Dual Adversarial Neural Transfer for Low-Resource Named Entity Recognition

Joey Tianyi Zhou[1,†], **Hao Zhang**[2,†], **Di Jin**[3], **Hongyuan Zhu**[4,‡],
Meng Fang[5], **Rick Siow Mong Goh**[1], **Kenneth Kwok**[1]
[1]IHPC, A*STAR [2]A*AI, A*STAR [3]CSAIL, MIT [4]I^{2}R, A*STAR [5]Tencent Robotics X
{zhouty,gohsm,kenkwok}@ihpc.a-star.edu.sg,
{zhang_hao@scei,zhuh@i2r}.a-star.edu.sg,
jindi15@mit.edu, mfang@tencent.com

Abstract

We propose a new neural transfer method termed Dual Adversarial Transfer Network (DATNet) for addressing low-resource Named Entity Recognition (NER). Specifically, two variants of DATNet, i.e., DATNet-F and DATNet-P, are investigated to explore effective feature fusion between high and low resource. To address the noisy and imbalanced training data, we propose a novel Generalized Resource-Adversarial Discriminator (GRAD). Additionally, adversarial training is adopted to boost model generalization. In experiments, we examine the effects of different components in DATNet across domains and languages, and show that significant improvement can be obtained especially for low-resource data, without augmenting any additional hand-crafted features and pre-trained language model.

1 Introduction

Named entity recognition (NER) is an important step in most natural language processing (NLP) applications. It detects not only the type of named entity, but also the entity boundaries, which requires deep understanding of the contextual semantics to disambiguate the different entity types of same tokens. To tackle this challenging problem, most early studies were based on handcrafted rules, which suffered from limited performance in practice. Current methods are devoted to developing learning based algorithms, especially neural network based methods, and have been advancing the state-of-the-art progressively (Collobert et al., 2011; Huang et al., 2015; Lample et al., 2016; Chiu and Nichols, 2016; Ma and Hovy, 2016). These end-to-end models generalize well on new entities based on features automatically learned from the data. However, when the annotated corpora is small, especially in the low resource scenario (Zhang et al., 2016), the performance of these methods degrades significantly since the hidden feature representations cannot be learned adequately.

Recently, more and more approaches have been proposed to address low-resource NER. Early works (Chen et al., 2010; Li et al., 2012) primarily assumed a large parallel corpus and focused on exploiting them to project information from high- to low-resource. Unfortunately, such a large parallel corpus may not be available for many low-resource languages. More recently, cross-resource word embedding (Fang and Cohn, 2017; Adams et al., 2017; Yang et al., 2017) was proposed to bridge the low- and high-resources and enable knowledge transfer. Although the aforementioned transfer-based methods show promising performance in low-resource NER, there are two issues remain further study: 1) *Representation Difference* - they did not consider the representation difference across resources and enforced the feature representation to be shared across languages/domains; 2) *Resource Data Imbalance* - the training size of high-resource is usually much larger than that of low-resource. The existing methods neglect such difference in their models, resulting in poor generalization.

In this work, we present a general neural transfer framework termed **Dual Adversarial Transfer Network (DATNet)** to address the above issues in a unified framework for low-resource NER. Specifically, to handle the representation difference, we first investigate on two architectures of hidden layers (Bi-LSTM) for transfer. The first one is that all the units in hidden layers are common units shared across languages/domains. Another is composed of both private and common units, where the private part preserves the independent language/domain information. Extensive

[†] The first two authors contributed equally.
[‡] Corresponding author.

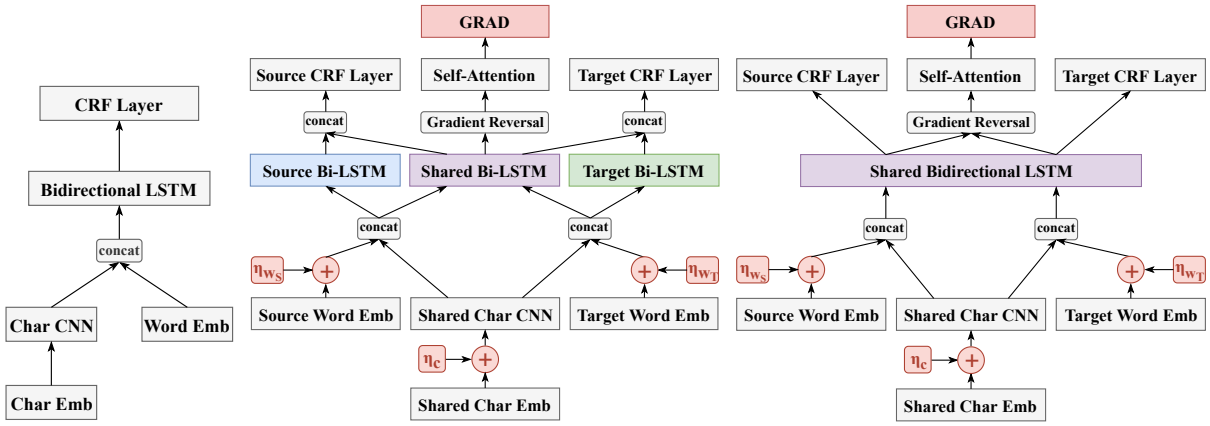
Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3461–3471
Florence, Italy, July 28 - August 2, 2019.

experiments are conducted to show that there is not always a winner and two transfer strategies have their own advantages over each other in different situations, which is largely ignored by existing research. On top of common units, the adversarial discriminator (AD) loss is introduced to encourage the resource-agnostic representation so that the knowledge from high resource can be more compatible with low resource. To handle the resource data imbalance issue, we further propose a variant of the AD loss, termed *Generalized Resource-Adversarial Discriminator (GRAD)*, to impose the resource weight during training so that low-resource and hard samples can be paid more attention to. In addition, we create adversarial samples to conduct the *Adversarial Training (AT)*, further improving the generalization and alleviating over-fitting problem. We unify two kinds of adversarial learning, i.e., GRAD and AT, into one transfer learning model, termed Dual Adversarial Transfer Network (DATNet), to achieve end-to-end training and obtain significant improvements on a series of NER tasks In contrast with prior methods, we do *not* use additional hand-crafted features and do *not* use cross-lingual word embeddings as well as pre-trained language models (Peters et al., 2018; Radford, 2018; Akbik et al., 2018; Devlin et al., 2018) when addressing the cross-language tasks.

2 Related Work

Named Entity Recognition NER is typically framed as a sequence labeling task which aims at automatic detection of named entities (e.g., person, organization, location and etc.) from free text (Marrero et al., 2013). The early works applied CRF, SVM, and perception models with hand-crafted features (Ratinov and Roth, 2009; Passos et al., 2014; Luo et al., 2015). With the advent of deep learning, research focus has been shifting towards deep neural networks (DNN), which requires little feature engineering and domain knowledge (Lample et al., 2016; Zukov Gregoric et al., 2018; Zhou et al., 2019). (Collobert et al., 2011) proposed a feed-forward neural network with a fixed sized window for each word, which failed in considering useful relations between long-distance words. To overcome this limitation, (Chiu and Nichols, 2016) presented a bidirectional LSTM-CNNs architecture that automatically detects word- and character-level features. Ma and Hovy (2016) further extended it into bidirectional LSTM-CNNs-CRF architecture, where the CRF module was added to optimize the output label sequence. Liu et al. (2018) proposed task-aware neural language model termed LM-LSTM-CRF, where character-aware neural language models were incorporated to extract character-level embedding under a multi-task framework.

Transfer Learning for NER Transfer learning can be a powerful tool to low resource NER tasks. To bridge high and low resource, transfer learning methods for NER can be divided into two types: the parallel corpora based and the shared representation based transfer. Early works mainly focused on exploiting parallel corpora to project information between the high- and low-resource languages (Yarowsky et al., 2001; Chen et al., 2010; Li et al., 2012; Feng et al., 2018). For example, (Chen et al., 2010) and (Feng et al., 2018) proposed to jointly identify and align bilingual named entities. Ni *et al.* (Ni and Florian, 2016; Ni et al., 2017) utilized the Wikipedia entity type mappings to improve low-resource NER. (Mayhew et al., 2017) created a cross-language NER system, which works well for very minimal resources by translate annotated data of high-resource into low-resource. On the other hand, the shared representation methods do not require the parallel correspondence (Rei and Søgaard, 2018). For instance, (Fang and Cohn, 2017) proposed cross-lingual word embeddings to transfer knowledge across resources. (Yang et al., 2017) presented a transfer learning approach based on deep hierarchical recurrent neural network, where full/partial hidden features between source and target tasks are shared. (Al-Rfou' et al., 2015) built massive multilingual annotators with minimal human expertise by using language agnostic techniques. (Cotterell and Duh, 2017) proposed character-level neural CRFs to jointly train and predict low- and high-resource languages. (Pan et al., 2017) proposes a large-scale cross-lingual named entity dataset which contains 282 languages for evaluation. In addition, multi-task learning (Yang et al., 2016; Luong et al., 2016; Rei, 2017; Aguilar et al., 2017; Hashimoto et al., 2017; Lin et al., 2018) shows that jointly training on multiple tasks/languages helps improve performance. Different from transfer learning methods, multi-task learning aims at improving the performance of all the resources instead of low resource only.



(a) Base Model (b) DATNet-P (c) DATNet-F

Figure 1: The general architecture of proposed models.

Adversarial Learning Adversarial learning originates from Generative Adversarial Nets (GAN) (Goodfellow et al., 2014), which shows impressing results in computer vision. Recently, many papers have tried to apply adversarial learning to NLP tasks. (Liu et al., 2017) presented an adversarial multi-task learning framework for text classification. (Gui et al., 2017) applied the adversarial discriminator to POS tagging for Twitter. (Kim et al., 2017) proposed a language discriminator to enable language-adversarial training for cross-language POS tagging. Apart from adversarial discriminator, adversarial training is another concept originally introduced by (Szegedy et al., 2014; Goodfellow et al., 2015) to improve the robustness of image classification model by injecting malicious perturbations into input images. Recently, (Miyato et al., 2017) proposed a semi-supervised text classification method by applying adversarial training, where for the first time adversarial perturbations were added onto word embeddings. (Yasunaga et al., 2018) applied adversarial training to POS tagging. Different from all these adversarial learning methods, our method is more general and integrates both the adversarial discriminator and adversarial training in an unified framework to enable end-to-end training.

3 Dual Adversarial Transfer Network

In this section, we introduce two transfer architectures for DATNet in detail. For the base model, we follow the state-of-the-art LSTM-CNN-CRF based structure (Huang et al., 2015; Lample et al., 2016; Chiu and Nichols, 2016; Ma and Hovy, 2016) for NER task, as shown in Figure 1(a).

3.1 Character-level Encoder

Previous works have shown that character features can boost sequence labeling performance by capturing morphological and semantic information (Lin et al., 2018). For low-resource dataset to obtain high-quality word features, character features learned from other language/domain may provide crucial information for labeling, especially for rare and out-of-vocabulary words. Character-level encoder usually contains BiLSTM (Lample et al., 2016) and CNN (Chiu and Nichols, 2016; Ma and Hovy, 2016) approaches. In practice, (Reimers and Gurevych, 2017) observed that the difference between the two approaches is statistically insignificant in sequence labeling tasks, but character-level CNN is more efficient and has less parameters. Thus, we use character-level CNN and share character features between high- and low-resource tasks to enhance the representations of low-resource.

3.2 Word-level Encoder

To learn a better word-level representation, we concatenate the character-level features of each word with a latent word embedding as $\mathbf{w}_i = [\mathbf{w}_i^{char}, \mathbf{w}_i^{emb}]$, where the latent word embedding $\mathbf{w}_i^{emb}$ is initialized with pre-trained embeddings and fixed during training. One unique characteristic of NER is that the historical and future input for a given time step could be useful for label inference. To exploit such a characteristic, we use bidirectional LSTM architecture (Hochreiter and Schmidhuber, 1997) to extract contextualized word-level features. In this way, we can gather the information from the past and future

for a particular time frame t as follows, $\overrightarrow{\mathbf{h}}_t = \texttt{lstm}(\overrightarrow{\mathbf{h}}_{t-1}, \mathbf{w}_t)$, $\overleftarrow{\mathbf{h}}_t = \texttt{lstm}(\overleftarrow{\mathbf{h}}_{t+1}, \mathbf{w}_t)$. After the LSTM layer, the representation of a word is obtained by concatenating its left and right context representation as follows, $\mathbf{h}_t = [\overrightarrow{\mathbf{h}}_t, \overleftarrow{\mathbf{h}}_t]$.

To consider the resource representation difference on word-level features, we introduce two kinds of transferable word-level encoder in our model, namely DATNet-Full Share (DATNet-F) and DATNet-Part Share (DATNet-P). In DATNet-F, all the BiLSTM units are shared by both resources while word embeddings for different resources are disparate. The illustrative figure is depicted in the Figure 1(c). Different from the DATNet-F, the DATNet-P decomposes the BiLSTM units into the shared component and the resource-related one, which is shown in the Figure 1(b). Different from existing works (Yang et al., 2017; Fang and Cohn, 2017; Cotterell and Duh, 2017; Cao et al., 2018), in this work, we investigate the performance of two different shared representation architectures on different tasks and give the corresponding guidance (see Section 4.5).

3.3 Generalized Resource-Adversarial Discriminator

In order to make the feature representation extracted from the source domain more compatible with those from the target domain, we encourage the outputs of the shared BiLSTM part to be resource-agnostic by constructing a resource-adversarial discriminator, which is inspired by the Language-Adversarial Discriminator proposed by (Kim et al., 2017). Unfortunately, previous works did not consider the imbalance of training size for two resources. Specifically, the target domain consists of very limited labeled training data, e.g., 10 sentences. In contrast, labeled training data in the source domain are much richer, e.g., 10k sentences. If such imbalance was not considered during training, the stochastic gradient descent (SGD) optimization would make the model more biased to high resource (Lin et al., 2017b). To address this imbalance problem, we impose a weight α on two resources to balance their influences. However, in the experiment we also observe that the easily classified samples from high resource comprise the majority of the loss and dominate the gradient. To overcome this issue, we further propose Generalized Resource-Adversarial Discriminator (GRAD) to enable adaptive weights for each sample which focuses the model training on hard samples.

To compute the loss of GRAD, the output sequence of the shared BiLSTM is firstly encoded into a single vector via a self-attention module (Bahdanau et al., 2015), and then projected into a scalar r via a linear transformation. The loss function of the resource classifier is formulated as:

$$\begin{aligned}\ell_{GRAD} = -\sum_i \{&\mathbf{I}_{i\in\mathcal{D}_S}\alpha(1-r_i)^\gamma \log r_i \\ &+ \mathbf{I}_{i\in\mathcal{D}_T}(1-\alpha) r_i^\gamma \log(1-r_i)\}\end{aligned} \tag{1}$$

where $\mathbf{I}_{i\in\mathcal{D}_S}, \mathbf{I}_{i\in\mathcal{D}_T}$ are the identity functions to denote whether a sentence is from high resource (source) and low resource (target), respectively; α is a weighting factor to balance the loss contribution from high and low resource; the parameter $(1-r_i)^\gamma$ (or r_i^γ) controls the loss contribution from individual samples by measuring the discrepancy between prediction and true label (easy samples have smaller contribution); and γ scales the contrast of loss contribution from hard and easy samples. In practice, the value of γ does not need to be tuned much and usually set as 2 in our experiment. Intuitively, the weighting factors α and $(1-r_i)^\gamma$ reduce the loss contribution from high resource and easy samples, respectively. Note that though the resource classifier is optimized to minimize the resource classification error, when the gradients originated from the resource classification loss are back-propagated to the other model parts than the resource classifier, they are negated for parameter updates so that these bottom layers are trained to be resource-agnostic.

3.4 Label Decoder

The label decoder induces a probability distribution over sequences of labels, conditioned on the word-level encoder features. In this paper, we use a linear chain model based on the first-order Markov chain structure, termed the chain conditional random field (CRF) (Lafferty et al., 2001), as the decoder. In this decoder, there are two kinds of cliques: local cliques and transition cliques. Specifically, local cliques correspond to the individual elements in the sequence. And transition cliques, on the other hand, reflect the evolution of states between two neighboring elements at time $t-1$ and t and we define the transition distribution as θ. Formally, a linear-chain CRF can be written as $p(\mathbf{y}|\mathbf{h}_{1:T}) =$

Benchmark	Resource	Language	# Training Tokens (# Entities)	# Dev Tokens (# Entities)	# Test Tokens (# Entities)
CoNLL-2003	Source	English	204,567 (23,499)	51,578 (5,942)	46,666 (5,648)
Cross-language NER					
CoNLL-2002	Target	Spanish	207,484 (18,797)	51,645 (4,351)	52,098 (3,558)
CoNLL-2002	Target	Dutch	202,931 (13,344)	37,761 (2,616)	68,994 (3,941)
Cross-domain NER					
WNUT-2016	Target	English	46,469 (2,462)	16,261 (1,128)	61,908 (5,955)
WNUT-2017	Target	English	62,730 (3,160)	15,733 (1,250)	23,394 (1,740)

Table 1: Statistics of CoNLL and WNUT Named Entity Recognition Datasets.

$\frac{1}{Z(\mathbf{h}_{1:T})} \exp\left\{\sum_{t=2}^{T} \theta_{y_{t-1},y_t} + \sum_{t=1}^{T} \mathbf{W}_{y_t}\mathbf{h}_t\right\}$, where $Z(\mathbf{h}_{1:T})$ is a normalization term and $\mathbf{y}$ is the sequence of predicted labels as follows: $\mathbf{y} = y_{1:T}$. Model parameters are optimized to maximize this conditional log likelihood, which acts as the objective function of the model. We define the loss function for source and target resources as follows, $\ell_S = -\sum_i \log p(\mathbf{y}|\mathbf{h}_{1:T})$, $\ell_T = -\sum_i \log p(\mathbf{y}|\mathbf{h}_{1:T})$.

3.5 Adversarial Training

So far our model can be trained end-to-end with standard back-propagation by minimizing the following loss:

$$\ell = \ell_{GRAD} + \ell_S + \ell_T \tag{2}$$

Recent works have demonstrated that deep learning models are fragile to *adversarial examples* (Goodfellow et al., 2015). In computer vision, those adversarial examples can be constructed by changing a very small number of pixels, which are virtually indistinguishable to human perception (Pin-Yu et al., 2018). Recently, adversarial samples are widely incorporated into training to improve the generalization and robustness of the model, which is so-called adversarial training (AT) (Miyato et al., 2017). It emerges as a powerful regularization tool to stabilize training and prevent the model from being stuck in local minimum. In this paper, we explore AT in context of NER. To be specific, we prepare an adversarial sample by adding the original sample with a perturbation bounded by a small norm ϵ to maximize the loss function as follows:

$$\eta_{\mathbf{x}} = \arg\max_{\eta:\|\eta\|_2 \leq \epsilon} \ell(\Theta; \mathbf{x} + \eta) \tag{3}$$

where Θ is the current model parameters set. However, we cannot calculate the value of η exactly in general, because the exact optimization with respect to η is intractable in neural networks. Following the strategy in (Goodfellow et al., 2015), this value can be approximated by linearizing it as follows, $\eta_{\mathbf{x}} = \epsilon \frac{\mathbf{g}}{\|\mathbf{g}\|_2}$, where $\mathbf{g} = \nabla \ell(\Theta; \mathbf{x})$ where ϵ can be determined on the validation set. In this way, adversarial examples are generated by adding small perturbations to the inputs in the direction that most significantly increases the loss function of the model. We find such η against the current model parameterized by Θ, at each training step, and construct an adversarial example by $\mathbf{x}_{adv} = \mathbf{x} + \eta_{\mathbf{x}}$. Note that we generate this adversarial example on the word and character embedding layer, respectively, as shown in the Figure 1(b) and 1(c). Then, the classifier is trained on the mixture of original and adversarial examples to improve the generalization. To this end, we augment the loss in Eqn. 2 and define the loss function for adversarial training as:

$$\ell_{AT} = \ell(\Theta; \mathbf{x}) + \ell(\Theta; \mathbf{x}_{adv}) \tag{4}$$

where $\ell(\Theta; \mathbf{x}), \ell(\Theta; \mathbf{x}_{adv})$ represents the loss from an original example and its adversarial counterpart, respectively. Note that we present the AT in a general form for the convenience of presentation. For different samples, the loss and parameters should correspond to their counterparts. For example, for the source data with word embedding $\mathbf{w}_S$, the loss can be defined as follows, $\ell_{AT} = \ell(\Theta; \mathbf{w}_S) + \ell(\Theta; \mathbf{w}_{S,adv})$ with $\mathbf{w}_{S,adv} = \mathbf{w}_S + \eta_{\mathbf{w}_S}$ and $\ell = \ell_{GRAD} + \ell_S$. Similarly, we can compute the perturbations $\eta_{\mathbf{c}}$ for char-embedding and $\eta_{\mathbf{w}_T}$ for target word embedding.

4 Experiments

4.1 Datasets

In order to evaluate the performance of DATNet, we conduct the experiments on following widely used NER datasets: CoNLL-2003 English NER (Kim and De, 2003), CoNLL-2002 Spanish & Dutch NER (Kim, 2002), WNUT-2016 & 2017 English Twitter NER (Zeman, 2017). The statistics of these datasets are described in Table 1. We use the official split of train/validation/test sets. Different from previous works that either append the one-hot gazetteer feature to the input of

Mode	Methods		Additional Features			CoNLL Datasets		WNUT Datasets	
			POS	Gazetteers	Orthographic	Spanish	Dutch	WNUT-2016	WNUT-2017
Mono-language /domain	(Gillick et al., 2016)		×	×	×	82.59	82.84	-	-
	(Lample et al., 2016)		×	√	×	85.75	81.74	-	-
	(Partalas et al., 2016)		√	√	√	-	-	46.16	-
	(Limsopatham and Collier, 2016)		×	×	√	-	-	52.41	-
	(Lin et al., 2017a)		√	√	×	-	-	-	40.42
	Our Base Model	Best	×	×	×	85.53	85.55	44.96	35.20
		Mean & Std				85.35±0.15	85.24±0.21	44.37±0.31	34.67±0.34
Cross-language /domain	(Yang et al., 2017)		×	√	×	85.77	85.19	-	-
	(Lin et al., 2018)		×	√	×	85.88	86.55	-	-
	(Feng et al., 2018)		√	×	×	86.42	**88.39**	-	-
	(von Däniken and Cieliebak, 2017)		×	√	×	-	-	-	40.78
	(Aguilar et al., 2017)		√	×	√	-	-	-	41.86
	DATNet-P	Best	×	×	×	**88.16**	88.32	50.85	41.12
		Mean & Std				87.89±0.18	88.09±0.13	50.41±0.32	40.52±0.38
	DATNet-F	Best	×	×	×	87.04	87.77	**53.43**	**42.83**
		Mean & Std				86.79±0.20	87.52±0.19	53.03±0.24	42.32±0.32

Table 2: Comparison with State-of-the-art Results in CoNLL and WNUT datasets (F1-score).

Tasks	CoNLL-2002 Spanish NER						WNUT-2016 Twitter NER					
# Target train sentences	10	50	100	200	500	1000	10	50	100	200	500	1000
Base	21.53	42.18	48.35	63.66	68.83	76.69	3.80	14.07	17.99	26.20	31.78	36.99
+ AT	19.23	41.01	50.46	64.83	70.85	77.91	4.34	16.87	18.43	26.32	35.68	41.69
+ P-Transfer	29.78	61.09	64.78	66.54	72.94	78.49	7.71	16.17	20.43	29.20	34.90	41.20
+ F-Transfer	39.72	63.00	63.36	66.39	72.88	78.04	15.26	20.04	26.60	32.22	38.35	44.81
DATNet-P	39.52	62.57	64.05	**68.95**	**75.19**	**79.46**	9.94	17.09	25.39	30.71	36.05	42.30
DATNet-F	**44.52**	**63.89**	**66.67**	68.35	74.24	78.56	**17.14**	**22.59**	**28.41**	**32.48**	**39.20**	**45.25**

Table 3: Experiments on Extremely Low Resource (F1-score).

the CRF layer (Collobert et al., 2011; Chiu and Nichols, 2016; Yang et al., 2017) or introduce the orthographic feature as additional input for learning social media NER in tweets (Partalas et al., 2016; Limsopatham and Collier, 2016; Aguilar et al., 2017), we do *not* use hand-crafted features and only words and characters are considered as the inputs. Our goal is to study the effects of transferring knowledge from high-resource dataset to low-resource dataset. To be noted, we used only training set for model training for all datasets except the WNUT-2016 NER dataset. Since in this dataset, all the previous studies merged the training set and validation set together for training. Specifically, we use CoNLL-2003 English NER dataset as high-resource (i.e., source) for all the experiments, CoNLL-2002 and WNUT datasets as low-resource (i.e., target) in cross-language and cross-domain NER settings, respectively.

4.2 Experimental Setup

We use 50-dimensional publicly available pre-trained word embeddings for English, Spanish and Dutch of CoNLL and WNUT datasets in our experiments, which are trained by word2vec on the corresponding Wikipedia articles (Lin et al., 2018), and the 30-dimensional randomly initialized character embeddings are used for all the datasets. We set the filters as 20 for char-level CNN and the dimension of hidden states of the word-level LSTM as 200 for both base model and DATNet-F. For DATNet-P, we set 100 for source, share, and target LSTMs, respectively. Parameters optimization is performed by Adam (Kingma and Ba, 2015) with gradient clipping of 5.0 and learning rate decay strategy. We set the initial learning rate of $\beta_0 = 0.001$ for all experiments. At each epoch t, learning rate β_t is updated using $\beta_t = \beta_0/(1 + \rho \times t)$, where ρ is decay rate with 0.05. To reduce over-fitting, we apply Dropout (Srivastava et al., 2014) to the embedding layer and the output of the LSTM layer, respectively.

4.3 Comparison with State-of-The-Art Results

In this section, we compare our approach with state-of-the-art methods on CoNLL and WNUT benchmark datasets. Note that our models do not use any additional large-scale language resources, so we do not consider the language models (Peters et al., 2018; Radford, 2018; Devlin et al., 2018) for fair comparison. In the experiment, we exploit all the source data (i.e., CoNLL-2003 English NER) and target data to improve performance of target tasks. The averaged results with standard deviation over 10 repetitive runs are summarized in Table 2, and we also report the best results on each task for fair comparison with other SOTA methods. From results, we observe that incorporating the additional resource is helpful to improve performance. DATNet-P achieves the highest F1 score on CoNLL-2002 Spanish and sec-

CoNLL-2002 Spanish NER				WNUT-2016 Twitter NER			
Model	F1-score	Model	F1-score	Model	F1-score	Model	F1-score
Base	85.35	+AT	86.12	Base	44.37	+AT	47.41
+P-T (no AD)	86.15	+AT +P-T (no AD)	86.90	+P-T (no AD)	47.66	+AT +P-T (no AD)	48.44
+F-T (no AD)	85.46	+AT +F-T (no AD)	86.17	+F-T (no AD)	49.79	+AT +F-T (no AD)	50.93
+P-T (AD)	86.32	+AT +P-T (AD)	87.19	+P-T (AD)	48.14	+AT +P-T (AD)	49.41
+F-T (AD)	85.58	+AT +F-T (AD)	86.38	+F-T (AD)	50.48	+AT +F-T (AD)	51.84
+P-T (GRAD)	86.93	+AT +P-T (GRAD) (***DATNet-P***)	**88.16**	+P-T (GRAD)	48.91	+AT +P-T (GRAD) (***DATNet-P***)	50.85
+F-T (GRAD)	85.91	+AT +F-T (GRAD) (***DATNet-F***)	87.04	+F-T (GRAD)	51.31	+AT +F-T (GRAD) (***DATNet-F***)	**53.43**

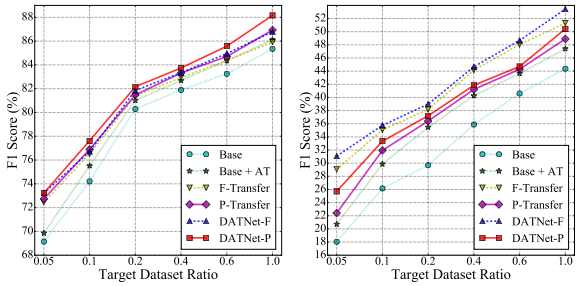
* AT: Adversarial Training; P-T: P-Transfer; F-T: F-Transfer; AD: Adversarial Discriminator; GRAD: Generalized Resource-Adversarial Discriminator.

Table 4: Quantitative Performance Comparison between Models with Different Components.

α	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8
Ratio	CoNLL-2002 Spanish NER														
$\rho = 0.1$	78.37	78.63	**78.70**	78.32	77.96	77.92	77.88	77.78	77.85	77.90	77.65	77.57	77.38	77.49	77.29
$\rho = 0.2$	80.99	81.71	**82.18**	81.57	81.53	81.55	81.44	81.25	81.32	81.16	81.02	81.16	80.63	80.79	80.54
$\rho = 0.4$	83.76	83.73	84.18	**84.48**	84.26	84.12	83.54	83.40	83.52	84.18	83.42	83.47	83.28	83.33	83.19
$\rho = 0.6$	85.18	85.24	85.85	85.68	85.84	**86.10**	85.71	85.74	85.42	85.60	85.20	85.40	85.26	85.24	84.98

Table 5: Analysis of Discriminator Weight α in GRAD with Varying Data Ratio ρ (F1-score).

ond F1 score on CoNLL-2002 Dutch dataset while DATNet-F beats others on WNUT-2016 and 2017 Twitter datasets. Different from other SOTA models, DATNets do *not* use any addition features[1].



(a) CoNLL-2002 Spanish (b) WNUT-2016 Twitter

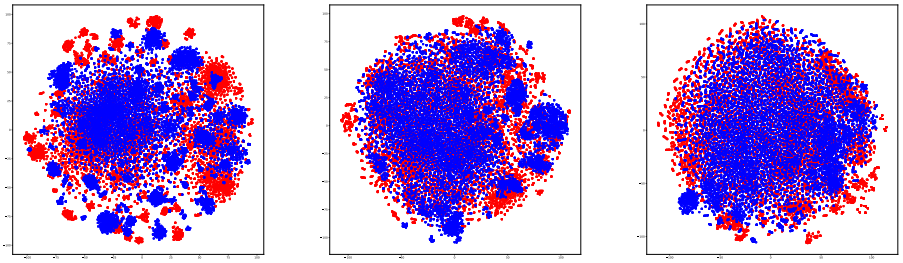
Figure 2: Comparison with Different Target Data Ratio, where AT stands for adversarial training, F(P)-Transfer denotes the DATNet-F(P) without AT.

4.4 Transfer Learning Performance

In this section, we investigate on improvements with transfer learning under multiple low-resource settings with partial target data. To simulate a low-resource setting, we randomly select subsets of target data with varying data ratio at 0.05, 0.1, 0.2, 0.4, 0.6, and 1.0. The results for cross-language and cross-domain transfer are shown in Figure 2(a) and 2(b), respectively, where we compare the results with each part of DATNet under various data ratios. From those figures, we have the following observations: 1) both adversarial training and adversarial discriminator in DATNet consistently contribute to the performance improvement; 2) the transfer learning component in the DATNet consistently improves over the base model results and the improvement margin is more substantial when the target data ratio is lower. For example, when the data ratio is 0.05, DATNet-P model outperforms the base model by more than 4% absolutely in F1-score on Spanish NER and DATNet-F model improves around 13% absolutely in F1-score compared to base model on WNUT-2016 NER.

In the second experiment, we further investigate DATNet on the extremely low resource cases, e.g., the number of training target sentences is 10, 50, 100, 200, 500 and 1,000. The setting is quite challenging and fewer previous works have studied before. The results are summarized in Table 3. We have two interesting observations: 1) DATNet-F outperforms DATNet-P on cross-language transfer when the target resource is extremely low, however, this situation is reversed when the target dataset size is large enough (here for this specific dataset, the threshold is 100 sentences); 2) DATNet-F is always superior to DATNet-P on cross-domain transfer. For the first observation, DATNet-F with more shared hidden units is more efficient to transfer knowledge than DATNet-P when data size is extremely small. For the second observation, because cross-domain transfer are in the same language, more knowledge is common between source and target domains, requiring more shared hidden features to carry with these knowledge compared to cross-language transfer. Therefore, for cross-language transfer with extremely low resource and cross-domain transfer, we suggest using DATNet-F to achieve better performance. As for cross-language transfer with relatively more training data, DATNet-P is preferred.

[1] We are not sure whether (Feng et al., 2018) has incorporated the validation set into training. And if we merge training and validation sets, we can push the F1 score to **88.71**.



(a) no AD (b) AD (c) GRAD

Figure 3: The visualization of extracted features from shared bidirectional-LSTM layer. The left, middle, and right figures show the results when no Adversarial Discriminator (AD), AD, and GRAD is performed, respectively. Red points denotes source CoNLL-2003 English examples, blue points denotes target CoNLL-2002 Spanish examples.

$\epsilon_{\mathbf{w}_T}$	1.0	3.0	5.0	7.0	9.0
Ratio	CoNLL-2002 Spanish NER				
$\rho = 0.1$	75.90	76.23	77.38	77.77	**78.13**
$\rho = 0.2$	81.54	81.65	81.32	**81.81**	81.68
$\rho = 0.4$	83.62	83.83	83.43	**83.99**	83.40
$\rho = 0.6$	84.44	84.47	**84.72**	84.04	84.05

Table 6: Analysis of Maximum Perturbation $\epsilon_{\mathbf{w}_T}$ in AT with Varying Data Ratio ρ (F1-score).

4.5 Ablation Study of DATNet

In the proposed DATNet, both GRAD and AT play important roles in low resource NER. In this experiment, we further investigate how GRAD and AT help to transfer knowledge across language/domain. In the first experiment, we used t-SNE (Maaten and Hinton, 2008) to visualize the feature distribution of BiLSTM outputs without AD, with normal AD (GRAD without considering data imbalance), and with the proposed GRAD in Figure 3. From this figure, we can see that GRAD in DATNet makes the distribution of extracted features from source and target datasets much more similar by considering data imbalance, which indicates that the outputs are resource-invariant.

To better understand the working mechanism, Table 4 further reports the quantitative performance comparison between models with different components. We observe that GRAD shows the stable superiority over the normal AD regardless of other components. There is not always a winner between DATNet-P and DATNet-F on different settings. DATNet-P architecture is more suitable to cross-language transfer while DATNet-F is more suitable to cross-domain transfer.

From the previous results, we know that AT helps enhance the overall performance by adding perturbations to inputs with the limit of $\epsilon = 5$, i.e., $\|\eta\|_2 \leq 5$. In this experiment, we further investigate how target perturbation $\epsilon_{\mathbf{w}_T}$ with fixed source perturbation $\epsilon_{\mathbf{w}_S} = 5$ in AT affects knowledge transfer and the results on Spanish NER are summarized in Table 6. The results generally indicate that less training data require a larger ϵ to prevent over-fitting, which further validates the necessity of AT in the case of low resource data.

Finally, we analyze the discriminator weight α in GRAD and results are summarized in Table 5. From the results, it is interesting to find that α is directly proportional to the data ratio ρ, basically, which means that more target training data requires larger α (i.e., smaller $1-\alpha$ to reduce training emphasis on the target domain) to achieve better performance.

5 Conclusion

In this paper we develop a transfer learning model DATNet for low-resource NER, which aims at addressing representation difference and resource data imbalance problems. We introduce two variants, DATNet-F and DATNet-P, which can be chosen according to cross-language/domain user case and target dataset size. To improve model generalization, we propose dual adversarial learning strategies, i.e., AT and GRAD. Extensive experiments show the superiority of DATNet over existing models and it achieves significant improvements on CoNLL and WNUT NER benchmark datasets.

Acknowledgments

This paper is supported by the Singapore Government's Research, Innovation and Enterprise 2020 Plan, Advanced Manufacturing and Engineering domain (Programmatic Grant No. A1687b0033, A18A1b0045) and the Agency for Science, Technology and Research, under the AME Programmatic Funding Scheme (Project No. A18A2b0046, A1718g0048).

References

Oliver Adams, Adam Makarucha, Graham Neubig, Steven Bird, and Trevor Cohn. 2017. Cross-lingual word embeddings for low-resource language modeling. In *EACL*, pages 937–947. Association for Computational Linguistics.

Gustavo Aguilar, Suraj Maharjan, Adrian Pastor López Monroy, and Thamar Solorio. 2017. A multi-task approach for named entity recognition in social media data. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 148–153.

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

Rami Al-Rfou', Vivek Kulkarni, Bryan Perozzi, and Steven Skiena. 2015. Polyglot-ner: Massive multilingual named entity recognition. In *SDM*.

Dzmitry Bahdanau, KyungHyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, and Shengping Liu. 2018. Adversarial transfer learning for chinese named entity recognition with self-attention mechanism. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 182–192.

Yufeng Chen, Chengqing Zong, and Keh-Yih Su. 2010. On jointly recognizing and aligning bilingual named entities. In *ACL*, pages 631–639.

Jason Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *JMLR*, pages 2493–2537.

Ryan Cotterell and Kevin Duh. 2017. Low-resource named entity recognition with cross-lingual, character-level neural conditional random fields. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 91–96. Asian Federation of Natural Language Processing.

Pius von Däniken and Mark Cieliebak. 2017. Transfer learning and sentence level features for named entity recognition on tweets. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 166–171.

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

Meng Fang and Trevor Cohn. 2017. Model transfer for tagging low-resource languages using a bilingual dictionary. In *ACL*, pages 587–593.

Xiaocheng Feng, Xiachong Feng, Bing Qin, Zhangyin Feng, and Ting Liu. 2018. Improving low resource named entity recognition using cross-lingual knowledge transfer. In *IJCAI*, pages 4071–4077.

Dan Gillick, Cliff Brunk, Oriol Vinyals, and Amarnag Subramanya. 2016. Multilingual language processing from bytes. In *NAACL HLT*, pages 1296–1306.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *ICLR*.

Tao Gui, Qi Zhang, Haoran Huang, Minlong Peng, and Xuanjing Huang. 2017. Part-of-speech tagging for twitter with adversarial neural networks. In *EMNLP*, pages 2411–2420.

Kazuma Hashimoto, caiming xiong, Yoshimasa Tsuruoka, and Richard Socher. 2017. A joint many-task model: Growing a neural network for multiple nlp tasks. In *EMNLP*, pages 1923–1933.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *CoRR*, abs/1508.01991.

Joo-Kyung Kim, Young-Bum Kim, Ruhi Sarikaya, and Eric Fosler-Lussier. 2017. Cross-lingual transfer learning for pos tagging without cross-lingual resources. In *EMNLP*, pages 2832–2838.

Sang Erik F. Tjong Kim. 2002. Introduction to the conll-2002 shared task: Language-independent named entity recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*.

Sang Erik F. Tjong Kim and Meulder Fien De. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *NAACL HLT*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *NAACL HLT*, pages 260–270.

Qi Li, Haibo Li, Heng Ji, Wen Wang, Jing Zheng, and Fei Huang. 2012. Joint bilingual name tagging for parallel corpora. In *CIKM '12*, pages 1727–1731.

Nut Limsopatham and Nigel Collier. 2016. Bidirectional lstm for named entity recognition in twitter messages. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pages 145–152.

Bill Y. Lin, Frank Xu, Zhiyi Luo, and Kenny Zhu. 2017a. Multi-channel bilstm-crf model for emerging named entity recognition in social media. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 160–165.

T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar. 2017b. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*.

Ying Lin, Shengqi Yang, Veselin Stoyanov, and Heng Ji. 2018. A multi-lingual multi-task architecture for low-resource sequence labeling. In *ACL*.

L. Liu, J. Shang, F. Xu, X. Ren, H. Gui, J. Peng, and J. Han. 2018. Empower sequence labeling with task-aware neural language model. In *AAAI*.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial multi-task learning for text classification. In *ACL*.

Gang Luo, Xiaojiang Huang, Chin-Yew Lin, and Zaiqing Nie. 2015. Joint entity recognition and disambiguation. In *EMNLP*, pages 879–888.

Minh-Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2016. Multi-task sequence to sequence learning. In *ICLR*.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *ACL*, pages 1064–1074.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *JMLR*, 9:2579–2605.

Mónica Marrero, Julián Urbano, Sonia Sánchez-Cuadrado, Jorge Morato, and Juan Miguel Gómez-Berbís. 2013. Named entity recognition: Fallacies, challenges and opportunities. *Computer Standards & Interfaces*, (5):482–489.

Stephen Mayhew, Chen-Tse Tsai, and Dan Roth. 2017. Cheap translation for cross-lingual named entity recognition. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2536–2545. Association for Computational Linguistics.

Takeru Miyato, Andrew M Dai, and Ian Goodfellow. 2017. Adversarial training methods for semi-supervised text classification. In *ICLR*.

Jian Ni, Georgiana Dinu, and Radu Florian. 2017. Weakly supervised cross-lingual named entity recognition via effective annotation and representation projection. In *ACL*, pages 1470–1480.

Jian Ni and Radu Florian. 2016. Improving multilingual named entity recognition with wikipedia entity type mapping. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1275–1284.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958. Association for Computational Linguistics.

Ioannis Partalas, Cédric Lopez, Nadia Derbas, and Ruslan Kalitvianski. 2016. Learning to search for recognizing named entities in twitter. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pages 171–177.

Alexandre Passos, Vineet Kumar, and Andrew McCallum. 2014. Lexicon infused phrase embeddings for named entity resolution. *arXiv preprint arXiv:1404.5367*.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL HLT*, pages 2227–2237. Association for Computational Linguistics.

Chen Pin-Yu, Sharma Yash, Zhang Huan, Yi Jinfeng, and Cho-Jui Hsieh. 2018. Ead: Elastic-net attacks to deep neural networks via adversarial examples. In *AAAI*.

Alec Radford. 2018. Improving language understanding by generative pre-training.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, pages 147–155.

Marek Rei. 2017. Semi-supervised multitask learning for sequence labeling. In *ACL*, pages 2121–2130.

Marek Rei and Anders Søgaard. 2018. Zero-shot sequence labeling: Transferring knowledge from sentences to tokens. In *NAACL HLT*, pages 293–302.

Nils Reimers and Iryna Gurevych. 2017. Reporting score distributions makes a difference: Performance study of lstm-networks for sequence tagging. In *EMNLP*, pages 338–348.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, pages 1929–1958.

Christian Szegedy, Wojciech Zaremba, Dumitru Erhan Ian Goodfellow Ilya Sutskever, Joan Bruna, and Rob Fergus. 2014. Intriguing properties of neural networks. In *ICLR*.

Zhilin Yang, Ruslan Salakhutdinov, and William W. Cohen. 2016. Multi-task cross-lingual sequence tagging from scratch. *CoRR*, abs/1603.06270.

Zhilin Yang, Ruslan Salakhutdinov, and William W. Cohen. 2017. Transfer learning for sequence tagging with hierarchical recurrent networks. In *ICLR*.

David Yarowsky, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the first international conference on Human language technology research*, pages 1–8.

Michihiro Yasunaga, Jungo Kasai, and Dragomir Radev. 2018. Robust multilingual part-of-speech tagging via adversarial training. In *NAACL HLT*, pages 976–986.

Daniel et al. Zeman. 2017. Conll 2017 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19.

Boliang Zhang, Xiaoman Pan, Tianlu Wang, Ashish Vaswani, Heng Ji, Kevin Knight, and Daniel Marcu. 2016. Name tagging for low-resource incident languages based on expectation-driven learning. In *NAACL HLT*, pages 249–259.

Joey Tianyi Zhou, Meng Fang, Hao Zhang, Chen Gong, Xi Peng, Zhiguo Cao, and Rick Siow Mong Goh. Learning with annotation of various degrees. *IEEE Transactions on Neural Networks and Learning Systems*.

Joey Tianyi Zhou, Hao Zhang, Di Jin, Xi Peng, Yang Xiao, and Zhiguo Cao. 2019. Roseq: Robust sequence labeling. *IEEE Transactions on Neural Networks and Learning Systems*, PP:1–11.

Andrej Zukov Gregoric, Yoram Bachrach, and Sam Coope. 2018. Named entity recognition with parallel recurrent neural networks. In *ACL*, pages 69–74.