Label Refinement via Contrastive Learning for Distantly-Supervised Named Entity Recognition

Huaiyuan Ying[1]* **Shengxuan Luo**[2,3]* **Tiantian Dang**[4] **Sheng Yu**[2,3]
[1]Department of Mathematical Sciences, Tsinghua University
[2]Center for Statistical Science, Tsinghua University
[3]Department of Industrial Engineering, Tsinghua University
[4]Department of Automation, Tsinghua University
{yinghy,luosx18,dtt19}@mails.tsinghua.edu.cn
syu@tsinghua.edu.cn

Abstract

Distantly-supervised named entity recognition (NER) locates and classifies entities using only knowledge bases and unlabeled corpus to mitigate the reliance on human-annotated labels. The distantly annotated data suffer from the noise in labels, and previous works on DSNER have proved the importance of pre-refining distant labels with hand-crafted rules and extra existing semantic information. In this work, we explore the way to directly learn the distant label refinement knowledge by imitating annotations of different qualities and comparing these annotations in contrastive learning frameworks. the proposed distant label refinement model can give modified suggestions on distant data without additional supervised labels, and thus reduces the requirement on the quality of the knowledge bases. We perform extensive experiments and observe that recent and state-of-the-art DSNER methods gain evident benefits with our method.

1 Introduction

Named entity recognition (NER) refers to the sequence tagging task of detecting the interested entities in unstructured texts and classifying them into predefined categories. NER serves as a foundation part of information extraction in natural language processing (NLP) with applications in many downstream tasks such as question answering (Khalid et al., 2008; Jin et al., 2021), knowledge graph construction (Jia et al., 2018; Zhao et al., 2018), and dialog systems (Bowden et al., 2018). Supervised NER models have been developing rapidly in recent years and have achieved enormous success (Huang et al., 2015; Wang et al., 2020). However, acquiring abundant high-quality human annotations where every word within a sentence should be labeled can be very expensive and limits the application of NER models in many domains.

To alleviate the reliance on human annotations, a practical approach is to introduce distant supervision (Mintz et al., 2009) to automatically generate labeled data by matching entities in easily-obtained knowledge bases. Meanwhile, after years of development, there are many open access knowledge bases or dictionaries such as WikiData[1] and YAGO[2] in the general domain and UMLS (Lindberg et al., 1993) and MeSH[3] in the biomedical domain, which makes it possible to annotate large scale training data for NER models automatically. However, distant annotation suffers from two issues: **incomplete annotation** and **noisy annotation**. The knowledge bases with limited coverage of entities usually label only part of the entities in text, and the remaining entities are incorrectly labeled as background, denoted as incomplete annotations. The noisy annotation occurs when an entity with more than one word and the knowledge bases only contain a sub-sequence of the entity, resulting in partial annotation and sometimes wrong labeling of the entity type.

The neural network-based NER model has a strong ability in fitting the noise in training data, resulting in poor performance with distant labels. Some previous works focusing on distantly-supervised named entity recognition (DSNER) attempt to mitigate the two issues by applying techniques including: (1) entity selection (Yang et al., 2018; Zhang et al., 2021) or noisy entity removal (Onoe and Durrett, 2019); (2) label smoothing (Yang et al., 2018; Shang et al., 2018), (3) iteration and early stopping Liang et al. (2020); (4) PU-learning (Meng et al., 2021).

In addition, many works have found that pre-

* Equal contribution.

[1] https://dumps.wikimedia.org/wikidatawiki/entities/

[2] https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/

[3] https://www.nlm.nih.gov/mesh/download_mesh.html

Findings of the Association for Computational Linguistics: NAACL 2022, pages 2656 - 2666

refining the distant labels by artificial rules (Shang et al., 2018; Zhang et al., 2021; Liang et al., 2020; Meng et al., 2021) and extra semantic information (Zhang et al., 2021; Liang et al., 2020) effectively improves the performance of DSNER models. Inspired by this, we propose a framework to train an automatic distant label refinement model. Specifically, we generate annotations of different qualities by adding noises to weaken the annotations or enhancing the annotations with semantic parsing (Chen and Manning, 2014). Then we apply contrastive learning to guide a sequence scoring model to learn which annotations are better based on the sentence given two different annotations. Finally, the sequence scoring model can make a refinement suggestion on each token in arbitrary sentences and corresponding annotations. Therefore, it is model-agnostic, which can be stably and effectively used in data preprocessing for all DSNER models and consequently improves the performance of the DSNER models.

Due to the complex nomenclature (Névéol et al., 2018) and massive amounts of terminology in the biomedical domain, incomplete and noisy annotations are more evident in the distantly annotated corpus. Therefore, we focus on distant data refinement for DSNER in this work. Experiments show that the framework is good at amending the noisy or incomplete entities in distant data and significantly increases recall and F1 scores for DSNER models. The proposed method is named **CReDEL**, for **C**ontrastive **Re**finement of **D**istant **E**ntity **L**abels. The source code of our model is publicly available at `https://github.com/yinghy18/CReDEL`.

We summarize our contributions as follows:

- We propose an automatic label refinement method to mitigate the issues in distant NER data. The CReDEL is model-agnostic and consequently improves the performance of all DSNER models.
- We introduce a contrastive learning technique combined with a novel contrastive sample generation module. Trained on automatically annotated enormous corpus with knowledge bases, it empowers the proposed scoring model to evaluate annotation qualities.
- We conducted experiments on BC5CDR and NCBI-Disease to verify the effectiveness of CReDEL with classical NER and state-of-the-art DSNER methods. We show that our model brings consistent improvement for these methods.

2 Related Work

Distantly-Supervised Named Entity Recognition

Compared to fully supervised NER, DSNER gets rid of human annotations and uses knowledge bases or dictionaries to annotate the corpus automatically. Some DSNER works adopt entity selection (Yang et al., 2018; Zhang et al., 2021) or noisy entity removal (Onoe and Durrett, 2019) strategies, while some works design new components to handle multiple possible labels (Yang et al., 2018; Shang et al., 2018). Besides, Liang et al. (2020) applies early stopping to prevent fitting the noise and iteratively self-trains the model to recognize more entities. Peng et al. (2019) formulates DSNER as a positive-unlabeled learning problem. Meng et al. (2021) uses a noise-robust loss and a noisy label removal module and uses a self-training method to improve the generalization ability.

The works mentioned above (Shang et al., 2018; Zhang et al., 2021; Liang et al., 2020; Meng et al., 2021) also demonstrate that the pre-refinement of distant labels significantly improves the performance of DSNER methods. Before applying the DSNER models, these works refine the distant labels by tailoring corpus-aware dictionary (Shang et al., 2018), extending entity boundary by a distant phrase mining model (Shang et al., 2018; Zhang et al., 2021), annotating potential entities via POS tagging, and hand-crafted annotation rules (Liang et al., 2020; Meng et al., 2021). In this work, we propose a model to learn the refinement knowledge of distant NER annotations, which consequently has a lower requirement of the dictionary, corrects labels automatically, and facilitates DSNER models.

Contrastive learning

Intuitively, the contrastive model is trained via comparing between input data (Le-Khac et al., 2020), in which way the embeddings learn to put together similar samples and push away different ones. The core of contrastive learning is thus generating positive and negative sample pairs (Kalantidis et al., 2020). Previous works in NLP fields provide various ways of designing positive-negative sample

pairs. Yuan et al. (2020) uses CODER to learn term representations by maximizing similarities between positive term-term pairs, where the word "positive" indicates that the terms are synonyms in the UMLS. Gao et al. (2021) constructs positive sentence embedding pairs using the embeddings of the same sentence passed through different dropouts in BERT in their unsupervised version. Contrastive learning has also been applied to the NER task. Lin et al. (2020) figures out the triggers for entity recognition by generating negative samples after randomly mixing triggers and sentences. Das et al. (2021) also utilizes contrastive learning to optimize distributional divergence and improve few-shot NER performance.

The design of contrastive loss always attempts teaching the model to minimize the distance within clusters despite the difference in forms, such as triplet margin loss (Balntas et al., 2016), contrastive loss (Chopra et al., 2005) and probabilistic NCE-based loss (Ma and Collins, 2018). We also mix hard negative samples in this work and transform the typical margin loss for CReDEL.

3 Methods

In this section, we detailedly describe the whole pipeline of CReDEL (Figure 1). Firstly, a knowledge base is used to obtain the distantly labeled tagging sequence. Then we construct sample pairs on a large corpus and train a scoring model via contrastive learning. We also introduce a module to generate high-quality positive samples by modifying the entity boundary. For the purpose of improving the quality of distant labeled training data, we apply the scoring model to modify datasets used by existing DSNER models, leading to their better performances.

3.1 Distant Labels Generation

For a given knowledge base and corpus, the distantly labeled data always refer to the corpus tagged by matching all entities in the knowledge base following the previous works (Peng et al., 2019; Zhang et al., 2021). The matching algorithm is maximum matching which greedily searches the longest string in the knowledge base. We adopt the "BIO" tagging scheme in this work to represent if a token is at the beginning (B) or inside (I) of a matched entity or does not belong (O) to any entity.

3.2 Contrastive Model Training

Given a sentence consisting of m words $\boldsymbol{X} = (x_1, \cdots, x_m)$ and two "BIO" tag sequences $\boldsymbol{Y_P}, \boldsymbol{Y_N}$ of $\boldsymbol{X}$, the scoring model takes the triple $(\boldsymbol{X}, \boldsymbol{Y_P}, \boldsymbol{Y_N})$ as input and learns to assign a better score to the tag sequence of higher quality, $\boldsymbol{Y_P}$, and a worse score for the other one $\boldsymbol{Y_N}$. This naturally comes down to contrastive learning, and the tag pair $(\boldsymbol{Y_P}, \boldsymbol{Y_N})$ are sampled as follows:

Negative samples generation For one sentence $\boldsymbol{X}$, we imitate the distant annotations to generate two pairs of positive-negative tag sequences: (1) Positive sample $\boldsymbol{Y_{1,P}}$ is the original distant tag sequence described in Section 3.1, while negative sample $\boldsymbol{Y_{1,N}}$ is obtained by randomly subtracting one entity from $\boldsymbol{Y_{1,P}}$. This is the negative tags created by imitating **incomplete** annotations. (2) Positive sample $\boldsymbol{Y_{2,P}}$ is the previous negative one, $\boldsymbol{Y_{1,N}}$. The negative sample $\boldsymbol{Y_{2,N}}$ is produced by changing $\boldsymbol{Y_{2,P}}$ to a different tag sequence within "BIO" tagging scheme with probability p in each position. This pair contains the incomplete annotation and its inferior version by imitating **noisy** annotations. To avoid the model from remembering our dictionary tagging, the pair $(\boldsymbol{Y_{1,P}}, \boldsymbol{Y_{1,N}})$ is only used in development set.

As the scoring model trained through contrastive learning can be insensitive to entity boundary with the aforementioned triples, we also generate another training tag pair using Parser[4]. In this triple, the negative sample $\boldsymbol{Y_{3,N}}$ is the distant tag sequence $\boldsymbol{Y_{1,P}}$ and the positive sample $\boldsymbol{Y_{3,P}}$ is the **parse-enhanced** version of $\boldsymbol{Y_{3,N}}$, which can be obtained from the rules in appendix A.2.

These three cases are hard negative samples and are shown in 1. However, the model requires more easy samples to learn some basic rules of the scoring task, so three other kinds of easy samples are mixed with existing hard samples. Specifically, a certain percentage of existing negative samples will be changed into (1) a random permutation of the distant tag sequence $\boldsymbol{Y_{1,P}}$, (2) a sequence of the same length containing only tag "B", (3) a **shift** of the $\boldsymbol{Y_{1,P}}$ by two tokens, For example, the first tag of $\boldsymbol{Y_{1,P}}$ will be the third in the negative tag sequence, and the second tag will become the fourth in the negative one.

Training Procedure For one triple $(\boldsymbol{X}, \boldsymbol{Y_P}, \boldsymbol{Y_N})$,

[4]In this paper, we use the Stanford Parser(**?**).

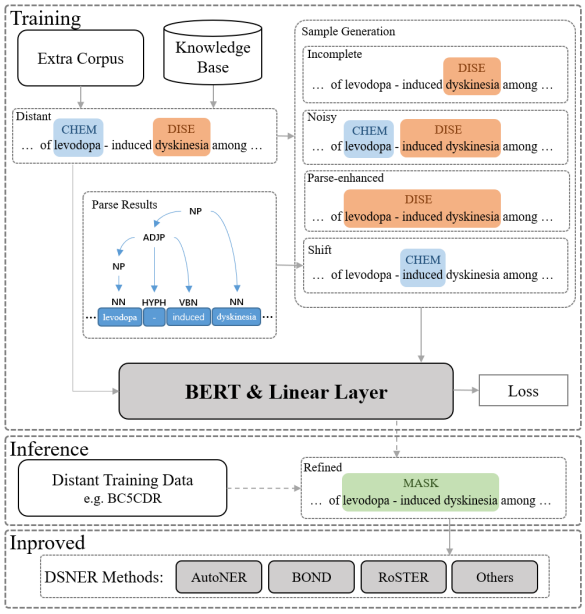

Figure 1: The framework of CReDEL. In training, CReDEL takes a sentence and a generated sample tag pair of distant labels as input, and uses contrastive loss to update model parameters. In inference, CReDEL masks tokens according to the distance between the probability score of the input sentence and distant labels. We use the same sentence as an example in both training and inference for brevity. The refined distant data is then used to train the DSNER models. "CHEM" and "DISE" refer to chemical and disease.

the architecture of the scoring model is as follows:

$$\begin{aligned} \boldsymbol{h} &= BERT(\boldsymbol{X}), \\ \boldsymbol{a} &= [\boldsymbol{s}_O, \boldsymbol{s}_I, \boldsymbol{s}_B] = softmax(\boldsymbol{h}\boldsymbol{W}). \end{aligned} \tag{1}$$

The sentence $\boldsymbol{X}$ is passed through the BERT language model and turned into the hidden representation $\boldsymbol{h}$. Then a linear layer with output dimension three acts on $\boldsymbol{h}$ to get the probability score for "B", "I" and "O" in each position respectively, denoted by $\boldsymbol{a}$, which is a matrix of m rows and 3 columns.

Bounded Cross-Entropy Distance We expect the output score sequence $\boldsymbol{a}$ close to the one-hot encoded positive tag sequence $\boldsymbol{p}$ and away from the encoded negative tag sequence $\boldsymbol{n}$. This is where contrastive learning applies. The "close" and "away" are defined through a distance. We first utilize cross-entropy (CE) loss function value to describe this distance, but the model collapses after enlarging the numerical gap through multiplying this distance by a constant only. To avoid this, a parameter ε is introduced and gives an upper bound to the distance. Finally, our loss function is the margin loss with distance, written as:

$$\mathcal{L} = \max(d(\boldsymbol{a}, \boldsymbol{p}) - d(\boldsymbol{a}, \boldsymbol{n}) + margin, 0), \tag{2}$$

where d is the introduced distance with upper bound:

$$d(x, y) = -\log \frac{\exp(-\mathrm{CE}(x, y)) + \varepsilon}{1 + \varepsilon}. \varepsilon > 0. \tag{3}$$

3.3 Distantly Label Improving

CReDEL will not directly predict a tag sequence, as we find the scoring model cannot effectively discriminate "B" from "I" and a single "I" may appear in the tags without a leading "B". Alternatively, we use the output score defined below to refine distantly labeled data.

During inference, CReDEL takes a sentence and distant tag sequence as input and outputs the token-level distance $d(\boldsymbol{a}, \boldsymbol{Y})$ of classification probability score matrix $\boldsymbol{a}$ and the one-hot encoded distant tag sequence as in 3.2. Then, it evaluates each tag by setting a threshold and masking all positions where the distance is larger than this threshold. Finally the masked sequence $\boldsymbol{Y_{mask}}$ obtained from the original distant tag sequence $\boldsymbol{Y}$ is fed to NER or DSNER models. Here mask means that this tag in this position will not contribute to the calculating of loss in DSNER models.

Since specific NER datasets usually contain several types of entities while the proposed model considers all entities, we add an entity typing module to classify the types of entities extracted from the contrastive scoring model. Then we only mask the tokens predicted to be the desired type in datasets. The entity typing module consists of a classical BERT-based classification model which takes an entity and its surrounding words as input to predict the entity type. The architecture of the entity typing module is described in Appendix A.3.

4 Experiments

4.1 Datasets

Our knowledge base for distant annotation is selected from UMLS by retaining the entities from specific high-quality sources and applying basic cleaning procedures (see Appendix A.1). For simplicity, the knowledge base is still denoted as UMLS in the follow-up sections.

We select 1 million sentences extracted from PubMed abstracts and distantly label them with

UMLS for the training of CReDEL. The size of this distant data is much larger than biomedical NER datasets and provides more semantic knowledge for the contrastive scoring model.

The public datasets we used to test the improvements are listed below:

BC5CDR (Li et al., 2015) is a biomedical Chemical Disease Relation dataset, which is also widely used in biomedical Named Entity Recognition. It consists of 1,500 PubMed articles with 4,409 annotated chemicals, 5,818 diseases entities. The training, development, and test set has 500 articles respectively. Since the types of entities in BC5CDR are "Disease" or "Chemical", We only use the terms of these two types in knowledge bases to label the training set.

NCBI-Disease (Doğan et al., 2014) is a dataset focusing on disease entities. The corpus texts are made of 793 abstracts and 6,881 annotated entities. We use the raw texts of the training set, consisting of 593 abstracts, and test on its test set consisting of 100 human-annotated abstracts. We only regard the "Disease" entity type in this corpus.

For both BC5CDR and NCBI-Disease, we adopt two sources of knowledge bases, the UMLS in Section 3.1 and the core dictionaries containing domain-specific terms from AutoNER (Shang et al., 2018), to annotate the two datasets. The four distantly annotated datasets are denoted as BC5CDR-UMLS, NCBI-UMLS, BC5CDR-AutoNER, and NCBI-AutoNER, respectively.

4.2 Settings

The proposed method pre-refines the training data for DSNER models. Therefore, we select three recent DSNER models to train on the distant data or the refined distant data and compare their performance. Besides, we select classical fully-supervised methods for better illustration and dictionary match as a baseline:

Dictionary Match is the distant annotation method in Section 3.1. For better adaptation to specific datasets, we only keep the entities of corresponding types. **Fully-supervised Methods**. We select two fully-supervised NER models for comparison: (1) **BiLSTM-CRF** (Huang et al., 2015) is the classical RNN-based NER model. It is trained with the distant annotated data or the data refined by CReDEL without language model pre-training. (2) **RoBERTa** (Liu et al., 2019) is a pre-trained language model, and we fine-tune the RoBERTa on the same data as we use in BiLSTM-CRF. **Distantly-supervised Methods**. Three recent or the state-or-art DSNER methods are applied to the distant annotated data or the data refined by CReDEL, including: (1) **AutoNER** (Shang et al., 2018) is a DSNER model with a "Tie or Break" tagging scheme containing entity span detection module and type classification module. (2) **BOND** (Liang et al., 2020) fine-tunes a RoBERTa on distant data with early stopping, then it iteratively expands the dictionary and self-trains the model. (3) **RoSTER** (Meng et al., 2021) uses the noise-tolerant mean absolute error loss with the self-training method and augmented sequences generated by the pre-trained language model without fine-tuning.

4.3 Implementation Details

The contrastive scoring model uses the BERT pre-trained model with dimension 768 followed by a three-class classification layer. In the loss function, The hyperparameter ε is set to 0.2 and the margin equals 0.3. The training adopts a warming up procedure with the learning rate initialized as 8×10^{-6} and reaching its peak 3×10^{-5} at warming step 1000. We use a batch size of 16 sentence with a max length of 256. The training will take about one day on RTX 2080 Ti.

For the compared methods, we preserve their default or recommended parameters unchanged most of the time. The ensemble model number is set to 1 in RoSTER as our computation resources are limited. And the full dictionary of AutoNER containing high-quality phrases without type information is abandoned because our dictionary does not have a counterpart to compare with.

5 Results

5.1 NER Performance Comparison

The datasets improved by the CReDEL with Parser tend to produce more complete term phrases after training, which can be seen in the case study in Section 5.2. However, the case study also shows that the complete term may be away from the test annotations which are part of a long entity. The completeness cannot be captured with traditional exact match F1-score, where predicted entities contribute to the true positive only if the left and right boundary and the entity type all match the test annotations. Motivated by the Boundary IoU (intersection over union) metric (Cheng et al., 2021), we promote another precision, recall, and F1 com-

	BC5CDR-UMLS			BC5CDR-AutoNER			NCBI-UMLS			NCBI-AutoNER		
Methods	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
Matching	0.658	0.613	0.635	0.917	0.578	0.709	0.879	0.334	0.484	0.844	0.544	0.662
Original Distant												
BiLSTM-CRF	0.638	0.480	0.547	0.918	0.506	0.652	0.811	0.401	0.537	0.88	0.276	0.420
AutoNER	0.641	0.516	0.572	0.880	0.560	0.685	0.803	0.485	0.605	0.863	0.269	0.410
RoBERTa	0.657	0.554	0.601	0.888	0.619	0.729	0.819	0.451	0.581	0.882	0.328	0.479
BOND-stage2	0.603	0.658	0.629	0.896	0.612	0.727	0.819	0.437	0.570	0.877	0.352	0.503
RoSTER	0.645	0.701	0.672	0.825	0.713	0.765	0.913	0.545	0.683	0.873	0.397	0.546
Refined w/o Parser												
BiLSTM-CRF	0.672	0.492	0.568	0.846	0.550	0.667	0.847	0.553	**0.669**	0.827	0.439	0.574
AutoNER	0.660	0.557	0.604	0.802	0.639	0.711	0.800	0.588	0.678	0.775	0.376	0.506
RoBERTa	0.657	0.643	0.650	0.807	0.686	0.741	0.834	0.610	**0.704**	0.839	0.476	0.607
BOND-stage2	0.631	0.720	**0.673**	0.819	0.677	0.742	0.888	0.644	**0.746**	0.846	0.514	0.640
RoSTER	0.648	0.787	0.711	0.761	0.780	0.770	0.748	0.734	**0.741**	0.828	0.666	**0.738**
Refined												
BiLSTM-CRF	0.617	0.623	**0.620**	0.844	0.577	**0.686**	0.856	0.516	0.644	0.808	0.445	**0.575**
AutoNER	0.634	0.608	**0.621**	0.793	0.697	**0.742**	0.814	0.584	**0.680**	0.739	0.394	**0.514**
RoBERTa	0.629	0.712	**0.668**	0.789	0.747	**0.768**	0.814	0.602	0.692	0.822	0.485	**0.610**
BOND-stage2	0.570	0.802	0.666	0.689	0.836	**0.756**	0.843	0.637	0.726	0.800	0.557	**0.657**
RoSTER	0.619	0.849	**0.716**	0.717	0.868	**0.786**	0.714	0.709	0.711	0.781	0.624	0.694

Table 1: The Boundary Intersection metric and correspoding precision, recall and F1 score of the distantly supervised methods on datasets BC5CDR and NCBI-Disease annotated by our UMLS dictionary and AutoNER dictionary. Boldface numbers indicate that this experiment with the left-side model achieves the best performance on one specific dataset among the three data conditions.

puting method, called **BI** (Boundary Intersection) score , by considering intersection as follows:

$$\begin{aligned} \text{Prec.} &= \frac{|P_e \cap G_e|}{|P_e|}, \\ \text{Rec.} &= \frac{|P_e \cap G_e|}{|G_e|}, \\ \text{F1} &= \frac{2 \times \text{Prec.} \times \text{Rec.}}{\text{Prec.} + \text{Rec.}}. \end{aligned} \tag{4}$$

Here P_e and G_e are the set of predicted label positions and true label positions within some entity. Intuitively speaking, the new F1 is token-level F1 after excluding the condition that both prediction and ground truth are 'O'. This metric can describe the actual performance of the methods on data improved by CReDEL with parser in the aspect of improving term completeness. Moreover, it is suitable for real-world settings like extracting terms. The new F1-score, precision, and recall on all datasets and methods are demonstrated in Table 1. The "Original Distant", "Refined w/o Parser" and "Refined" refer to the distant data, the data refined by CReDEL trained without parse-enhanced samples, and the data refined by CReDEL, respectively.

We should compare the scores on the same dataset by the same DSNER model trained with original distant data and the refined data. Under the BI score metric, Table 1 tells us that the data modified by the CReDEL with Parser achieves an almost universal improvement in F1-score compared to original data. In BC5CDR-UMLS, BC5CDR-AutoNER, and NCBI-AutoNER it also beats the CReDEL without Parser. The promotion of F1 is mainly obtained through relatively more considerable progress in recall score and a slight change in precision score. This can prove the superiority of our method.

We also produce the exact match F1-scores in Table 2 denoted as the precision*, recall*, and F1-score* on the same datasets. In general, both data modified by CReDEL with and without Parser achieve higher recall scores on all the datasets and all the methods, and the one modified by CReDEL with Parser further increases the recall on much more than half the experiments. Meanwhile, the precision scores are boosted on about half of the experiments, especially on the two datasets tagged by the UMLS subset. Consequently, over $\frac{3}{4}$ of the data reach a higher F1-score among the experiments in Table 2 and not in a trade-off manner.

It is worth noticing that the gain in F1-score differs in range, and we claim that for a dictionary not dataset-specific, such as our dictionary, which can be generally applied, the improvement will be rather noticeable. On the other hand, the construction of AutoNER dictionaries includes more handcrafted cleaning rules which reduce the noise in annotations. They only consist of about 1k terms rather different from general domain dictionaries

	BC5CDR-UMLS			BC5CDR-AutoNER			NCBI-UMLS			NCBI-AutoNER		
Methods	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
Matching	0.593	0.393	0.473	0.859	0.482	0.617	0.636	0.227	0.335	0.606	0.447	0.514
Original Distant												
BiLSTM-CRF	0.568	0.431	0.490	0.836	0.524	0.644	0.621	0.225	0.330	0.613	0.473	0.534
AutoNER	0.558	0.469	0.509	0.798	0.586	**0.675**	0.585	0.230	0.330	0.595	0.510	**0.549**
RoBERTa	0.597	0.545	0.570	0.793	0.665	**0.723**	0.617	0.309	0.412	0.600	0.509	0.551
BOND-stage2	0.565	0.603	0.583	0.788	0.666	**0.721**	0.623	0.293	0.398	0.596	0.505	0.546
RoSTER	0.596	0.628	0.612	0.733	0.726	**0.729**	0.586	0.326	0.419	0.649	0.551	**0.596**
Refined w/o Parser												
BiLSTM-CRF	0.572	0.488	0.527	0.787	0.551	**0.648**	0.604	0.311	0.411	0.614	0.541	**0.575**
AutoNER	0.547	0.537	**0.542**	0.711	0.640	0.673	0.507	0.288	0.367	0.525	0.526	0.525
RoBERTa	0.568	0.662	0.612	0.738	0.696	0.716	0.637	0.502	**0.561**	0.567	0.623	**0.594**
BOND-stage2	0.585	0.644	0.613	0.743	0.692	0.717	0.639	0.470	**0.542**	0.614	0.620	**0.617**
RoSTER	0.553	0.725	0.627	0.674	0.774	0.720	0.605	0.602	**0.603**	0.484	0.673	0.563
Refined												
BiLSTM-CRF	0.537	0.564	**0.550**	0.759	0.541	0.632	0.634	0.333	**0.437**	0.637	0.516	0.570
AutoNER	0.544	0.532	0.538	0.671	0.659	0.665	0.503	0.328	**0.397**	0.553	0.530	0.541
RoBERTa	0.559	0.697	**0.621**	0.695	0.745	0.720	0.581	0.469	0.519	0.548	0.610	0.577
BOND-stage2	0.575	0.693	**0.628**	0.682	0.751	0.715	0.600	0.489	0.538	0.591	0.636	0.613
RoSTER	0.569	0.749	**0.647**	0.652	0.806	0.721	0.582	0.570	0.576	0.455	0.699	0.542

Table 2: The exact match precision*, recall* and F1-score* of the distantly supervised methods on the same datasets and tagging dictionaries. Boldface numbers indicate that this experiment with the left-side model achieves the best performance on one specific dataset among the three data conditions.

	BC5CDR-UMLS	**NCBI-UMLS**
Positive	20,520	18,380
Noisy	11,719	10,199
Mask	28,305	24,652
Masked Noisy	6,290	5,345
FP	4,379	4,285
Masked FP	957	945
FN	6,627	5,264
Masked FN	4,975	4,085
All	148,721	124,250

Table 3: The statistics of tokens in test data by comparing the human annotation, distant annotation, and the refinement results of CReDEL.

and thus have a higher precision score.

5.2 Efficacy of Distant Label Refinement

This section conducts experiments to explain what CReDEL does when noisy annotations exist in distant labels. Specifically, we define the tokens whose labels are different in distant data and golden test data as noisy tokens. The tokens of "O" labels in test data refer to the negative tokens while the remaining tokens refer to the positive tokens. Then we count the positive tokens, noisy tokens, masked, and noise tokens masked by CReDEL. Besides, we also count the false-positive (FP) tokens and false-negative (FN) tokens as well as the masked FP and FN tokens.

For NCBI-Disease, there are 124,250 tokens in all sentences in the test set, while 18,380 tokens are positive, 10,199 are noisy. CReDEL masks 24,652 tokens, and 5,345 of them are noisy. The accuracy of correct masks is much higher than random masking, indicating that CReDEL is conducive to reducing noise. The results in BC5CDR are similar. In addition, CReDEL is better at handling the FN tokens than FP tokens since CReDEL masks 77.6% of FN tokens in NCBI-Disease, and only 22.1% FN tokens are masked. The above observations also explain the increase in recall for the DSNER model trained with the refined data.

For better understanding, the case study is shown in Table 4. In the first example, the phrase "ectopic intracranial retinoblastoma" appears in the golden annotation as a disease entity, while the distant annotation makes an incomplete annotation. CReDEL disagrees with the distant annotation and suggests masking this phrase correctly and masking "child with" redundantly. Correspondingly, the BOND trained with the refined data correctly predicts "ectopic intracranial retinoblastoma" and the BOND trained by original data makes mistakes. In the second example, the golden and distant annotation both label "levodopa" as a chemical entity and "dyskinesia" as a disease entity, while CReDEL masks all tokens in "levodopa-induced dyskinesia". The BOND trained with the refined data predicts the disease entity "levodopa-induced dyskinesia", which is better than the prediction of BOND trained with distant data. This case fits our parse-enhanced entity boundary modifying strategy, and both CRe-

PMID:9400934 from NCBI-Disease	
Golden	The RB1 gene mutation in a child with [ectopic intracranial retinoblastoma]$_{\text{DISEASE}}$.
Distant	The RB1 gene mutation in a child with ectopic intracranial retinoblastoma.
CReDEL	The RB1 gene mutation in a child$_{\text{MASK}}$ with$_{\text{MASK}}$ ectopic$_{\text{MASK}}$ intracranial$_{\text{MASK}}$ retinoblastoma$_{\text{MASK}}$.
BOND (Distant)	The RB1 gene mutation in a child with ectopic intracranial retinoblastoma.
BOND (CReDEL)	The RB1 gene mutation in a child with [ectopic intracranial retinoblastoma]$_{\text{DISEASE}}$.
PMID:23952588 from BC5CDR	
Golden	Risk factors and predictors of [levodopa]$_{\text{CHEMICAL}}$-induced [dyskinesia]$_{\text{DISEASE}}$ among multiethnic Malaysians with [Parkinson's disease]$_{\text{DISEASE}}$.
Distant	Risk factors and predictors of [levodopa]$_{\text{CHEMICAL}}$-induced [dyskinesia]$_{\text{DISEASE}}$ among multiethnic Malaysians with [Parkinson's disease]$_{\text{DISEASE}}$.
CReDEL	Risk factors and predictors of levodopa$_{\text{MASK}}$-$_{\text{MASK}}$induced$_{\text{MASK}}$ dyskinesia$_{\text{MASK}}$ among multiethnic$_{\text{MASK}}$ Malaysians$_{\text{MASK}}$ with [Parkinson's disease]$_{\text{DISEASE}}$.
BOND (Distant)	Risk factors and predictors of levodopa-induced [dyskinesia]$_{\text{DISEASE}}$ among multiethnic Malaysians with [Parkinson's disease]$_{\text{DISEASE}}$.
BOND (CReDEL)	Risk factors and predictors of [levodopa-induced dyskinesia]$_{\text{DISEASE}}$ among multiethnic Malaysians with [Parkinson's disease]$_{\text{DISEASE}}$.

Table 4: Case study in NCBI-Disease and BC5CDR. The dashed line splits the table into data of golden, distant, and refined and prediction of BOND trained with distant data and data refined by CReDEL.

Methods	Prec.	Rec.	F1
BOND (CReDEL)	0.575	0.693	0.628
w/o typing	0.549	0.635	0.589
w. $(\boldsymbol{Y_{1,P}}, \boldsymbol{Y_{1,N}})$	0.560	0.690	0.618
w. easy only	0.561	0.597	0.578
w. hard only	*	*	*
BOND w. parse after	0.550	0.385	0.453
RoSTER (CReDEL)	0.569	0.749	0.647
w/o typing	0.533	0.727	0.615
w. $(\boldsymbol{Y_{1,P}}, \boldsymbol{Y_{1,N}})$	0.570	0.722	0.637
w. easy only	*	*	*
w. hard only	*	*	*
RoSTER w. parse after	0.403	0.379	0.391

Table 5: Here * represents model collapsing with strange predictions or not converging after ablation. The exact match F1-score of the CReDEL, the CReDEL w/o the entity typing module ,the CReDEL trained using negative sample case $(\boldsymbol{Y_{1,P}}, \boldsymbol{Y_{1,N}})$ and easy samples, the CReDEL trained using easy samples/hard samples only, and the DSNER method with parse improvement after it. The results are computed on the dataset BC5-UMLS by methods BOND and RoSTER.

DEL and the DSNER model trained with refined data prefer to predict complete entities.

6 Ablation Study

To verify the effectiveness of each design of CReDEL, we conduct the ablation studies. We mainly discuss three parts, the first of which is the effectiveness of parse and it has been fully exploited in the results part. The second ablation is CReDEL **w/o typing**, that is, without the final entity typing module. The other part of ablation considers the selection of training sample cases. We train CReDEL (1) **w.** $(\boldsymbol{Y_{1,P}}, \boldsymbol{Y_{1,N}})$, using only sample pair $(\boldsymbol{Y_{1,P}}, \boldsymbol{Y_{1,N}})$ in Section 3.2 and easy samples; (2) **w. easy only**, as no hard samples are included; and (3) **w. hard only**, where only easy samples are used. The results are shown in Table 5. The model trained without easy samples simply cannot converge. Its predicted phrases are meaningless and in chaos. The data modified by CReDEL with only easy samples lead the RoSTER method to collapse with zeros or NaN in output probabilities and achieve low scores with BOND method. Other ablations also result in a drop in the F1 score, revealing the contributions of their corresponding part in the model design.

Finally we discuss the condition where the parser does not act on the training set of CReDEL but on the output of DSNER models without CReDEL, denoted as Method **w. parse after**. As an automatic refinement model, we should outperform the baseline refinement with Parser. We apply the same parse-enhancing rule and find the F1-score is far behind CReDEL as demonstrated in Table 5. Even adopting the new BI metric, this model also performs badly with 0.538 and 0.489 BI F1-score on method BOND and RoSTER respectively.

These drops can be explained. Without entity type classification, the model will mask much more phrases that are valid but beyond the dataset annotation types. The BOND or RoSTER models potentially learn to pay attention to these phrases, resulting in a decline in precision. The model trained with $(\boldsymbol{Y_{1,P}}, \boldsymbol{Y_{1,N}})$, on the other hand, tends to fit the positive samples $\boldsymbol{Y_{1,P}}$ which are fixed by dictionary annotations. The BERT model may recognize these words in some batches, and the trained CReDEL can lose generalization abilities. If using

Parser after DSNER methods, it is only a rigid tool rather than a model exploiting semantic information. The CReDEL with Parser takes effect because it is combined with our other designs.

7 Conclusion

This paper proposes a novel approach to automatically learn the refinement knowledge of distantly annotated NER labels and modify the distant labels to enhance DSNER models. The proposed method consists of a contrastive samples generation module, a contrastive training procedure, and a distantly label improving strategy. Experiments demonstrate that our method consistently and significantly improves DSNER and NER models on distantly annotated NER data, and it can be stably applied to all the datasets and methods.

Acknowledgements

Upon the completion of this paper, we would like to thank Zheng Yuan and Hongyi Yuan for their helpful comments and suggestions. We also gratefully acknowledge the support of Jun Xia and his providing of raw data.

References

Vassileios Balntas, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. 2016. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *Bmvc*, volume 1, page 3.

Kevin Bowden, Jiaqi Wu, Shereen Oraby, Amita Misra, and Marilyn Walker. 2018. Slugnerds: A named entity recognition tool for open domain dialogue systems. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Danqi Chen and Christopher D Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 740–750.

Bowen Cheng, Ross Girshick, Piotr Dollár, Alexander C Berg, and Alexander Kirillov. 2021. Boundary iou: Improving object-centric image segmentation evaluation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15334–15342.

Sumit Chopra, Raia Hadsell, and Yann LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE.

Sarkar Snigdha Sarathi Das, Arzoo Katiyar, Rebecca J Passonneau, and Rui Zhang. 2021. Container: Few-shot named entity recognition via contrastive learning. *arXiv preprint arXiv:2109.07589*.

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.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

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

Yan Jia, Yulu Qi, Huaijun Shang, Rong Jiang, and Aiping Li. 2018. A practical approach to constructing a knowledge graph for cybersecurity. *Engineering*, 4(1):53–60.

Qiao Jin, Zheng Yuan, Guangzhi Xiong, Qianlan Yu, Chuanqi Tan, Mosha Chen, Songfang Huang, Xiaozhong Liu, and Sheng Yu. 2021. Biomedical question answering: A comprehensive review. *arXiv preprint arXiv:2102.05281*.

Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. 2020. Hard negative mixing for contrastive learning. *arXiv preprint arXiv:2010.01028*.

Mahboob Alam Khalid, Valentin Jijkoun, and Maarten De Rijke. 2008. The impact of named entity normalization on information retrieval for question answering. In *European Conference on Information Retrieval*, pages 705–710. Springer.

Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. 2020. Contrastive representation learning: A framework and review. *IEEE Access*.

Jiao Li, Yueping Sun, R Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. 2015. Annotating chemicals, diseases, and their interactions in biomedical literature. In *Proceedings of the fifth BioCreative challenge evaluation workshop*, pages 173–182. The Fifth BioCreative Organizing Committee.

Chen Liang, Yue Yu, Haoming Jiang, Siawpeng Er, Ruijia Wang, Tuo Zhao, and Chao Zhang. 2020. Bond: Bert-assisted open-domain named entity recognition with distant supervision. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1054–1064.

Bill Yuchen Lin, Dong-Ho Lee, Ming Shen, Ryan Moreno, Xiao Huang, Prashant Shiralkar, and Xiang Ren. 2020. Triggerner: Learning with entity triggers as explanations for named entity recognition. *arXiv preprint arXiv:2004.07493*.

Donald AB Lindberg, Betsy L Humphreys, and Alexa T McCray. 1993. The unified medical language system. *Yearbook of Medical Informatics*, 2(01):41–51.

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 preprint arXiv:1907.11692*.

Zhuang Ma and Michael Collins. 2018. Noise contrastive estimation and negative sampling for conditional models: Consistency and statistical efficiency. *arXiv preprint arXiv:1809.01812*.

Yu Meng, Yunyi Zhang, Jiaxin Huang, Xuan Wang, Yu Zhang, Heng Ji, and Jiawei Han. 2021. Distantly-supervised named entity recognition with noise-robust learning and language model augmented self-training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10367–10378.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, page 1003–1011, USA. Association for Computational Linguistics.

Aurélie Névéol, Antonio Jimeno Yepes, L Neves, and Karin Verspoor. 2018. Parallel corpora for the biomedical domain. In *International Conference on Language Resources and Evaluation*.

Yasumasa Onoe and Greg Durrett. 2019. Learning to denoise distantly-labeled data for entity typing. 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)*.

Minlong Peng, Xiaoyu Xing, Qi Zhang, Jinlan Fu, and Xuan-Jing Huang. 2019. Distantly supervised named entity recognition using positive-unlabeled learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2409–2419.

Jingbo Shang, Liyuan Liu, Xiaotao Gu, Xiang Ren, Teng Ren, and Jiawei Han. 2018. Learning named entity tagger using domain-specific dictionary. In *2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, pages 2054–2064. Association for Computational Linguistics.

Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2020. Automated concatenation of embeddings for structured prediction. *arXiv preprint arXiv:2010.05006*.

Yaosheng Yang, Wenliang Chen, Zhenghua Li, Zhengqiu He, and Min Zhang. 2018. Distantly supervised ner with partial annotation learning and reinforcement learning. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2159–2169.

Zheng Yuan, Zhengyun Zhao, Haixia Sun, Jiao Li, Fei Wang, and Sheng Yu. 2020. Coder: Knowledge infused cross-lingual medical term embedding for term normalization. *arXiv preprint arXiv:2011.02947*.

Wenkai Zhang, Hongyu Lin, Xianpei Han, Le Sun, Huidan Liu, Zhicheng Wei, and Nicholas Yuan. 2021. Denoising distantly supervised named entity recognition via a hypergeometric probabilistic model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14481–14488.

Zhanfang Zhao, Sung-Kook Han, and In-Mi So. 2018. Architecture of knowledge graph construction techniques. *International Journal of Pure and Applied Mathematics*, 118(19):1869–1883.

A Appendix

A.1 Processing of UMLS

The Unified Medical Language System (UMLS) (Lindberg et al., 1993) is a large-scale resource containing over 4 million unique medical concepts. The restricted set of source ontologies in UMLS include "CPT", "MEDLINEPLUS", "RXNORM", "SNOMED-CT" and so on (Table 6). After that, we apply necessary cleanings such as removing ambiguous or illegal words and abbreviations that are prone to mismatch.

Source	Name or Explanation
CPT	Current Procedural Terminology
HPO	Human Phenotype Ontology
MEDLINEPLUS	Certified patient-oriented web-content
MSH	Medical Subject Headings
MTH	UMLS Metathesaurus Names
NCI	National Cancer Institute Thesaurus
RXNORM	NLM's Nomenclature for Clinical Drugs for Humans
SNOMEDCT	US edn. of the Systematized Nomenclature of Medicine-Clinical Terms

Table 6: The restricted set of source ontologies.

A.2 Parse-based Entity Boundary Modifying

The parser will give a syntax tree of a sentence, with the tokens in the sentence as leaves and syntax labels as other nodes. We set four rules for the parse modifying: (1) If one token is within an entity, we find its parent in the syntax tree generated by the Stanford parser and tag the whole new noun phrase as an entity in the "BIO" tagging scheme if the parent label is "Noun Phrase". (2) If the new entity in (1) starts with a comparative form

of an adjective, cardinal number, conjunction, or pronoun, then we repeatedly delete the first token until it does not belong to one of those labels. (3) If the extension of the original entity encounters conjunction or some specific punctuation, then we reserve the original entity. (4) If none of the tokens in an original entity is a noun word and none of the parent labels is "Noun Phrase", then we delete the entity from the original tag sequence.

A.3 Entity Typing Module

In the entity typing module, we train a BERT-base classifier with the distant data labeled by the same knowledge base. The classifier takes an entity and its surrounding words as input. Then it predicts the entity type of the input entity. In training, the entity type is from the knowledge base, and the model updates with cross-entropy loss.