Textual Entailment with Dynamic Contrastive Learning for Zero-shot NER

Anonymous ACL submission

Abstract

In this paper, we study the problem of zero-shot NER, which aims at building a Named Entity Recognition (NER) system from scratch. It needs to identify the entities in the given sentences when we have zero token-level annotations for training. Previous works usually use sequential labeling models to solve the NER task and obtain weakly labeled data from entity dictionaries in the zero-shot setting. However, these labeled data are quite noisy since we need the labels for each token and the entity coverage of the dictionaries is limited. Here we propose to formulate the NER task as a Textual Entailment problem and solve the task via Textual Entailment with Dynamic Contrastive Learning (TEDC). TEDC not only alleviates the noisy labeling issue, but also transfers the knowledge from pre-trained textual entailment models. Additionally, the dynamic contrastive learning framework contrasts the entities and non-entities in the same sentence and improves the model's discrimination ability. Experiments on two datasets show that TEDC can achieve state-of-the-art performance on the task of zero-shot NER.

1 Introduction

Named Entity Recognition (NER) (Nadeau and Sekine, 2007) is a basic and important task in Natural Language Processing (NLP). It aims at recognizing named entities in a given sentence. With recent developments of deep learning techniques, NER has achieved great success based on supervised training on a large amount of labeled data. However, it's expensive and time-consuming to collect high-quality annotations especially for the token-level. To solve the issue of lack of quality labeled data, zero-shot learning (ZSL) (Xian et al., 2017) has drawn a lot of attention recently. The goal of ZSL is to achieve decent performance for new tasks without human annotations by transferring previous knowledge. In this paper, we focus on solving the task of *Zero-shot Named Entity Recognition* that learns a NER system with zero token-level annotations for training. Zero-shot NER is a challenging task that has been rarely studied. Previous works either use POS taggers (Straka and Straková, 2017) or entity dictionaries to provide additional annotations. POS taggger based methods (Fries et al., 2017) require extra human efforts of designing POS tag based regular expressions. Dictionary based methods either ignore the context information (Guerini et al., 2018) or use noisy sequential labeled data to train simple LSTM (Huang et al., 2015) models.

To effectively build a NER system from scratch, we formulate the NER task as a Textual Entailment (TE) problem (Yin et al., 2020) and propose to use Textual Entailment with Dynamic Contrastive Learning (TEDC) to solve this task. TE studies the relation of two assertive sentences, Premise (P) and Hypothesis (H): whether H is true given P. In the meantime, NER aims at identifying whether a word segment is an entity or not given a sentence. To the best of our knowledge, we are the first to formulate the NER task as a TE task by realizing this analogy. This formulation not only utilizes the pre-trained textual entailment model, but also fits for the situation where we don't have full annotations. TE only needs the label for one entity other than the whole sequence to train the model. Furthermore, we combine the textual entailment model with a dynamic contrastive learning framework to contrast the entities and non-entities in the same sentence. The contrastive learning framework helps the model to output the entities with a higher probability to be entailed with the input sentence other than non-entities. And we propose to adjusts the weights of the contrastive loss during the training dynamically.

In summary, the main contributions of our work are as follows: 1) We are the first work that formulates the NER task as a textual entailment prob-

lem. This formulation is more suitable for the situation when we don't have annotations for the whole sequence. 2) We propose to use Textual Entailment with Dynamic Contrastive Learning (TEDC) to solve the zero-shot NER task. 3) Experiments on two real-world datasets show that TEDC achieves state-of-the-art performance for zero-shot NER.

2 Proposed Method

In our method, we obtain initial supervision from the entity dictionaries and use the matched entities to provide entailment pairs as our training data. We introduce how to use textual entailment to solve the NER task in section 2.1 and illustrate the dynamic contrastive learning framework in section 2.2.

2.1 Textual Entailment for NER

Instead of treating NER as a sequence labeling problem, we use a textual entailment model to solve the NER task. It can not only alleviate the noisy labeling problem, but also transfer knowledge from pre-trained textual entailment models.

2.1.1 Entailment Pairs

To transfer the NER task into textual entailment, we need to form textual entailment pairs. Given an input sentence, "John is playing piano", the NER task is to recognize that "John" is a PERSON, which is equivalent to ask if "John is a PERSON" is true. The input sentence acts as a premise, while the assertion "John is a PERSON", acts as a hypothesis. Then the NER task is transferred into a textual entailment problem which is to determine whether the hypothesis is true given the premise.

Formally, given an input sentence A, $x_A = \{w_1, w_2, ..., w_n\}$, which contains n tokens, we need to recognize whether a sub-sequence, $s_{i,j} = \{s_i, s_{i+1}, ..., s_j\}$, where $i >= 1$ and $j <= n$, contained in x_A is an entity or not. Given t entity types in the dataset $T = \{E_1, E_2, ..., E_t\}$, an entailment pair is constructed as (x_A, x_B), where $x_B = \{s_i, s_{i+1}, ..., s_j, \text{is, a}, E_k\}$ and $E_k \in T$. To train the entailment model, we need to construct both positive and negative entailment pairs. For the positive examples, we use the entities in dictionaries as the supervision. If a sub-sequence exactly matches with the surface name of an entity in the dictionaries, we use it to construct a positive entailment pair with its entity type. For the negative examples, we sample from the collection with all the sub-sequences not existing in the entity dictionaries to balance the rate of positive/negative examples.

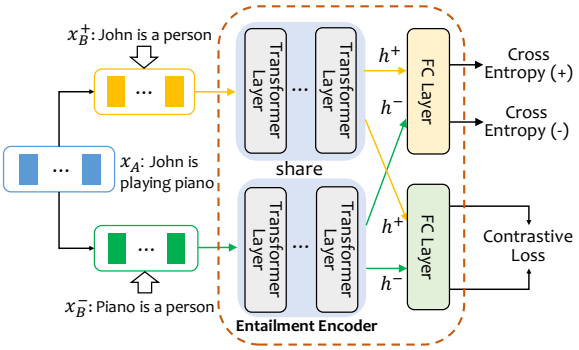

Figure 1: The overall framework of the TEDC model.

Since most sub-sequences are non-entities, we are more likely to obtain true negative examples when we only sample a small portion from the collection. In the experiments, we control the sampling of negative pairs by fixing the rate of negative/positive examples at r.

2.1.2 Entailment Encoder

As shown in Figure 1, we concatenate the sentences (x_A, x_B) in the entailment pair and feed it into the entailment encoder. Here we use Roberta (Liu et al., 2019) to encode the input sequence and a fully connected layer is applied for binary textual entailment classification:

$$h = \text{RoBERTa}(x_A, x_B), \quad (1)$$

$$p = \text{softmax}(Wh + b), \quad (2)$$

where $h \in \mathbb{R}^{d_h}$ is the embedding for the [CLS] token, $W \in \mathbb{R}^{2 \times d_h}$ and $b \in \mathbb{R}^2$ are parameters, p is the output probability for textual entailment. In order to regularize the neural network, we add an additional dropout layer with dropout rate d on the output of the [CLS] embedding h.

For the textual entailment, we use cross-entropy loss to train the model to identify whether the hypothesis about the word segments are true or false:

$$\mathcal{L}_{ce} = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log p_i, \quad (3)$$

where N is the number of training examples.

2.2 Dynamic Contrastive Learning

To improve the model's ability to discriminate entities and non-entities, we combine textual entailment with dynamic contrastive learning for the zero-shot NER task. Additional to the cross-entropy loss, we add a dynamic contrastive learning loss to contrast the entities and non-entities. Given a positive entailment pair, we contrast it with r negative pairs that are constructed with the same sentence, which means we contrast positive and negative examples with the same premise but dif-

ferent hypotheses. As shown in Figure 1, "John is a person" is a positive hypothesis for the sentence "John is playing piano", and "Piano is a person" is a negative hypothesis for this sentence.

The contrastive loss is proposed to push the similarity of the positive entailment pairs higher than the negative entailment pairs. Here we use a fully connected layer on top of the entailment embeddings to simulate the similarity, $s_i = W_s h_i + b_s$, where h_i is the embedding for the i-th entailment pair, $W_s \in \mathbb{R}^{1\times d_h}$ and $b_s \in \mathbb{R}$ are parameters and s_i is the similarity between the premise and the hypothesis of the i-th example. In the implementation, we put the positive entailment pair and its negative pairs in the same batch to calculate the contrastive loss:

$$\mathcal{L}_{cr} = -\log \frac{\exp(\mathrm{s}^+/\tau)}{\sum_{i=1}^{N_{cr}} \exp(s_i/\tau)}, \quad (4)$$

where s^+ is the similarity for the positive entailment pair, s_i is the similarity for all the entailment pairs for the same sentence, $N_{cr} = 1 + r$ and τ is the temperature parameter.

We add the cross entropy loss and the contrastive loss together to train the whole model. To provide a stable training procedure and control the importance of the contrastive loss, we use a cosine function (Loshchilov and Hutter, 2016) to dynamically increase the weight for the contrastive loss:

$$\mathcal{L} = \mathcal{L}_{ce} + d \cdot \mathcal{L}_{cr}, \quad (5)$$

$$d = \max(0, \frac{1}{2} * (1 + \cos((1 - \frac{t}{T})\pi), \quad (6)$$

where d is the dynamic weight, t is the current training step, T is the total training step. During the trainig, d will increase from 0 to 1, which is also a warm-up process.

Training Process. The training process of TEDC has two phases: the pre-training stage and the fine-tuning stage. For the pre-training, we use RoBERTa to initialize the entailment encoder, and pre-train TEDC on a textual entailment dataset, MNLI (Williams et al., 2018), to transfer knowledge. During the fine-tuning, we minimize the loss illustrated in Equation 5 on the entailment pairs constructed from NER datasets.

Inference strategy. After the two-phase training process, we use the model to recognize entities for a test sentence. For each input, we generate entailment pairs by accompanying the sentence with all possible sub-sequences with each entity type. For each pair, we send it into the entailment model to obtain the result whether it is Positive or Negative.

Dataset	BC5CDR	NCBI-Disease
Entity Types	Disease, Chemical	Disease
Dictionary Coverage	51.7%	47.3%
# of Total S(E)	18,256(12,850/15,935/28,785)	8,552(6,892)
# of Training S(E)	5,827(4,182/5,203/9,385)	6,433(5,154)
# of Validation S(E)	5,928(4,244/5,347/9,591)	1,048(787)
# of Test S(E)	6,501(4,424/5,385/9,809)	1,071(960)
# of E (length <= 3)	93.3%	96.6%

Table 1: Statistics of the datasets. For the dictionary coverage, we show the percentage of unique entities contained in the vocabularies. S is short for sentences while E means entities. For the BC5CDR dataset, we show the number of entities as the number of Disease/Chemical/Total Entities.

We recognize all the Positive entailment pairs as entities. To balance the computational cost and the performance, we set a maximum entity length L to limit the number of sub-sequence candidates.

3 Experiments

3.1 Datasets and Dictionaries

Two real-world NER datasets are used in the experiments: BC5CDR (Wei et al.) and NCBI-Disease (Doğan et al., 2014). 1) **BC5CDR** is a task dataset from the BioCreative V Chemical and Disease Mention Recognition challenge[1]. It consists of 1,500 PubMed[2] articles containing 12,852 Disease entities and 15,935 Chemical entities. 2) **NCBI-Disease**[3] is fully annotated at the mention level for Disease Name Recognition. It contains PubMed 793 abstracts with 6,892 Disease entities. We use the data splits provided in the original dataset for training, validation and test.

In the zero-shot NER setting, we use entities from dictionaries as the initial supervision. For these two datasets, we use the CTD Chemical and Disease vocabularies[4] to serve as the entities in the knowledge base. These dictionaries contain 17,097 Chemical entities and 13,061 Disease entities.

3.2 Experiment settings

In order to validate the effectiveness of our method, we compare TEDC with the following four baselines: 1) **Dictionary Match** recognizes the word segments as entity mentions if they match the entities in the dictionaries. 2) **NN$_\mathbf{g}$** (Guerini et al., 2018) is a three-layer bidirectional LSTM that classifies an input sequence of tokens either as entity or non-entity for a certain entity category. 3) **AutoNER** (Shang et al., 2018) builds a binary classi-

[1]https://biocreative.bioinformatics.udel.edu/tasks/biocreative-v/track-3-cdr/

[2]https://pubmed.ncbi.nlm.nih.gov/

[3]https://www.ncbi.nlm.nih.gov/CBBresearch/Dogan/DISEASE/

[4]http://ctdbase.org/downloads/

Datasets	BC5CDR									NCBI-Disease		
	Total			Disease			Chemical			Disease		
Metrics	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
Dictionary Match	63.35	55.95	59.42	65.41	46.64	54.46	61.65	67.70	64.56	57.01	65.82	61.10
NNg	70.08	69.19	69.63	69.01	61.92	65.27	70.95	76.36	73.56	55.71	67.71	61.13
AutoNER	81.64	76.36	78.91	76.85	65.71	70.84	84.96	85.02	84.99	74.09	66.49	69.25
Fuzzy-LSTM-CRF	84.01	69.15	75.86	77.30	64.04	70.05	89.31	73.14	80.42	81.95	69.82	75.40
Roberta	86.77	66.10	75.04	81.11	55.41	65.84	89.89	77.23	83.08	80.17	66.50	72.67
TEDC w/o Contrastive	87.69	83.20	85.39	84.07	78.41	81.14	91.15	87.92	89.51	82.60	73.57	77.83
TEDC w/o Dynamic	88.53	**85.37**	86.92	84.69	**79.04**	81.77	90.10	**92.24**	91.15	83.60	**76.17**	79.71
TEDC	**89.16**	84.96	**87.01**	**86.57**	77.62	**81.85**	**91.30**	91.77	**91.53**	**85.23**	75.17	**79.88**

Table 2: Experiment results for the zero-shot NER task on two datasets: BC5CDR and NCBI-Disease.

Datasets	BC5CDR			NCBI-Disease		
Metrics	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
Dictionary Match	/	/	/	/	/	/
NNg	19.78	21.72	20.71	10.08	26.39	14.59
AutoNER	14.86	2.24	3.90	28.67	17.92	22.05
Fuzzy-LSTM-CRF	61.31	52.83	56.76	**63.93**	46.20	53.64
Roberta	54.58	26.12	35.33	50.58	23.97	32.53
TEDC	**64.70**	**60.47**	**62.51**	63.89	**52.03**	**57.35**

Table 3: Performance of zero-shot entities.

fier to distinguish Break from Tie between adjacent tokens. 4) **Fuzzy-LSTM-CRF** (Shang et al., 2018) customizes the conventional CRF layer in LSTM-CRF into a Fuzzy CRF layer. 5) **Roberta** (Liu et al., 2019) is a pre-trained language model. We use Roberta-base as in our proposed model, TEDC.

Since these baselines can only use full supervisions, we use the dictionaries to obtain pseudo token-level labels to train the models. For all the baselines, we use the recommended parameters provided by the original paper. For TEDC, we use the Adam optimizer with a learning rate of 1e-5. For NN_g and TEDC, we set the rate of negative/positve examples r as 20. We feed one positive entailment pair and 20 negative pairs in one batch and therefore the batch size is 21. For TEDC, the dropout rate d is 0.2 and the maximum input length is 512. We use the Roberta-base model with 12 transformer layers in our experiments. The hidden dimension d_h in transformer layers is 768.

3.3 Results

We report the experiment results for zero-shot NER on two real-world datasets, BC5CDR and NCBI-Disease. We use Precision, recall and F1 for the performance evaluation. For the BC5CDR dataset, we not only show the total performance for two entity types, Disease and Chemical, but also illustrate the detailed information for separated entity types.

Overall Performance. As shown in Table 2, our proposed model outperformances all the other baselines for both datasets on all the three metrics. It can be observed that our TEDC improves the best baseline for BC5CDR (AutoNER) and NCBI-Disease (Fuzzy-LSTM-CRF) by 10.26% and 5.94% in F1 score, respectively. Both the best baselines try to transfer knowledge from dictionaries and might suffer from the noisy labeled data. The proposed TEDC alleviates this issue by formulating the NER task as textual entailment. Additionally, the dynamic contrastive framework improves the model's ability of discriminating the entities and non-entities.

Ablation Study. We also conduct the ablation experiments to explore the impact of the contrastive component and dynamic weight. For the variant model without dynamic weight, we set the weight for the contrastive loss as 0.1. Table 2 shows that the model with contrastive learning framework achieves the better performance, which verifies TEDC's ability for discriminating entities and non-entities. Furthermore, the model without dynamically adjusted contrastive loss performs slightly worse, which indicates the effectiveness of the dynamic weight.

Performance on Zero-Shot Entities. Table 3 reports the performance on the entities that have never been seen in the training data. Compared to the baselines (especially NNg and AutoNER), the superiority of our TEDC becomes more significant on the those unseen entities, which demonstrates the ability of our TEDC in handling the task of zero-shot NER.

4 Conclusion

In this work, we propose a Textual Entailment model with Dynamic Contrastive Learning (TEDC) for zero-shot NER. TEDC is the first to model the NER task as a TE problem to fit the situation when there are not annotations for the whole sequence. Furthermore, the dynamic contrastive learning framework improves the model's ability to discriminate entities and non-entities.

References

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. Ncbi disease corpus: a resource for disease name recognition and concept normalization. *Journal of biomedical informatics*, 47:1–10.

Jason Fries, Sen Wu, Alex Ratner, and Christopher Ré. 2017. Swellshark: A generative model for biomedical named entity recognition without labeled data. *arXiv preprint arXiv:1704.06360*.

Marco Guerini, Simone Magnolini, Vevake Balaraman, and Bernardo Magnini. 2018. Toward zero-shot entity recognition in task-oriented conversational agents. In *SIGdial*, pages 317–326.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv*.

Ilya Loshchilov and Frank Hutter. 2016. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Jingbo Shang, Liyuan Liu, Xiaotao Gu, Xiang Ren, Teng Ren, and Jiawei Han. 2018. Learning named entity tagger using domain-specific dictionary. In *EMNLP*, pages 2054–2064.

Milan Straka and Jana Straková. 2017. Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99.

Chih-Hsuan Wei, Yifan Peng, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Jiao Li, Thomas C Wiegers, and Zhiyong Lu. Overview of the biocreative v chemical disease relation (cdr) task.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL*, pages 1112–1122.

Yongqin Xian, Bernt Schiele, and Zeynep Akata. 2017. Zero-shot learning-the good, the bad and the ugly. In *CVPR*, pages 4582–4591.

Wenpeng Yin, Nazneen Fatema Rajani, Dragomir Radev, Richard Socher, and Caiming Xiong. 2020. Universal natural language processing with limited annotations: Try few-shot textual entailment as a start. In *EMNLP*, pages 8229–8239.