_\theta(Y^{DIS}_i = y_i^{DIS})$$

(4)

$$L^{NER} = -\frac{1}{N}\sum_{i \in [1,N]} \log P_\theta(Y^{NER}_i = y_i^{NER})$$

where $x$ is the sentence, $y^{DIS}_i \in \{0, 1\}$ is the ground truth label for the language classification task, $\hat{y}^{DIS}_i \in \{0, 1\}$ is the negative label for the language classification task, and $y^{NER}_i \in \mathbb{R}^{N \times |C|}$ is the ground truth of named entity recognition task for corresponding input $x$.

### 3.2 Language-Independent Data Selection

To obtain the pseudo labels $\hat{y}^{T-NER}_i$ for target-language examples, we apply the learned mBERT-TLADV model on the unlabeled target-language data $x^T$. However, the pseudo-labeled dataset $\mathcal{D} = \{x^T, \hat{y}^{T-NER}_i\}$ may then contain lots of language-specific samples. We then leverage the adversarial discriminator to select pseudo language-independent samples from the generated set.

The language discriminator tries to make the encoder unable to distinguish the language of a token through confrontation. In this way, the encoder should pay more attention to features that are less related to the source language when learning the NER task. After adversarial training, the language discriminator can still correctly classify certain embeddings with a high probability. We define these as language-specific samples. Other samples are ambiguous regarding language (for example, sentences with probability close to 0.5), and they are defined as samples that are more language-independent.

In order to quantify the language independence of each sample, we use the language discriminator to calculate the probability of whether the sentence $x^T$ is from the source language $P_\theta(Y^{DIS}, x^T)$, the formula is as follows:

$$P_\theta(Y^{DIS}, x^T) = \sigma(W^{DIS1} \text{ReLU}(W^{DIS2} h^T))$$

(5)

where $x^T$ and $h^T$, respectively, denote the target-language sentence and its feature vector, and $P_\theta(Y^{DIS}, x^T)$ is the probability of $x^T$ mentioned in Eq. (3).

In order to select from the pseudo-labeled data, we design an index $\ell_{core}$ to represent the language independence of a sentence $x^T$ (the degree of model confusion on different languages). We assume it follows a uniform distribution and reaches its maximum when $P_\theta(Y^{DIS}, x^T) = 0.5$. Conversely, the index is at its minimal value when $P_\theta(Y^{DIS}, x^T)$ is equal to 0 or 1.

$$\ell_{core}(x^T) = 1 - \|P_\theta(Y^{DIS}, x^T) - 0.5\|$$

(6)

We select target-language samples with the highest $\ell_{core}$ in the top $\rho$ as language-independent data. $\rho$ is a hyper-parameter which is the data ratio of pseudo-labeled data. Finally, we obtain the selected target-language pseudo-labeled dataset $\mathcal{D}_{\text{subset}} = \{\hat{x}_{\text{subset}}, \hat{y}_{\text{subset}}^{T-NER}\}$, a subset of target language pseudo-labeled dataset.

There are two reasons for selecting language-independent samples by the language discriminator. First, these samples’ feature vectors contain less language-specific information which is helpful for cross-lingual transfer learning. Second, the NER classifier is trained on source-language labeled data. Therefore, it is more likely to generate high-quality predictive labels on selected target-language samples that have similar feature vectors to source-language samples.

### 3.3 Knowledge Distillation on Language-Independent Data

To leverage such less language-dependent data, we train a target-language NER model on the selected pseudo-labeled data $\mathcal{D}_{\text{subset}}$. Considering a lot of helpful information can be carried in soft targets instead of hard targets Hinton et al.
Table 1: Statistics of the benchmark datasets.

| Language | Type | Train | Dev | Test |
|----------|------|-------|-----|------|
| English [en] | # of Sentence | 14,987 | 3,466 | 3,684 |
| (CoNLL-2003) | # of Entity | 23,499 | 5,942 | 5,648 |
| German [de] | # of Sentence | 12,705 | 3,068 | 3,160 |
| (CoNLL-2003) | # of Entity | 11,851 | 4,833 | 3,673 |
| Spanish [es] | # of Sentence | 8,323 | 1,915 | 1,517 |
| (CoNLL-2002) | # of Entity | 18,798 | 4,351 | 3,558 |
| Dutch [nl] | # of Sentence | 15,806 | 2,895 | 5,195 |
| (CoNLL-2002) | # of Entity | 13,344 | 2,616 | 3,941 |

Following [Wu et al., 2020a], the loss function \( L_{KD} \) is defined as the mean squared error (MSE) between the prediction output \( \mathcal{P}(Y^{T-\text{NER}}) \) and the soft labels of the selected data, which is formulated as:

\[
L_{KD} = \frac{1}{N} \sum_{i=1}^{N} (\mathcal{P}(Y^{T-\text{NER}}) - \tilde{y}_{\text{subset}}^{T-\text{NER}})^2
\]

where \( \mathcal{P}(Y^{T-\text{NER}}) \) is the prediction probability of the \( x^T \). By minimizing the MSE loss, the student model is trained supervised on the target-language selected data pseudo-labels.

For inference in the target language, we only apply the student model on test cases to predict the probability distribution of entity labels for each token in sentences, as Eq.(7). To ensure the entity labels follow the NER tagging scheme, the prediction result is generated by Viterbi decoding [Chen et al., 2019b].

4 Experiments

4.1 Experiment Settings

Datasets

We conduct experiments in 4 different languages: English [en], Spanish [es], Dutch [nl], German [de]. Spanish and Dutch data is from the CoNLL-2002 NER shared task [Tjong Kim Sang, 2002a], while English and German are from CoNLL-2003 [Tjong Kim Sang, 2002b]. Table 1 presents some basic statistics of the datasets used in our experiments.

Note that for each target language, we only use text in its training set to train our model with these unlabeled target language data. In adversarial learning, we randomly sample data from all target languages and construct a target-language dataset of the same size as the English training dataset.
AdvPicker: Effectively Leveraging Unlabeled Data via Adversarial Discriminator...

A Model de es nl Avg

| Method               | de    | es    | nl    | Avg |
|----------------------|-------|-------|-------|-----|
| Täckström et al. [2012] | 40.40 | 59.30 | 58.40 | 52.70 |
| Tsai et al. [2016]    | 48.12 | 60.55 | 61.56 | 56.74 |
| Ni et al. [2017]      | 58.50 | 65.10 | 65.40 | 63.00 |
| Mayhew et al. [2017]  | 57.23 | 64.10 | 63.37 | 61.57 |
| Xie et al. [2018]     | 57.76 | 72.37 | 71.25 | 67.13 |
| Jain et al. [2019]    | 61.5  | 73.5  | 69.9  | 68.30 |
| Bari et al. [2020]    | 65.24 | 75.93 | 74.61 | 71.93 |
| Wu and Dredze [2019]  | 69.56 | 74.96 | 77.57 | 73.57 |
| Keung et al. [2019]   | 71.9  | 74.3  | 77.6  | 74.60 |
| Wu et al. [2020a]     | 73.16 | 76.75 | 80.44 | 76.78 |
| Wu et al. [2020b]     | 73.61 ± 0.39 | 77.3 ± 0.78 | 81.2 ± 0.83 | 77.37 ± 0.67 |
| mBERT-ft              | 72.59 ± 0.31 | 75.12 ± 0.83 | 80.34 ± 0.27 | 76.02 ± 0.47 |
| mBERT-TLADV           | 73.89 ± 0.56 | 76.92 ± 0.62 | 80.12 ± 0.56 | 77.14 ± 0.58 |
| AdvPicker             | 75.01 ± 0.50 | 79.00 ± 0.21 | 82.90 ± 0.44 | 78.97 ± 0.38 |

Table 2: Results of our approach and prior state-of-the-art methods for zero-shot cross-lingual NER. * denotes the version of the method without additional data.

Implementation Details

We implement AdvPicker using PyTorch 1.6.0. For data pre-processing, we leverage WordPiece Wu et al. [2016] to tokenize each sentence into a sequence of sub-words which are then fed into the model. For the encoder (i.e., \( E \) in Eq. (1)) and student model (i.e., \( h_{stu} \) in Eq. (7)), we employ the pre-trained cased multilingual BERT in HuggingFace’s Transformers [Wolf et al., 2020] as backbone model, which has 12 transformer blocks, 12 attention heads, and 768 hidden units.

We empirically select the following hyper-parameters. Specifically, referring to the settings of Wu et al. [2020b], we freeze the parameters of the embedding layer and the bottom three layers of the multilingual BERT used in the encoder and the NER student model. We train all models using a batch size of 32, maximum sequence length of 128, a dropout rate of 0.1, and use AdamW Loshchilov and Hutter [2019] as optimizer. For sequence prediction, we apply Viterbi decoding Chen et al. [2019b] on all models in our experiments. Following Keung et al. [2020], in all experiments the other hyper-parameters are tuned on each target language dev set. We train all models for 10 epochs and choose the best model checkpoint with the target dev set. For adversarial learning, we set the learning rate of 6e-5 for the NER loss \( L_{NER} \) and 6e-7 for both loss encoder \( L_E \) and discriminator loss \( L_{DIS} \). For knowledge distillation, we use a learning rate of 6e-5 for the student models. We set the hidden dimension of the discriminator as 500. For data selection, \( \rho \) is set to 0.8. Following Jong Kim Sang [2002a], we use the entity level F1-score as evaluation metric. Moreover, experiments are repeated 5 times for different random seeds on each corpus.

Note that the selected data from the discriminator is generated by combination of output from mBERT-TLADVs with different random seeds, as we observe only a small number of samples with high \( \ell_{score} \) in the selected data generated by each model. Specifically, for each target-language sentence \( x_T \), there are 5 corresponding soft label sequences generated from 5 different mBERT-TLADV models. From those, only sequences that have the highest sum of each predicted label confidence are kept.

Our models are trained on a Tesla P100 GPU (16GB). mBERT-TLADV has 178M parameters and trains in \( \approx 130 \) min, while the student models \( h_{stu} \) have 177M parameters and take \( \approx 21 \) min.

4.2 Comparison with State-of-The-Art Results

Table 2 reports the zero-shot cross-lingual NER results of different methods on the 3 target languages. These include AdvPicker, prior SOTA methods, and two re-implemented baseline methods, i.e., mBERT-TLADV (Section 3.1) and mBERT-ft (mBERT fine-tuned on labeled source-language data). Note that some existing methods use the translation model as an additional data transfer source, whereas our method does not. For a fair comparison, we compare against the version of UniTrans Wu et al. [2020a] w/o translation (as reported in their paper). Our method outperforms the existing methods with F1-scores of 75.01, 79.90, and 82.90, when using only source-language labeled data and target-language unlabeled data. Particularly, compared with Unitrans* (previous SOTA), AdvPicker achieves an improvement of

https://github.com/huggingface
AdvPicker: Effectively Leveraging Unlabeled Data via Adversarial Discriminator...

Table 3: Accuracy of discriminators for different models in the three target languages.

|           | de  | es  | nl  | Avg |
|-----------|-----|-----|-----|-----|
| mBERT     | 99.08 | 99.66 | 98.98 | 99.24 |
| mBERT-ft  | 98.38 | 98.66 | 97.27 | 98.10 |
| mBERT-TLADV | 79.62 | 82.89 | 77.45 | 79.99 |

Table 4: Teacher model F1 scores over target language training sets, w/o adversarial approach and distillation.

| Data type   | de   | es   | nl   | Total |
|-------------|------|------|------|-------|
| Selected data | 9733  | 6724  | 12668 | 29125 |
| Other data  | 2434  | 1681  | 3168  | 7283  |

Table 5: Sentence numbers of the Selected/Other splits in the target language training sets.

| Model                  | de     | es     | nl     | Avg   |
|------------------------|--------|--------|--------|-------|
| AdvPicker              | 75.02  | 79.00  | 82.90  | 78.97 |
| mBERT-ft               | 72.59 (-2.43) | 75.12 (-3.88) | 80.34 (-2.56) | 76.02 (-2.95) |
| mBERT-TLADV            | 73.89 (-1.13) | 76.92 (-2.08) | 80.62 (-2.28) | 77.14 (-1.83) |
| AdvPicker w/o KD       | 73.98 (-1.04) | 77.91 (-1.09) | 80.55 (-2.35) | 77.48 (-1.49) |
| AdvPicker w All-Data   | 74.02 (-1.00) | 78.72 (-0.28) | 80.69 (-2.21) | 77.81 (-1.16) |

Table 6: Ablation study for the proposed AdvPicker, where numbers in parenthesis denote performance change.

F1-score ranging from 1.41 in German to 1.71 in Dutch. Furthermore, our result is comparable to the full UniTrans using also translation (0.04 F1 difference on average).

Besides, AdvPicker achieves an average F1-score improvement of 1.83 over mBERT-TLADV and 2.95 over mBERT-ft. These results well demonstrate the effectiveness of the proposed approach, which is mainly attributed to more effectively leveraging unlabeled target language data and selecting the language-independent data for cross-lingual transfer.

4.3 Quality of Selected Data

Language-Independence We use the selected dataset \( D_{\text{subset}} \) to train student models via knowledge distillation. \( P_\theta(Y^{\text{T-NER}}) \) in \( D_{\text{subset}} \) is calculated over feature vectors generated by mBERT-TLADV. To validate the language-independence of these feature vectors, we apply three discriminators defined as in Eq. (3) to classify the token feature vectors from three different encoders: mBERT, mBERT-ft, and mBERT-TLADV.

Unlike in the adversarial learning setting, we fix the parameters of the three encoders and only train the discriminators. We use each language training set to train discriminators and evaluate on each target language corresponding test set. Table 3 reports the discriminator accuracy for 3 different encoders. We can see that the classification accuracy is reduced with adversarial training, which suggests that the similarity of feature vectors between the source language and the target language is improved; which further demonstrates the feature vectors become more language-independent when adversarial training is applied.

Pseudo Labels To evaluate the language-independent quality of pseudo labels, we calculate the F1 scores of the pseudo labels and the number of sentences involved. We denote language-independent data and language-specific data as “selected data” and “other data” respectively.
### Table 7: F1 scores over the Select/Other test set splits in different target languages. AdvPicker has better performance also on Other data for all languages.

Table 4 reports the pseudo labels F1 scores of both language-independent data and language-specific data for each target language using mBERT-TLADV. Generally, the average F1 score of language-independent data is 12.87 points higher than language-specific data, which suggests that language-independent data has higher quality pseudo-labels. Furthermore, the selected data contains less language-specific information.

Table 5 reports the number of language-independent and language-specific examples for each language. From these results (Tables 4, 5), we observe that the selected data is still high-quality, even if we set a very loose threshold (80% of unlabeled data being selected).

#### 4.4 Model Performance over Selected/Other Data Splits

In order to better analyse the behaviour of AdvPicker across data variations, we use the trained language discriminator to split the target language test sets into Selected and Other partitions (similarly to how the training set is processed). Table 7 shows the different models’ F1 scores for the partitioned data.

From Table 7, we can draw these conclusions:

1. As expected, models perform better over Selected data than over Other data;
2. AdvPicker is only trained on Selected data, but nonetheless outperforms all baseline models in both data partitions;
3. AdvPicker’s approach effectively selects examples with better features and is not over-biased towards Selected data.

#### 4.5 Ablation Study

To validate the contributions of different process in the proposed AdvPicker, we introduce the following variants of our approach and baselines to perform an ablation study: 1) AdvPicker w/o KD, which directly combines the prediction of test data from mBERT-TLADVs with different seeds without knowledge distillation on pseudo-labeled training data. 2) AdvPicker w All-Data, which trains a student model on all target-language pseudo-labeled data generated by mBERT-TLADV. 3) mBERT-ft, mBERT fine-tuned on source-language labeled data. 4) mBERT-TLADV (Section 3.1), meaning mBERT trained on source-language labeled data with token-level adversarial learning.

Table 6 reports the performance of each method and their performance drops compared to AdvPicker. Moreover, we can draw more in-depth observations as follows:

1. Comparing AdvPicker with AdvPicker w/o KD and AdvPicker w All-Data, we can see that selecting the language-independent data is reasonable. That also validates the effectiveness of the model trained on language-independent data via knowledge distillation.
2. mBERT-ft outperforms mBERT-TLADV. Such results well demonstrate that token-level adversarial learning is helpful to train a language-independent feature encoder and brings performance improvement.
3. By comparing the F1 scores of AdvPicker w All-Data and AdvPicker on the target languages, we observe that training on selected data brings higher performance improvements on larger datasets, e.g., German [de] and Dutch [nl], and lower improvements on the smaller Spanish [es] dataset. Although selected data has high-quality pseudo labels, smaller sizes of selected datasets may limit performance improvements.

#### 4.6 Stability Analysis

Because BERT fine-tuning is known to be unstable in few-shot tasks, as discussed in [Zhang et al., 2021], mBERT-based methods’ performances on the CoNLL NER dataset are likely also unstable. To evaluate the stability of AdvPicker, we compare the standard deviation of F1 scores for mBERT-ft, Unitrans, and AdvPicker.
Table 2 includes the standard deviation of F1 scores over five runs for each model. AdvPicker has a lower average standard deviation in the three target languages than the other mBERT-based methods. Such results demonstrate that selected data can bring a degree of stability to the model, or limit instability, as the student model in AdvPicker is trained on selected data with the soft labels from other trained models.

5 Conclusion

In this paper, we propose a novel approach to combine the feature-based method and pseudo labeling via language adversarial learning for cross-lingual NER. AdvPicker is the first successful attempt in selecting language-independent data by adversarial discriminator to cross-lingual NER. Our experimental results show that the proposed system benefits strongly from this new data selection process and outperforms existing state-of-the-art methods, even without requiring additional extra resources.

References

Diego Mollá, Menno van Zaalen, and Daniel Smith. Named entity recognition for question answering. In Proceedings of the Australasian Language Technology Workshop 2006, pages 51–58, Sydney, Australia, November 2006. URL https://www.aclweb.org/anthology/U06-1009

Shijie Wu and Mark Dredze. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. 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), pages 833–844, Hong Kong, China, November 2019. Association for Computational Linguistics. doi:10.18653/v1/D19-1077 URL https://www.aclweb.org/anthology/D19-1077

Qianhui Wu, Zijia Lin, Börje F. Karlsson, Biqing Huang, and Jian-Guang Lou. Unitrans: Unifying model transfer and data transfer for cross-lingual named entity recognition with unlabeled data. In Christian Bessiere, editor, Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020, pages 3926–3932. ijcai.org, 2020a. doi:10.24963/ijcai.2020/543 URL https://doi.org/10.24963/ijcai.2020/543

Qianhui Wu, Zijia Lin, Guoxin Wang, Hui Chen, Börje F. Karlsson, Biqing Huang, and Chin-Yew Lin. Enhanced meta-learning for cross-lingual named entity recognition with minimal resources. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, pages 9274–9281, 2020b. URL https://doi.org/10.1609/aaai.v34i05.6466

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 7654–7673, Online, November 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-main.617 URL https://www.aclweb.org/anthology/2020.emnlp-main.617

Stephen Mayhew, Chen-Tse Tsai, and Dan Roth. Cheap translation for cross-lingual named entity recognition. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 2536–2545, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi:10.18653/v1/D17-1269 URL https://www.aclweb.org/anthology/D17-1269

Jiateng Xie, Zhilin Yang, Graham Neubig, Noah A. Smith, and Jaime Carbonell. Neural cross-lingual named entity recognition with minimal resources. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 369–379, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi:10.18653/v1/D18-1034 URL https://www.aclweb.org/anthology/D18-1034

Jian Ni, Georgiana Dinu, and Radu Florian. Weakly supervised cross-lingual named entity recognition via effective annotation and representation projection. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1470–1480, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi:10.18653/v1/P17-1135 URL https://www.aclweb.org/anthology/P17-1135

Oscar Täckström, Ryan McDonald, and Jakob Uszkoreit. Cross-lingual word clusters for direct transfer of linguistic structure. In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 477–487, Montréal, Canada, June 2012. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/N12-1052

Chen-Tse Tsai, Stephen Mayhew, and Dan Roth. Cross-lingual named entity recognition via wikification. In Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning, pages 219–228, Berlin, Germany, August 2016. Association for Computational Linguistics. doi:10.18653/v1/K16-1022 URL https://www.aclweb.org/anthology/K16-1022
Ayah Zirikly and Masato Hagiwara. Cross-lingual transfer of named entity recognizers without parallel corpora. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 390–396, Beijing, China, July 2015. Association for Computational Linguistics. doi:10.3115/v1/P15-2064 URL https://www.aclweb.org/anthology/P15-2064

Lifu Huang, Heng Ji, and Jonathan May. Cross-lingual multi-level adversarial transfer to enhance low-resource name tagging. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 3823–3833, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi:10.18653/v1/N19-1383 URL https://www.aclweb.org/anthology/N19-1383

M. Saiful Bari, Shafiq R. Joty, and Prathyusha Jwalapuram. Zero-resource cross-lingual named entity recognition. In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, pages 7415–7423. AAAI Press, 2020. URL https://aaai.org/ojs/index.php/AAAI/article/view/6237

Joey Tianyi Zhou, Hao Zhang, Di Jin, Hongyuan Zhu, Meng Fang, Rick SIow Mong Goh, and Kenneth Kwok. Dual adversarial neural transfer for low-resource named entity recognition. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3461–3471, Florence, Italy, July 2019. Association for Computational Linguistics. doi:10.18653/v1/P19-1336 URL https://www.aclweb.org/anthology/P19-1336

Phillip Keung, Yichao Lu, and Vikas Bhardwaj. Adversarial learning with contextual embeddings for zero-resource cross-lingual classification and NER. 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), pages 1355–1360, Hong Kong, China, November 2019. Association for Computational Linguistics. doi:10.18653/v1/D19-1138 URL https://www.aclweb.org/anthology/D19-1138

Alankar Jain, Bhargavi Paranjape, and Zachary C. Lipton. Entity projection via machine translation for cross-lingual NER. 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), pages 1083–1092, Hong Kong, China, November 2019. Association for Computational Linguistics. doi:10.18653/v1/D19-1100 URL https://www.aclweb.org/anthology/D19-1100

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. Adversarial training for unsupervised bilingual lexicon induction. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1959–1970, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi:10.18653/v1/P17-1179 URL https://www.aclweb.org/anthology/P17-1179

Xilun Chen, Ahmed Hassan Awadallah, Hany Hassan, Wei Wang, and Claire Cardie. Multi-source cross-lingual model transfer: Learning what to share. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3098–3112, Florence, Italy, July 2019a. Association for Computational Linguistics. doi:10.18653/v1/P19-1299 URL https://www.aclweb.org/anthology/P19-1299

Xilun Chen, Yu Sun, Ben Athiwaratkun, Claire Cardie, and Kilian Weinberger. Adversarial deep averaging networks for cross-lingual sentiment classification. Transactions of the Association for Computational Linguistics, 6:557–570, 2018. doi:10.1162/tacl_a_00039 URL https://www.aclweb.org/anthology/Q18-1039

Cristian Buciluă, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 535–541, 2006.

Andrei A. Rusu, Sergio Gomez Colmenarejo, Çağlar Gülçehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. Policy distillation. In Yoshua Bengio and Yann LeCun, editors, 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings, 2016. URL http://arxiv.org/abs/1511.06295

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531, 2015.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. In Proceedings of the NeurIPS Workshop on Energy Efficient Machine Learning and Cognitive Computing (EMC2), dec 2019. URL https://www.emc2-ai.org/assets/docs/neurips-19/emc2-neurips19-sanh-poster.pdf

Subhabrata Mukherjee and Ahmed Hassan Awadallah. XtremeDistil: Multi-stage distillation for massive multilingual models. In Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics, pages 2221–
