Paper Implementation - Zero-Resource Cross-Domain Named Entity Recognition

Siddharth Sundararajan
ss3177

Nitin Reddy Karolla
nrk60

Abstract

Existing models for cross-domain named entity recognition (NER) primarily rely on a lot of unlabeled corpus or labeled NER training data in target domains. However, this process is expensive and time-consuming. Therefore, we consider the idea of a cross-domain NER model that does not use any external resources. In this work, we implement the main idea presented in the paper - Zero-Resource Cross-Domain Named Entity Recognition (Liu et al., 2020) whose code is not publicly available. The aim is to build a cross-domain NER model that does not use any external resources. We evaluate using F-1 score as the metric. The experiments are conducted on CoNLL-2003 English NER data (Sang and Meulder, 2003) (source domain) and SciTech News data (Jia et al., 2019) (target domain). Our experimental results show that we are able to match the results shown in the paper (which outperforms strong unsupervised cross-domain sequence labeling models). As an extension, we modified the architecture and were able to outperform state-of-the-art-results.

1 Introduction

Named entity recognition (NER) is an old problem in text understanding and information extraction in the natual language processing domain. It has various applications that range from recommendation systems to automating summaries of resumes. Recently, supervised learning techniques have shown their effectiveness in detecting named entities (Ma and Hovy, 2016; Chiu and Nichols, 2016; Winata et al., 2019). However, there is a vast performance drop for low-resource target domains when massive training data are absent. This problem can be solved by utilizing the NER knowledge learned from high resource domains and then adapting it to low-resource domains, which is defined as cross-domain NER.

Cross-domain NER is a challenging task and hence, most of the existing methods use supervised learning (Yang et al., 2017; Lin and Lu, 2018). But, labeled data may not be always present in target domains and hence, the alternative approach is unsupervised learning (Jia et al., 2019). This alternative still requires an external unlabeled data corpus in source and target domains, which is generally difficult to obtain, especially low resource target domains. Therefore, we consider the problem of unsupervised zero-resource cross-domain adaptation of NER which only utilizes the NER training samples in a single source domain.

To overcome the shortcomings mentioned, the first proposed step is to conduct Multi-Task Learning (MTL) by adding an objective function that detects whether tokens are named entities or not. The aim of this objective function will help to learn general representations of named entities. Generally, there are cases where different entity categories could have a similar or the same context. For example, in the sentence "Arafat subsequently cancelled a meeting between Israeli and PLO officials," the person entity "Arafat", can be replaced with an organization entity within the same context. This confusion can be tackled by introducing the Mixture of Entity Experts (MoEE) framework that is combined with the BiLSTM-CRF structure to solve the original NER task.

The proposed model outperforms current strong unsupervised cross-domain sequence tagging approaches and is able to reach comparable results to the state-of-the-art unsupervised method that utilizes extensive resources. The code is available at https://github.com/Siddharthss500/zero-resource.

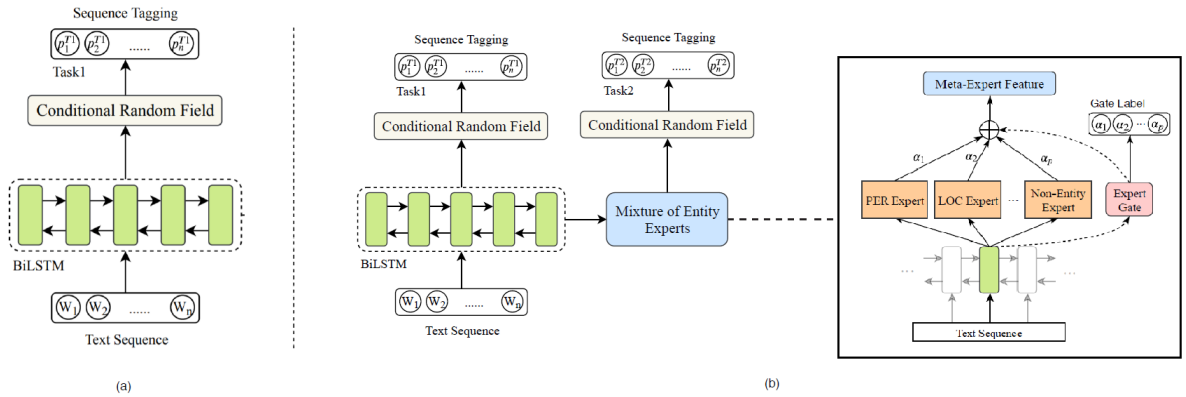

Figure 1: Model architecture (a) BiLSTM+CRF (Baseline) and (b) Baseline + Multi Task Learning (MTL) and Mixture of Entity Experts (MoEE) module.

2 Related Work

In cross-domain NER most of the existing work has been in the supervised setting, where both source and target domains have labeled data (Daumé, 2007; Obeidat et al., 2016; Yang et al., 2017; Lee et al., 2018; Yang et al., 2017) which are jointly trained models on the source and target domain with shared parameters. Adding adaptation layers on top of the existing models were done by (Lin and Lu, 2018). Label-aware feature representations for NER adaption were introduced by (Wang et al., 2018). However, only (Jia et al., 2019) has focused on the unsupervised setting of cross-domain NER. In (Jia et al., 2019), however, external unlabeled data corpora resources are required for training. This limitation motivated the authors to develop a model that doesn't need any external resources.

Tackling the low-resource scenario where there are zero or minimal existing resources has always been an interesting yet challenging task (Xie et al., 2018; Lee et al., 2019; Liu et al., 2019). Instead of utilizing large amounts of bilingual resources, (Liu et al., 2019) only utilized a few word pairs for zero-shot cross-lingual dialogue systems. Unsupervised machine translation approaches (Lample et al., 2018; Artetxe et al., 2018) have also been introduced to circumvents the need of parallel data. (Bapna et al., 2017) and (Shah et al., 2019) proposed to do cross-domain slot filling with minimal resources. The authors claim that they are the first to propose methods on cross-domain adaptation for NER with zero external resources.

3 Model

As seen in Figure 1, the first part is the baseline model - bidirectional LSTM and Conditional Random Field (CRF) into a BiLSTM-CRF structure (Lample et al., 2016). The second part of the figure is the proposed model - BiLSTM-CRF along with MTL and MoEE modules.

3.1 Multi Task Learning

Due to the large number of named entities across various domains, there is a problem of recognizing named entities. To tackle this problem, an objective function is introduced as $Task_1$ that predicts whether tokens are named entities or not. This is achieved by treating it as a three-way classification problem (B-I-O) for each token. The named entity token - O is treated as a non-entity token. Note that $Task_2$ is the original NER task. Let us assume that $X = [w_1, w_2, \ldots, w_n]$ is an input text sequence where $w_1, w_2, \ldots, w_n \epsilon V$, then MTL can be formulated as,

$$[h_1, \ldots, h_n] = BiLSTM([w_1, \ldots, w_n])$$
$$[p_1^{T_1}, \ldots, p_n^{T_1}] = CRF_1([h_1, \ldots, h_n])$$
$$[p_1^{T_2}, \ldots, p_n^{T_2}] = CRF_2([h_1, \ldots, h_n])$$

where, CRF_1 and CRF_2 denote the CRF layers for $Task_1$ (learn the objective function to predict whether tokens are named entities or not) and $Task_2$ (to predict a concrete category for each token or NER task). respectively, and $[p_1^{T_1}, p_2^{T_1}, \ldots, p_n^{T_1}]$ and $[p_1^{T_2}, p_2^{T_2}, \ldots, p_n^{T_2}]$ represent the corresponding predictions.

3.2 Mixture of Entity Experts

NER models can overfit to the source domain entities and lose the generalization to the target domain. To solve this problem, the MoEE module is introduced, which combines representations generated by experts to produce the final prediction.

MoEE is combined with the BiLSTM-CRF framework for the NER task ($Task_2$). Based on the hidden state of each token, it first generates a feature for each entity expert as well as the confidence score for each expert. It then combines the features of the entity experts to generate meta-expert feature for each token. This can be formulated as,

$$[expt_i^1, \ldots, expt_i^E] = [L^1(h_i), \ldots, L^E(h_i)]$$
$$[\alpha_1, \ldots, \alpha_E] = Softmax(Linear(h_i))$$
$$m_i = \sum_{a=1}^{E} \alpha_a * expt_i^a$$

where m_i is the meta-expert feature for the i-th hidden state of the BiLSTM, expt is the feature generated from the expert (expert gate consists of a linear layer followed by a softmax layer), and L denotes the linear layer.

Note that by adding the MoEE module along with the baseline (BiLSTM-CRF) and MTL module, the final proposed model can be formulated as,

$$[h_1, \ldots, h_n] = BiLSTM([w_1, \ldots, w_n])$$
$$[p_1^{T_1}, \ldots, p_n^{T_1}] = CRF_1([h_1, \ldots, h_n])$$
$$[m_1, \ldots, m_n] = MoEE([h_1, \ldots, h_n])$$
$$[p_1^{T_2}, \ldots, p_n^{T_2}] = CRF_2([m_1, \ldots, m_n])$$

3.3 Optimization

Tasks $Task_1$, $Task_2$ and $forget\ gate$ are optimized with cross entropy losses. They losses for each are,

$$L^{task_1} = \sum_{j=1}^{J} \sum_{k=1}^{|Y_j|} -\log(p_{jk}^{T_1} \cdot (y_{jk}^{T_1})^T)$$
$$L^{task_2} = \sum_{j=1}^{J} \sum_{k=1}^{|Y_j|} -\log(p_{jk}^{T_2} \cdot (y_{jk}^{T_2})^T)$$
$$L^{gate} = \sum_{j=1}^{J} \sum_{k=1}^{|Y_j|} -\log(p_{jk}^{gate} \cdot (y_{jk}^{gate})^T)$$

where, J and $|Y_j|$ denote the number of training data and the length of the tokens for each training sample, respectively; p_{jk} and y_{jk} denote the predictions and labels for each token, respectively; and the superscripts of p_{jk} and y_{jk} represent the tasks. Hence, the final objective function is to minimize the sum of all the aforementioned loss functions.

3.4 Extended Work

We implemented the models, baseline - BiLSTM-CRF, BiLSTM-CRF along with MTL and MoEE modules. We extended the work further by modifying the architectures. In the first approach (BiLSTM-CRF w/ MTL (Separate)), we added an extra BiLSTM layer where $Task_1$ and $Task_2$ had their own BiLSTM layers (with their inputs being a sentence embedding). In the second implementation, Mod1, the loss function was calculated from three CRF modules during training, one from entity ($Task_1$), two from NER ($Task_2$) and three from MoEE. Note that the predictions were made using just MoEE module. In our last implementation, Mod2, it was similar to Mod1 but the prediction were made using the $Task_2$ module.

4 Experiments

4.1 Data

We take the CoNLL-2003 English NER data (Sang and Meulder, 2003) containing 14K/3.5K/3.7K samples for the training/validation/test sets as our source domain. We take the dataset containing 2K sentences from SciTech News provided by (Jia et al., 2019) as our target domain.

Note that there was no pre-processing done on the data. As the target domain has more entities (IOBES format) as compared to the IOB format (in the source domain), we use a function to convert the entities from IOBES format to IOB format.

4.2 Experimental Setup

Embeddings We use Fast-Text word embeddings (Bojanowski et al., 2017) to represent the data. Note that entity names in the target domain are most likely to be out-of-vocabulary (OOV) words as they don't usually exist in the source domain training set. This problem was tackled by using the subword information in the fast-text embeddings. Note that we used two versions - with fine tuned fast-text and pre-trained fast-text embeddings.

Model	Author's Results		Our Results			
	FastText-Pretrained		FastText-Fine tuned		FastText-Pretrained	
	unfreeze	freeze	unfreeze	freeze	unfreeze	freeze
BiLSTM-CRF	63.18	67.89	63.38	67.64	64.41	67.56
BiLSTM-CRF w/ MTL	64.62	69.58	66.5	64.7	68.84	68.89
BiLSTM-CRF w/ MTL (Separate)	-	-	66.3	66.1	68.84	68.89
BiLSTM-CRF w/ MoEE	**65.24**	69.25	62.3	63.19	68.7	67.94
BiLSTM-CRF w/ MTL and MoEE	64.88	**70.04**	49.98	65.48	67.24	68.33
Mod1	-	-	66.27	65.57	**68.31**	**70.37**
Mod2	-	-	45.66	61.91	65.33	69.36

Table 1: F-1 scores on the target domain (multiplied by 100)

Baselines The baseline model is BiLSTM (Lample et al., 2016; Huang et al., 2015). We compare the baseline with BiLSTMCRF+MTL, BiLSTM-CRF+MoEE and the three implementations.

4.3 Results and Discussion

In Table 1, we compare the author's results against our own. We have two implementations - trained our models by fine tuning the fasttext embeddings and by using the pretrained fasttext embeddings. Freeze and unfreeze settings for fasttext embeddings were also used while training. In addition to the author's work we have made few changes to the architecture, mainly Mod1 and Mod2.

By looking at the results, we observe that we are able to replicate the author's results with a small difference. Note that adding an extra BiLSTM layer on MTL module does not make any difference to the F1-score. Additionally, our modification, Mod1, is able to outperform the author's results by a small margin while using the pretrained fasttext embeddings. Mod2 performs well but does not exceed the results of Mod1. This might be due to the fact that it is unable to capture the entity expert module well.

Generally, we see that the fine tuned freezed Fasttext embeddings bring better performance than unfreezed ones. Also, by fine tuning fasttext on the source domain, we achieve a low F1-score. We conjecture that the embeddings could be overfitting to the source domain if we unfreeze them in the training. Note that we were able to verify this by predicting on NER (same domain) and achieving F1-scores of around 0.95.

5 Conclusion and Future Work

In this report, we implement the main idea presented in the paper - Zero-resource crossdomain Named Entity Recognition. The main framework for the named entity recognition task, consists of multi-task learning and Mixture of Entity Experts modules. The former learns the general representations of named entities to cope with the model's inability to recognize named entities, while the latter learns to combine the representations of different entity experts, which are based on the BiLSTM hidden states. Our experimental results showed that we were able to replicate the original results. Our extensions of altering the loss function and training showed that our model is able to outperform author's results by a small margin.

As part of future work, we plan to add and try out different attention layers and also incorporate Bert.

6 Acknowledgement

This work is part of our NLP coursework (CS533) at Rutgers University, The State University of New Jersey. Our course instructor and guide was Professor Karl Stratos. We would like to thank him for exposing us to various topics in NLP research and literature. In addition, we would like to thank the authors of the paper and fellow peers for their help.

References

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised neural machine translation. *ArXiv*, abs/1710.11041.

Ankur Bapna, Gökhan Tür, Dilek Z. Hakkani-Tür, and Larry Heck. 2017. Towards zero-shot frame semantic parsing for domain scaling. In *INTERSPEECH*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

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

Hal Daumé. 2007. Frustratingly easy domain adaptation. *ArXiv*, abs/0907.1815.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging.

Chen Jia, Liang Xiao, and Yue Zhang. 2019. Cross-domain ner using cross-domain language modeling. In *ACL*.

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

Guillaume Lample, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Unsupervised machine translation using monolingual corpora only. *ArXiv*, abs/1711.00043.

Ji Young Lee, Franck Dernoncourt, and Peter Szolovits. 2018. Transfer learning for named-entity recognition with neural networks. *ArXiv*, abs/1705.06273.

Nayeon Lee, Zihan Liu, and Pascale Fung. 2019. Team yeon-zi at semeval-2019 task 4: Hyperpartisan news detection by de-noising weakly-labeled data. In *SemEval@NAACL-HLT*.

Bill Yuchen Lin and Wei Lu. 2018. Neural adaptation layers for cross-domain named entity recognition. *ArXiv*, abs/1810.06368.

Zihan Liu, Jamin Shin, Yuning Xu, Genta Indra Winata, Peng Xu, Andrea Madotto, and Pascale Fung. 2019. Zero-shot cross-lingual dialogue systems with transferable latent variables. In *EMNLP/IJCNLP*.

Zihan Liu, Genta Indra Winata, and Pascale Fung. 2020. Zero-resource cross-domain named entity recognition. *ArXiv*, abs/2002.05923.

Xuezhe Ma and Eduard H. Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *ArXiv*, abs/1603.01354.

Rasha Obeidat, Xiaoli Z. Fern, and Prasad Tadepalli. 2016. Label embedding approach for transfer learning. In *ICBO/BioCreative*.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *ArXiv*, cs.CL/0306050.

Darsh J. Shah, Raghav Gupta, Amir A. Fayazi, and Dilek Z. Hakkani-Tür. 2019. Robust zero-shot cross-domain slot filling with example values. In *ACL*.

Zhenghui Wang, Yanru Qu, Liheng Chen, Jian Shen, Weinan Zhang, Shaodian Zhang, Yimei Gao, Gen Gu, Ken Chen, and Yong Yu. 2018. Label-aware double transfer learning for cross-specialty medical named entity recognition. In *NAACL-HLT*.

Genta Indra Winata, Zhaojiang Lin, Jamin Shin, Zihan Liu, and Pascale Fung. 2019. Hierarchical meta-embeddings for code-switching named entity recognition. In *EMNLP/IJCNLP*.

Jiateng Xie, Zhilin Yang, Graham Neubig, Noah A. Smith, and Jaime G. Carbonell. 2018. Neural cross-lingual named entity recognition with minimal resources. In *EMNLP*.

Zhilin Yang, Ruslan Salakhutdinov, and William W. Cohen. 2017. Transfer learning for sequence tagging with hierarchical recurrent networks. *ArXiv*, abs/1703.06345.