applied sciences

MDPI

*Article*

# Distantly Supervised Named Entity Recognition with Self-Adaptive Label Correction

**Binling Nie [1,*] and Chenyang Li [2]**

[1] School of Digital Media, Hangzhou Dianzi University, Hangzhou 310005, China
[2] College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China; licy_cs@zju.edu.cn
* Correspondence: binlingnie@hdu.edu.cn

**Abstract:** Named entity recognition has achieved remarkable success on benchmarks with high-quality manual annotations. Such annotations are labor-intensive and time-consuming, thus unavailable in real-world scenarios. An emerging interest is to generate low-cost but noisy labels via distant supervision, hence noisy label learning algorithms are in demand. In this paper, a unified self-adaptive learning framework termed Self-Adaptive Label cOrrection (SALO) is proposed. SALO adaptively performs a label correction process, both in an implicit and an explicit manners, turning noisy labels into correct ones, thus benefiting model training. The experimental results on four benchmark datasets demonstrated the superiority of SALO over the state-of-the-art distantly supervised methods. Moreover, a better version of noisy labels by ensembling several semantic matching methods was built. Experiments were carried out and consistent improvements were observed, validating the generalization of the proposed SALO.

**Keywords:** named entity recognition; noisy label learning; label correction; distant supervision



**Citation:** Nie, B.; Li, C. Distantly Supervised Named Entity Recognition with Self-Adaptive Label Correction. *Appl. Sci.* **2022**, *12*, 7659. https://doi.org/10.3390/app12157659

Academic Editor: Kuei-Hu Chang

Received: 1 July 2022
Accepted: 27 July 2022
Published: 29 July 2022



## 1. Introduction

Named Entity Recognition (NER) aims to detect and classify Named Entities (NEs) mentioned in unstructured text into predefined categories, such as location, person, organization, etc. It is a foundational task in Natural Language Processing (NLP) and can benefit many downstream tasks, e.g., relation extraction, event extraction and question answering systems. The prevalence of deep neural networks has greatly advanced the research in NER [1–4]. Such success can be largely attributed to the large amounts of data with high-quality manual annotations. In practice, such annotations are expensive or even infeasible. On the contrary, with the help of a knowledge base (e.g., Wikidata [5] and Yago [6]), training labels can be automatically generated [7]. Nevertheless, they inevitably involve noisy labels, as illustrated in Figure 1. Unfortunately, the strong memorization power of deep neural networks makes them susceptible to the presence of noisy labels and overfitted to corrupted labels [8], leading to poor generalization. Even popular regularization techniques, such as data augmentation [9], weight decay [10], dropout [11] and batch normalization [12], fail to narrow the gap in performance between fully supervised and distantly supervised NER models.

Attempts have been made in designing noisy label learning algorithms for distantly supervised NER [13–15]. These methods alleviated the negative effect of noisy labels by discarding or reweighting mislabeled training data, but failed to fully exploit useful information from the mislabeled data. As the noisy label learning study [16] proved, a classifier trained on noisy labels has the ability to identify whether a label has been corrupted. As another proof of this study, they verify that leveraging a simple label-correction algorithm has a guaranteed success rate and recovers the correct labels of mislabeled data with high probability. Hence, this work explores how to better utilize mislabeled data by introducing label correction for distantly supervised NER.

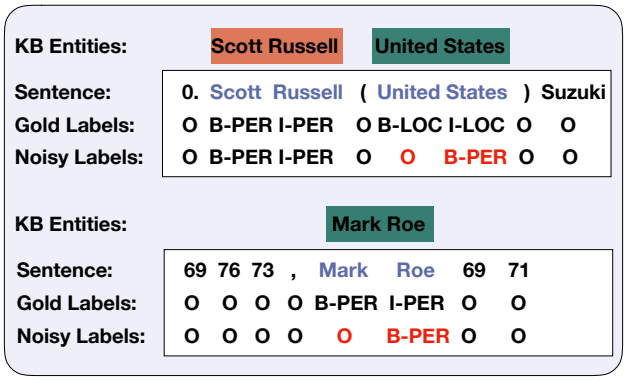



**Figure 1.** Two types of noisy labels exist in the current distant NER datasets (CoNLL03). (1) Category error: the exact label (a none type entity and a PERSON type entity) for the mention United States is wrong. (2) Span error: the mention Mark Roe is wrongly linked to Roe instead of Mark_Roe.

In this work, a unified self-adaptive learning framework termed SALO (Self-Adaptive Label cOrrection) is proposed that can dynamically detect corrupted labels and progressively correct them. SALO consists of three key components: implicit noise denoising, a noisy label detector and a pseudolabel estimator. Implicit noise denoising is devised to produce the supervision signals in each training step. A labeled token selection strategy is introduced with denoising classifiers to select consistent and high-confidence labels for hard supervision signals. A noisy label detector dynamically estimates a soft target distribution at each time step via a pretrained NER classifier, which intrinsically splits hard supervision signals into two classes (i.e., high-quality instances and problematic instances). Depending on the maximum probability of the distribution, problematic instances are under corrected label supervision or left intact. By doing so, the built model is prevented from being overconfident and fully mines valuable information from mislabeled data. A pseudolabel estimator is devised to regenerate the type and position of labels, respectively, based on a noisy example detector. Specifically, a label correction mechanism is introduced to smooth out problematic instances and even completely change the training labels if necessary. An iterative training algorithm is further designed to take full advantage of these data-correction processes, which significantly boost performance.

The main contributions are as follows:

- A self-adaptive learning framework termed SALO is proposed to improve the learning of NER models by dynamically incorporating adaptive label correction into training. An iterative training algorithm continuously optimizes the model while correcting noisy labels.
- A denoising classifier and a noisy label detector are introduced to identify noisy labels. The denoising classifier filters unreliable labeled tokens while the noisy label detector detects the wrongly hard supervision signals to be corrected.
- The built model is evaluated on four benchmark datasets, and the results demonstrate that SALO performs better than other baselines in all noisy datasets.

## 2. Related Works

### *2.1. Fully Supervised Named Entity Recognition*

NER is the foundation of information extraction, which automatically recognizes named entities from natural language texts. Early methods on NER were mainly based on rules [17], which had disadvantages such as poor scalability and high labor costs. Later, traditional statistical machine learning methods were widely used to solve NER problems, such as the hidden Markov model [18], maximum entropy [19], support vector machine [20] and linear chain conditional random field model [21]. However, these models heavily relied on artificially designed features or completely ignored the inherent semantic dependence of the context in text. Recently, deep neural network models take the advantages of representation learning to avoid relying on the hard-coded features, which

allows researchers to focus on designing different character, word, or sentence encoders to improve the representation ability of the NER model and optimize its performance [1–4].

### 2.2. Distantly Supervised Named Entity Recognition

Although deep neural network models have stronger capabilities of representation learning, they rely on large-scale labeled data for training. Nevertheless, the size of labeled data in many vertical fields is limited, which makes the effectiveness of deep neural network models unable to be guaranteed. Bellare and McCallum [13] proposed a missing label Conditional Random Fields (CRF) that only required some of the tokens in text to be labeled with high precision. Jie et al. [22] introduced a self-training approach for recognizing named entities with incomplete data annotations. Mayhew et al. [14] designed a constraint-driven iterative algorithm that learned to detect false negatives in the noisy set and down-weighted them for weighted NER. Shang et al. [15] proposed AutoNER with a new Tie or Break scheme to suit distant supervision from the dictionary. Cao et al. [23] utilized a lightweight scoring strategy to differentiate noisy data from high-quality weakly labeled sentences and proposed a unified neural framework from sequence labeling and classification perspectives. Yang et al. [24] applied partial annotation learning and an instance selector based on reinforcement learning for incomplete and noisy annotations, respectively. Peng et al. [25] took the task as a Positive-Unlabeled (PU) learning and proposed PU learning to unbiasedly and consistently estimate the task loss as if there is fully labeled data. Zhang et al. [26] proposed a probabilistic automatic relabeling method to estimate the pseudotruth label distribution during the training process. Liang et al. [7] leveraged the power of pretrained language models to improve the prediction performance of NER models with self-training. Liu et al. [27] explicitly estimated the confidence score of one label being corrupted into another based on local and global independence assumptions and designed a calibration method to determine the portion of trusted labels and model noise ratio of training data for noisy NER. Zhang et al. [28] cotrained two teacher–student networks to form inner and outer loops for coping with label noise to make a full exploration of mislabeled data. These methods focus on denoising to alleviate the negative effect of noisy labels. Inspired by the noisy label learning theorem [16], the built model SALO focuses on detecting and relabeling the noisy label.

## 3. Methodology

### 3.1. Method Overview

A novel **S**elf-**A**daptive **L**abel c**O**rrection framework (SALO) is proposed, and the overview of the proposed method is illustrated in Figure 2. SALO is composed of three key components: an implicit noise denoising module, a noisy example detector and a pesudolabel estimator. The devised clean token selection strategy in denoising learning can produce hard supervision signals in an ideal situation to alleviate the negative effect of label noise. A noisy example detector is designed to select reliable supervision signals, and pseudolabel correction progressively corrects problematic instances. Intuitively, if an incorrect supervision signal is corrected, it is transformed into a useful training instance to benefit model training. An iterative training scheme is adopted to steadily sanitize training data while refine the model.

It is worth noting that the framework can be built upon various NER architectures. The widely used RoBERTa and BiLSTM-CRF architectures are adopted as base models to demonstrate the generalization of the proposed framework.

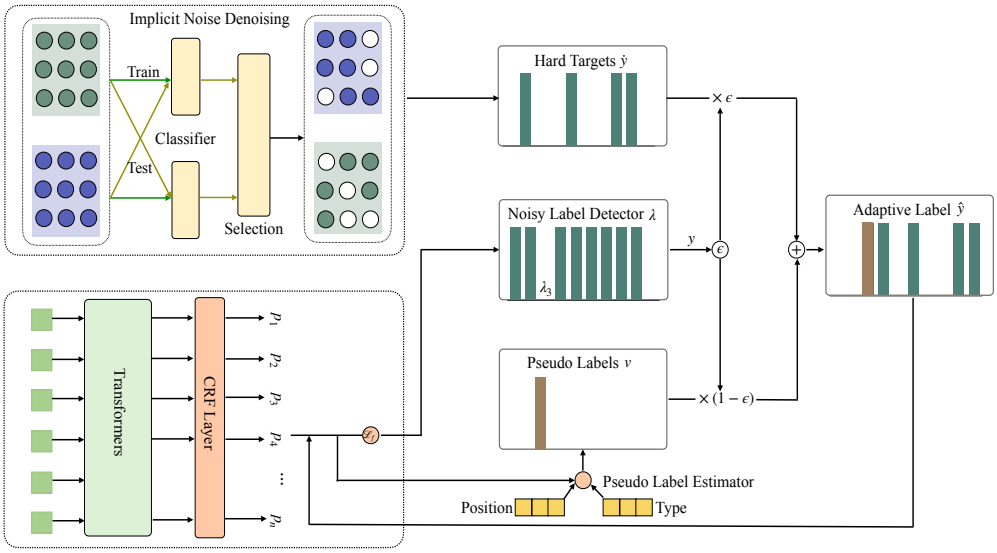



**Figure 2.** An overview of the proposed SALO. The model is iteratively fine-tuned for several steps. In each step, an Implicit Doise Denoising module is trained on the noisy data in a K-fold (k = 2 in the figure) cross-valid fashion to alleviate over-fitting to noisy labels; afterwards, a noisy label detector and a pseudolabel estimator are introduced for noise detecting and explicit noise correction, respectively. While the iteration continues, the noisy labels are steadily identified and corrected, thus benefiting the optimization of the NER model.

### *3.2. Adaptive Label Correction*

In this section, the proposed adaptive label correction framework for refining distantly supervised noisy data is introduced. It is worth noticing that there exists a significant difference between the built model and typical NER. In a common NER training process, one-hot distributions of the labels are fixed during training, while in the built framework, training targets $\dot{y}$ are dynamically predicted by the denoising models the during training process.

$$\dot{y} = argmax(f(x;\theta)) \tag{1}$$

where $f$ represents the corresponding denoising models and $\theta$ is the learned parameters.

An adaptive target distribution $\hat{y}$ is then constructed to replace $\dot{y}$, which aims to progressively improve the model confidence of adaptive label. Specifically, the training target is updated by the Exponential-Moving-Average (EMA) scheme as bulleted lists look like this:

$$\hat{y} = \epsilon \times \dot{y} + (1 - \epsilon) \times v \tag{2}$$

where $\epsilon \in [0, 1]$ controls the weight on the reannotated labels, and $v$ is the pseudolabel vector that is regenerated by self-adaptive label correction and depends on the current time step. The EMA scheme in Equation (2) enables the built model to completely change the training labels if necessary, which alleviates the instability issue of model predictions.

In this study, by assigning zero probability for the target label $q$ and nonzero probabilities for pseudo labels, the networks are able to explicitly control the supervisions assigned to the final NER model. Specifically, the first term $\epsilon \times \dot{y}$ and the second term $(1 - \epsilon) \times v$ in Equation (2), respectively, detect noisy examples and determine how to regenerate the supervision signal for noisy examples. The following subsections detail how to compute $\epsilon$ and $v$.

#### 3.2.1. Implicit Noise Denoising

It has been observed that there exists fluctuations in the predictions of a model with mislabeled instances [29]. Noisy labeled instances are supposed to be supervised by both

their labels and similar instances. For example, in Figure 1, the mention United States is mislabeled as the incomplete label PER. At the same time, the mention United States with similar context is labeled as its correct label LOC. Both of them encourage the model to fit its own supervision signals.

To quantify the fluctuation, the built denoising model derives $C$, a set of all possible correct label sequences that are compatible with the noisy labeled sequence. The built model regards the missing labels as latent variables and learns a latent variable CRF using the following loss:

$$L = -log \sum_{y \in C} (p(y|x)) \tag{3}$$

In this way, the prediction of the denoising model is the ensemble of predictions from all possible correct label sequences, which can quantify the fluctuations naturally. Additionally, the EMA schema allows the denoising model to converge to improved parameters.

The proposed second strategy it to take hard examples into consideration, which are erroneously assumed to be reliable and may not fluctuate. To alleviate the issue, all tokens are ranked according to prediction probabilities and retain high-confidence predictions as clean for which maximum likelihood is computed. Based on the devised clean token selection strategy in denoising learning, SALO can utilize the hard supervision signals in an ideal situation to alleviate the negative effect of label noise.

#### 3.2.2. Noisy Example Detector

In noisy label problems, examples have smaller training losses as they are more likely to be clean labels [16]. Inspired by that, we manipulate the training loss-based adaption factor $\epsilon \in [0, 1]$ in Equation (2) to detect whether hard supervision signals (the label predicted by its denoising model) are noisy.

For a training sentence pair $(X, \dot{Y})$ predicted by the denoising model and each target label $\dot{y} \in \dot{Y}$, we further separate the noisy detection for positive examples (entities) and negative examples (i.e., the O label) because we empirically observe that their forward loss is consistently different. To this end, $\epsilon$ is then obtained:

$$\begin{aligned} \epsilon &= \epsilon_i^P + \epsilon^N \\ \epsilon^P &= max(\dot{y}, \lambda^P),\ \dot{y} \in Y^P \\ \epsilon^N &= max(\dot{y}, \lambda^N),\ \dot{y} \in Y^N \end{aligned} \tag{4}$$

where $\lambda^P$ and $\lambda^N$ denote the lower bound of true positives and true negatives responding to $\epsilon^P$ and $\epsilon^N$, respectively.

The basic intuition behind Equation (4) is to assume all hard targets predicted by their denoising models are more likely to be high-quality instances when $\lambda^P$ and $\lambda^N$ select reasonable enough problematic instances. This design prevents the built model from mislabeling the correct examples with high confidence.

Furthermore, to ensure that the target label $\dot{y}$ always has the largest probability to be clean, it is needed to pinpoint noisy examples and still keep clean examples. Thus, a forward loss $L_f^P$ is defined to detect the lower bound of true positive $\lambda^P$ of $\epsilon^P$ for positive examples.

$$L_f^P = - \sum_{y \in Y^P} \dot{y} \times p \tag{5}$$

where $\dot{y}$ represents a target label in positive examples $Y_p$, and the larger value of $L_f^P$ has higher probability to be clean. If $L_f^P$ ranks at top $k$ in positive labels of the example, we set the $\lambda^P = 1$. The forward loss is detached from backpropagation.

Note that $\epsilon$ and $\lambda$ are all time-step specific variables, which allows the values to adapt to dynamic contexts.

3.2.3. Pseudolabel Estimator

With the proposed implicit noise denoising module and noisy example detector, training data with noisy labels can be identified. Intuitively, simply discarding the noisy subset of training data and retaining the clean subset helps alleviate over-fitting to noisy labels. However, even the data with problematic labels may contain useful information that can further boost model performance if properly corrected. As the noisy theorem states [16], if we obtain a reasonable approximation of ground-truth labels, the label correction algorithm is able to flip corrupted labels to clean labels. In this study, a pseudolabel estimator is introduced to acquire a reasonable approximation of ground-truth labels by exploiting the correlation and context awareness among distant labels.

To reannotate more robust label guessing for noisy examples, a simple yet effective technique is designed to regenerate the position and the type of labels individually. Pseudolabels are usually inferred by the model prediction. Noting that a $y$ is annotated as position-type (B-Person) at predicted probability $p$. The position embeddings and type embeddings can be utilized when estimating the pseudolabel $v$; specifically, whether the probabilities of the position or the type of labels should remain the same are calculated.

$$\begin{aligned} p^t &= \overline{p * e^t} \\ p^p &= \overline{p * e^p} \end{aligned} \tag{6}$$

where $p^t$ and $p^p$ denote the probability that the position/type is that of truth label. $e^t$ and $e^p$ represent one-hot position embeddings and type embeddings.

$$v = \begin{cases} (p^t \geq mean(p)) * e_t + (p^p \leq mean(p)) * e_p & \\ +(1 - (p^t \geq mean(p))) * (1 - (p^p \geq mean(p)) * a & \text{if } \epsilon = 0 \\ 0, & \text{if } \epsilon = 1 \end{cases} \tag{7}$$

where $a$ represents all combinations of position and type in the current dataset. If $p^t \geq mean(p)$, the type of the noisy example is more trustworthy than the position. We marginal all labels out except for the labels that have the same type as the hard example but have different positions. Similarly, we select all the labels with the same position as the noisy example and $O$ label and sum over the tag sequences over these labels. In partial marginalization of the CRF model, we can predict pseudolabels $v$ for noisy examples.

*3.3. Iterative Training*

Although effective, simply performing implicit noise denoising and explicit noise detection fails to take full advantage of each part. Instead, an iterative training scheme is designed to further boost NER performance. As shown in Algorithm 1, the noisy example detector model $f(^s_\theta)$ is firstly warmed up on the noisy labels to better identify whether a label has been corrupted. For each step, denoising classifiers are trained on the noisy data in $K$-fold cross-validation fashion. After pretraining implicit noise denoising, the label correction process is carried out for modifying training data labels. Then, a new step of training is performed. Here, we reinitialize the classifier in every step as it introduces randomness to avoid over-fitting, thus contributing to more robust data filtering. The iteration is terminated after repeating the whole process for $T$ steps, and the model with the best result on the validation set is selected. Finally, the test performance is reported using the selected model on the test set.

**Algorithm 1** Self-Adaptive Label cOrrection algorithm

**Input**: Noisy labeled sentences $\mathbb{D} = \{(x^1, y^1), (x^2, y^2), ..., (x^m, y^m)\}$
**Parameter**: Three network parameters $f(\theta_0^t)$, $f(\theta_1^t)$ $f(\theta^s)$
**Output**: the best model
Pre-train $f(\theta^s)$ with $\mathbb{D}$.
**for** t = 0, 1, ... T−1 **do**
  Randomly initialize $K$ models $f(\theta_0^t)$, $f(\theta_1^t)$,
  Randomly shuffle the noisy dataset $\mathbb{D}$ and divide it into $K$ fold $\mathbb{D}_1$, $\mathbb{D}_2$.
  Train $K$ models $f(\theta_i^t)$ with $i$th-fold dataset $\mathbb{D}_i$, where $i = 0, 1$ via Equation (3).
  Use the $(1-i)$th model $f(\theta_{1-i}^t)$ to annotate new labels $\dot{y}$ of $i$th-fold dataset $\mathbb{D}_i$ by Equation (2).
  Perform the detector model $f(\theta^s)$ to detect whether new labels predicted by $K$-trained denoising models are erroneous. If they are, generate pseudolabels $v$.
  Construct final adaptive labels $\hat{y}$.
  $\hat{y} = \epsilon \times \dot{y} + (1-\epsilon) \times v$
  Update the model $f(\theta_t^s)$ which minimize the loss on adaptive labels.
**end for**

## 4. Experiment

In this section, the performance of SALO was evaluated, comparing it with the state-of-the-art approaches. Additionally, extensive auxiliary experiments were carried out and provide comprehensive analyses to demonstrate the effectiveness of the proposed SALO.

### *4.1. Datasets*

Experiments are conducted on four challenging datasets with distantly supervised noisy labels, including CoNLL03 [30], Twitter [31], Webpage [21] and Wikigold [32]. Unless otherwise specified, the noisy labels provided in [7] were adopted for comparison, which are generated using POS tagger to detect candidate entities, filtering the ambiguous entities by Wikidata query service and building multisource gazetteers to match an entity with a type. Finally, a set of hand-crafted rules are used to match unmatched tokens [7]. To investigate the robustness of the proposed methods on different noisy data, a new version of distantly supervised labels was constructed by combining [7] and semantic matching methods (i.e., KnowBERT [33], Genre [34]). Semantic matching methods extend entity mentions with a pretrained language model and additional rules. For sentence 1 In Figure 1, semantic matching methods link the knowledge base entity United_states with location type to the entity mention United States. Semantic matching methods also detect the correct boundary of the entity mention Mark Roe by utilizing pretrained language models. This version of labels contains less noise than the one provided in [7], and it is used to validate the generalization of the proposed method.

### *4.2. Implementation Details*

The proposed SALO was iteratively optimized for 30 steps. For each step, the implicit denoising module is firstly pretrained for 500 iterations, then the remainder modules for one epoch. The batch size was set to 16 and the learning rates were $1 \times 10^{-2}$ and $1 \times 10^{-5}$ for BiLSTM and RoBERTa series, respectively. We set $K = 2$ folds in the implicit denoising module for simplicity. The codes are implemented upon HuggingFace (https://huggingface.co/transformers/, accesson on 24 April 2022).

### *4.3. Comparison with State of the Arts*

The proposed method was compared with two groups of baselines, i.e., fully supervised NER models and distantly supervised NER models.

The fully supervised NER methods are taken as the upper bound of these datasets. BiLSTM-CRF [35] effectively captures the sequential relationships among the input tokens

and jointly predicts labels in the sentence. RoBERTa-base [36] is a robustly optimized BERT pretraining approach.

Distantly supervised NER approaches explore tackling the noisy NER issue from different perspectives. AutoNER [15] adopts a revised fuzzy CRF layer to handle noisy labels with a new Tie or Break scheme. LRNT [23] is a name-tagging model solely based on weakly labeled data, which focuses on the effective portions of noisy data and utilizes Partial-CRFs to achieve global optimum. Self-training [22] is a self-training framework with marginal CRF to reannotate the missing labels. Liu et al. [27] propose a calibration method to estimate the confidence of entity labels with prior noise ratio and integrate a self-training framework to boost performance (denoted as Confidence Estimation). Bond [7] also adapts self-training directly to noisy labels. SCDL [28] jointly trains two teacher–student networks in a mutually beneficial manner to iteratively perform noisy label refinery. BiLSTM-CRF [35] and RoBERTa-base [36], which are directly trained on noisy data, are also included, which can be viewed as the lower bound of these datasets. The base model architecture of Self-Training and Confidence Estimation is BiLSTM-CRF, while that of Bond and SCDL is RoBERTa.

Table 1 presents the performance comparison of the proposed SALO against other methods. From the table, there are several observations as follows:

- The proposed SALO outperforms existing distantly supervised approaches on all datasets, which demonstrates the superiority of the built model in noisy NER label learning. An in-depth data analysis reveals that SALO is able to detect accurate boundaries of various few-shot entity mentions such as Lucy Vanderwende and Carolyn Rose in the Webpage dataset compared with BiLSTM-CRF.

**Table 1.** Performance of all methods on four datasets measured by $F_1$ score (Precision/Recall) (in %). Base models directly trained using clean labels and noisy ones can be referred as upper bound (marked with UB) and lower bound (marked with LB).

| Method | CoNLL03 | Twitter | Webpage | Wikigold |
|---|---|---|---|---|
| Fully Supervised | | | | |
| BiLSTM-CRF (UB) | 91.21 (91.35/91.06) | 52.18 (60.01/46.16) | 52.34 (50.07/64.76) | 54.90 (55.40/54.30) |
| RoBERTa-base (UB) | 90.11 (89.14/91.10) | 52.19 (51.76/52.63) | 72.39 (66.29/79.73) | 86.43 (85.33/87.56) |
| Distantly Supervised | | | | |
| BiLSTM-CRF (LB) | 59.50 (75.50/49.10) | 21.77 (46.91/14.18) | 43.34 (58.05/34.59) | 42.92 (47.55/39.11) |
| RoBERTa-base (LB) | 75.93 (82.29/70.47) | 46.45 (50.97/42.66) | 60.98 (59.24/62.84) | 52.57 (47.67/58.59) |
| AutoNER | 67.00 (75.21/60.40) | 26.10 (43.26/18.69) | 51.39 (48.82/54.23) | 47.54 (43.54/52.35) |
| LRNT | 69.74 (79.91/61.87) | 23.84 (46.94/15.98) | 47.74 (46.70/48.83) | 46.21 (45.60/46.84) |
| Self-training | 77.8 (-/-) | 42.3 (-/-) | 49.6 (-/-) | 51.3 (-/-) |
| Confidence-Estimation | 79.4 (-/-) | 43.6 (-/-) | 51.8 (-/-) | 54.0 (-/-) |
| Bond | 81.48 (82.05/80.92) | 48.01 (53.16/44.76) | 65.74 (67.37/64.19) | 60.07 (53.44/68.58) |
| SCDL | 83.69 (87.96/79.82) | 51.09 (59.87/44.57) | 68.47 (68.71/68.24) | 64.13 (62.25/66.12) |
| SALO (BiLSTM-CRF) | 80.08 (85.59/75.24) | 44.96 (54.02/38.50) | 54.90 (74.07/40.54) | 55.71 (53.31/58.33) |
| SALO (RoBERTa) | **84.90** (86.20/83.64) | **52.50** (68.48/42.57) | **69.66** (78.15/62.84) | **65.72** (63.31/68.33) |

- Models directly trained on noisy labels only obtain 41.88% and 58.98% average $F_1$ scores using BiLSTM-CRF and RoBERTa-base architectures, respectively. This reveals that noisy label learning is in demand for distantly supervised NER.
- Models trained in a fully supervised manner achieve upper bound performance, with 62.66% and 75.28% average $F_1$ scores using BiLSTM-CRF and RoBERTa-base architectures, respectively. The proposed SALO narrows the gap between fully supervised and distantly supervised NER methods, obtaining 58.91% and 68.20%, respectively. There still exists 3.75% and 7.08% gaps between fully supervised and distantly supervised models.
- Self-Training and Confidence Estimation are two strong baselines with the BiLSTM-CRF base model under the distantly supervised setting. The proposed SALO with

BiLSTM-CRF architecture obtains 3.66% and 1.71% improvements on average $F_1$ scores over Self-Training and Confidence Estimation, respectively. This improvement could be attributed to the adaptive label correction, which can successfully avoid the overconfidence issue of deep neural networks for noisy data and does not require to know/model the ratio of noise data in the training data like Confidence Estimation.

- Bond and SCDL which are two advanced baselines with the RoBERTa base model under the distantly supervised setting. The proposed SALO counterpart achieves 4.37% and 1.35% improvements on average $F_1$ scores compared with Bond and SCDL, respectively.
- The models that explicitly handle the overconfidence issue (i.e., SALO and Confidence-Estimation) generally perform better than other distantly supervised baselines.

### 4.4. Ablation Study

#### 4.4.1. Effectiveness of Implicit Denoising

The following two variants of SALO are conducted to further validate the effectiveness of implicit noise denoising. The w/o denoising model without the implicit noise denoising module directly generates pseudolabels. The hard denoising model only selects tokens with high confidence predictions with one denoising classifier. As shown in Table 2, SALO consistently achieves a better F score compared with the hard denoising approach. It can be attributed to the denoising ability of the proposed approach by retrieving the entities in the training set and quantifying the fluctuation of noisy labels. Not surprisingly, the w/o denoising method is much worse than the other two models, showing that disabling the noise denoising module raises challenges in handling incomplete annotations.

**Table 2.** The effectiveness of implicit noise denoising on Webpage: Precision, Recall and $F_1$ score (in %). Basemodel: BiLSTM-CRF.

| Method | Precision | Recall | $F_1$ |
|---|---|---|---|
| w/o denoising | 61.54 | 27.03 | 37.56 |
| hard denoising | 64.15 | 45.95 | 53.54 |
| SALO | 65.42 | 47.30 | 54.90 |

#### 4.4.2. Effectiveness of Noisy Example Detector

The following variants of SALO are conducted to further validate the effectiveness of the noisy example detector. The negative detector is exploited to search for negative noisy examples (Negative: The O is recognized as non-O tag) to facilitate a more effective training example. Otherwise, the positive detector aims at augmenting the training set via detecting positive noisy examples (Positive: All entity mention tags). The random detector $(\alpha, 1-\alpha)$ corrects all predictions of the denoising model via $\epsilon = \alpha, \hat{y} = \alpha * \dot{y} + (1-\alpha) * v$. $\alpha$ is a hyperparameter learned by self-adaptive training. The random detector$(\alpha, \beta)$ constructs pseudolabels via randomization, $\epsilon = \alpha, 1-\epsilon = \beta, \hat{y} = \alpha * \dot{y} + \beta * v$. $\alpha$ and $\beta$ are two hyperparameters learned by self-adaptive training.

Table 3 summarizes the experimental results. SALO achieves up to 3.84% and 2.83% $F_1$ score improvements compared with the random detector $(\alpha, 1-\alpha)$ and random detector $(\alpha, \beta)$, respectively, which verifies that the proposed method is able to distinguish challenging problematic labels. The positive detector achieves a 53.54% $F_1$ score, which is narrowly worse than the best $F_1$ score on Webpage 54.90 (SALO). The negative detector obtains the worst performance of all other detectors, especially leading a significant drop compared with the positive detector. The results confirm that SALO can successfully detect the corrupted labels of these highly difficult entity mentions and also demonstrates the top gains of SALO come from detecting problematic positive examples. Accurately correcting corrupted labels of entity mentions makes a much larger impact than rectifying the error O tags.

**Table 3.** The effectiveness of noisy example detector on Webpage: Precision, Recall and $F_1$ score (in %). Basemodel:BiLSTM-CRF.

| Method | Precision | Recall | $F_1$ |
|---|---|---|---|
| Negative Detector | 61.54 | 27.03 | 37.56 |
| Positive Detector | 64.15 | 45.95 | 53.54 |
| Random Detector ($\alpha$, $1-\alpha$) | 68.97 | 40.54 | 51.06 |
| Random Detector ($\alpha$, $\beta$) | 67.02 | 42.57 | 52.07 |
| SALO | 65.42 | 47.30 | 54.90 |

#### 4.4.3. Effectiveness of Pseudolabel Estimator

It is beneficial to understand how the adaptive label correction contributes to learning more robust models during training. The proposed method was compared with various soft label generation methods to demonstrate how the proposed pseudolabel estimator works better. The hard label regards the prediction of the denoising model as the pseudolabels, $v = \dot{y}$. Reweighting (confidence) re-estimates pseudolabels via confidence reweighting, $v = \frac{\sum p_{ij}}{k}$. (The confidence $p_{ij}$ is the probabilities of all classes for each token.)

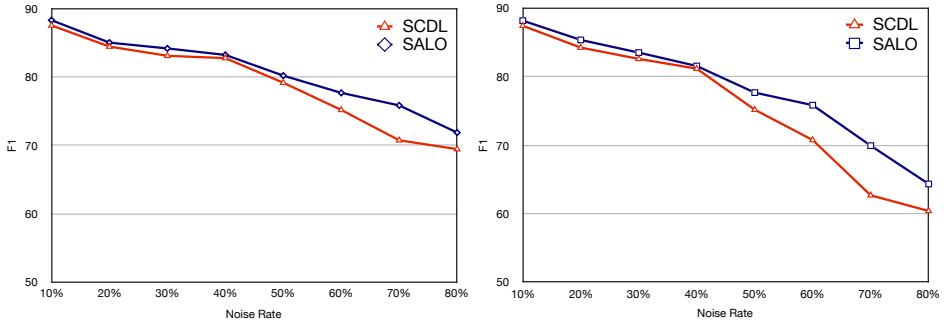
From Table 4, it can be observed that the proposed pseudolabel estimator improves the $F_1$ score and recall on Webpage. Specifically, the $F_1$ score and recall increase from 53.17%/45.27% and 52.85%/43.92% to 54.90%/47.30%, compared with hard label and reweighting (confidence), respectively. We believe that this is because hard corrupted labels mislead NER models; correcting them via pseudolabel estimator restores sufficient boundary or type information learned from noisy labels.

**Table 4.** The effectiveness of pseudolabel estimator on Webpage: Precision, Recall and $F_1$ score (in %). Basemodel:BiLSTM-CRF.

| Method | Precision | Recall | $F_1$ |
|---|---|---|---|
| Hard Label | 64.42 | 45.27 | 53.17 |
| Reweighting (Confidence) | 66.33 | 43.92 | 52.85 |
| SALO | 65.42 | 47.30 | 54.90 |

#### 4.4.4. Robustness to Different Noise Ratio

Figure 3 compares the performance of the state-of-the-art method SCDL and the proposed SALO under different noise ratios of two noise types, i.e., span noise and category noise. The data is constructed by randomly replacing/removing the gold tags of labeled dataset CoNLL03. SALO consistently outperforms SCDL under all noise levels, and the superiority becomes more significant when the noise ratio grows large, demonstrating the effectiveness of the proposed method. This is because with both implicit noise denoising and explicit noise correction modules, the proposed SALO is able to learn from existing patterns from correct tags and alleviate the fluctuations of noisy tags. Additionally, category noise leads to a larger performance drop than span noise under the same noise ratio, which implies that recovering form category noise is more challenging for noisy label learning algorithms.

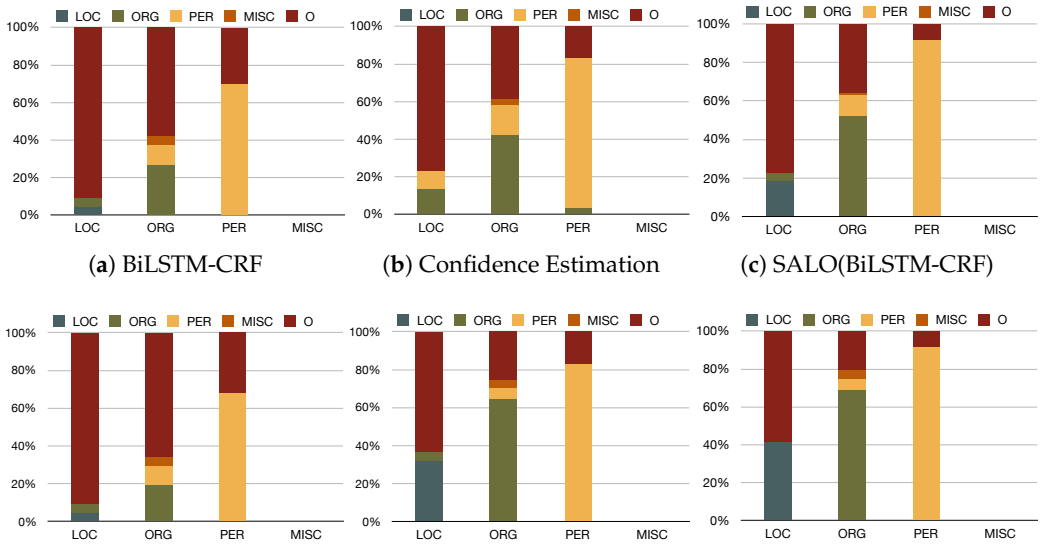



(**a**) (**b**)

**Figure 3.** Robustness to different noise ratios of two types of noise. Incomplete noise: tagging some entities to "O" tags and Inaccurate noise: replacing one type entity tag to another entity tag. (**a**) Span noise; (**b**) Category noise.

### *4.5. Case Study*

To further investigate how SALO improves performance, the prediction performance of distantly supervised models was compared with SALO on the noisy dataset Webpage. As can be seen in Figure 4, the distantly supervised model BiLSTM-CRF yields the largest amount of "O" (nonentity), which discloses their limited generalization ability. The distantly supervised model Confidence Estimation has a better performance gain in "ORG" and "PER" tags than BiLSTM-CRF, yet fails to recognize the "LOC" tag. In contrast, the proposed SALO achieves the best performance on all three tags, which proves that the built model has the ability to correct mismatched "O"s to their corresponding tags, especially in the "LOC" tag. Compared with the NER models with BiLSTM, RoBERTa-based models illustrate a similar increasing trend. Digging into the Webpage dataset, the RoBERTa-based model performs better on a few shot "LOC" tags and "ORG" tags. It is supposed that a pretrained language model with self adaptive training has the ability of transferring rich semantic and contextual information, thus benefiting NER performance.




(**a**) BiLSTM-CRF (**b**) Confidence Estimation (**c**) SALO(BiLSTM-CRF)

(**d**) RoBERTa-CRF (**e**) SCDL (**f**) SALO(RoBERTa-CRF)

**Figure 4.** Case Study. The horizontal axis represents the true entity type. The segments in a bar describe the portions of the entities being classified into different entities. The base NER model of (**a**–**c**) is BiLSTM, while that of (**d**–**f**) is RoBERTa.

### 4.6. Results on Label Extensions

Additional experiments were carried out on extended labels. From Tables 5 and 6, it can be observed that models trained using the extended labels consistently improve the $F_1$ score on all four datasets, indicating that such an extension method combining semantic entity linking may preserve more correct labels. Particularly, the average $F_1$ score improves 1.72%, 1.13%, 2.63% and 1.62% on four datasets with BiLSTM-CRF base architecture. We will release the extended labels and hope they benefit future research on this topic. The proposed SALO obtained remarkable performance on two versions of training labels, showing its generalization ability.

**Table 5.** Comparison of BiLSTM-CRF baseline and SOLO on CoNLL03, Tweet, Webpage and Wikigold datasets, in terms of $F_1$ score, Precision and Recall (in %). Results using original noisy labels and extended labels (denoted with -e) are both presented.

| Method | BiLSTM-CRF (Baseline) | | | BiLSTM-CRF (SALO) | | |
|---|---|---|---|---|---|---|
| Dataset | $F_1$ Score | Precision | Recall | $F_1$ Score | Precision | Recall |
| CoNLL03 | 59.50 | 75.50 | 49.10 | 80.08 | 85.59 | 75.24 |
| CoNLL03-e | 61.22 | 85.67 | 47.63 | 81.95 | 88.83 | 77.41 |
| Twitter | 21.77 | 46.91 | 14.18 | 44.96 | 54.02 | 38.50 |
| Twitter-e | 22.90 | 55.16 | 14.45 | 45.11 | 60.92 | 35.81 |
| Webpage | 43.34 | 58.05 | 34.59 | 54.90 | 74.07 | 40.54 |
| Webpage-e | 45.97 | 57.00 | 38.51 | 56.07 | 73.63 | 45.27 |
| Wikigold | 42.92 | 47.55 | 39.11 | 55.71 | 53.31 | 58.33 |
| Wikigold-e | 44.54 | 51.84 | 39.04 | 56.13 | 67.96 | 47.97 |

**Table 6.** Comparison of RoBERTa baseline and SOLO on CoNLL03, Tweet, Webpage and Wikigold datasets, in terms of $F_1$ score, Precision and Recall (in %). Results using original noisy labels and extended labels (denoted with -e) are both presented.

| Method | RoBERTa (Baseline) | | | RoBERTa (SALO) | | |
|---|---|---|---|---|---|---|
| Dataset | $F_1$ Score | Precision | Recall | $F_1$ Score | Precision | Recall |
| CoNLL03 | 75.93 | 82.29 | 70.47 | 84.90 | 86.20 | 83.64 |
| CoNLL03-e | 76.39 | 86.82 | 68.20 | 84.51 | 87.81 | 81.45 |
| Twitter | 46.45 | 50.97 | 42.66 | 52.50 | 68.48 | 42.57 |
| Twitter-e | 48.56 | 53.51 | 44.45 | 53.79 | 46.14 | 64.47 |
| Webpage | 60.98 | 59.24 | 62.84 | 69.66 | 78.15 | 62.84 |
| Webpage-e | 61.33 | 58.31 | 64.69 | 70.01 | 71.17 | 69.05 |
| Wikigold | 52.57 | 47.67 | 58.59 | 65.72 | 63.31 | 68.33 |
| Wikigold-e | 53.79 | 46.14 | 64.47 | 66.98 | 65.11 | 68.97 |

### 4.7. Analysis

Finally, the reason that SALO achieves performance improvement was analyzed. To this end, experiments were conducted on the CoNLL03 dataset and revealed that NER performance is highly correlated with the noise rate of training labels. As shown in Table 7, when using the baseline BiLSTM-CRF model on the clean (noise rate 0%) and noisy (noise rate 67.21%) labels, we obtain $F_1$ scores of 91.21% and 59.50%, respectively. Using implicit denoising models to repredict labels slightly alleviates label noise, with a noise rate of 63.33%. With the identical BiLSTM-CRT, the $F_1$ score improves to 66.67%. The proposed SALO obtains state-of-the-art performance, with 80.08% in terms of $F_1$ score. After the label correction, the noise rate is further reduced to 48.43%. This indicates that distinguishing from correct and wrong labels is crucial for noisy label learning and the built model manager to alleviate the fluctuations of noisy labels.

**Table 7.** Correlations between $F_1$ score (in %) and Noisy Rate (in %) of training labels with different noise label refinery strategies on CoNLL03.

| Clean Label | Noisy Label | Implicit Denoising | SALO | $F_1$ | Noise Rate |
|---|---|---|---|---|---|
| √ | | | | 91.21 | 0.00 |
| | √ | | | 59.50 | 67.21 |
| | √ | √ | | 66.67 | 63.33 |
| | √ | | √ | 80.08 | 48.43 |

## 5. Conclusions and Future Work

In this paper, a unified distantly supervised NER framework termed SALO is proposed. Different from prevailing approaches that solely discarding noisy data or reweighting samples, this work explores to make better use of the mislabeled data. Specifically, an automatic label correction mechanism is introduced to simultaneously identify mislabeled data and recover a reasonable approximation of ground-truth labels by exploiting the correlation and context awareness among distant labels. Experiments on four challenging datasets demonstrate that such a strategy is more effective than simply discarding or reweighting strategies, leading to state-of-the-art performance. Furthermore, a better version of noisy labels by ensembling several semantic matching methods was constructed. The proposed SALO consistently surpasses other baselines, demonstrating the robustness and generalization of the proposed SALO.

In the future, we would explore more effective label correction mechanisms utilizing knowledge base, further boosting NER performance. Moreover, the techniques proposed in this paper are generalizable, and we would explore the potential usage in related topics, e.g., relation extraction, entity linking and event extraction.

**Author Contributions:** Conceptualization, B.N.; Formal analysis, B.N.; Methodology, B.N.; Writing—original draft, B.N.; Writing—review & editing, C.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by Scientific research start-up fund for Hangzhou Dianzi University (No. KYS335621031), the Fundamental Research Funds for Zhejiang Province (GK229909299001-022).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; Dyer, C. Neural Architectures for Named Entity Recognition. In Proceedings of the North American Chapter of the Association for Computational Linguistics, San Diego, CA, USA, 12–17 June 2016; pp. 260–270.
2. Chen, H.; Lin, Z.; Ding, G.; Lou, J.; Zhang, Y.; Karlsson, B. GRN: Gated Relation Network to Enhance Convolutional Neural Network for Named Entity Recognition. In Proceedings of the National Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 6236–6243.
3. Liu, Y.; Meng, F.; Zhang, J.; Xu, J.; Chen, Y.; Zhou, J. GCDT: A Global Context Enhanced Deep Transition Architecture for Sequence Labeling. In Proceedings of the Association for Computational Linguistics, Florence, Italy, 4–13 October 2019; pp. 2431–2441.
4. Chen, M.; Lan, G.; Du, F.; Lobanov, V.S. Joint Learning with Pre-trained Transformer on Named Entity Recognition and Relation Extraction Tasks for Clinical Analytics. In Proceedings of the Empirical Methods in Natural Language Processing, Virtual, 16–20 November 2020; pp. 234–242.
5. Vrandečić, D.; Krötzsch, M. Wikidata: A free collaborative knowledgebase. *Commun. ACM* **2014**, *57*, 78–85. [CrossRef]
6. Tanon, T.P.; Weikum, G.; Suchanek, F.M. YAGO 4: A Reason-able Knowledge Base. In Proceedings of the European Semantic Web Conference, Heraklion, Greece, 31 May–4 June 2020; Volume 12123, pp. 583–596.
7. Liang, C.; Yu, Y.; Jiang, H.; Er, S.; Wang, R.; Zhao, T.; Zhang, C. BOND: BERT-Assisted Open-Domain Named Entity Recognition with Distant Supervision. In Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining, Virtual, 6–10 July 2020; pp. 1054–1064.

8. Arpit, D.; Jastrzebski, S.; Ballas, N.; Krueger, D.; Bengio, E.; Kanwal, M.S.; Maharaj, T.; Fischer, A.; Courville, A.C.; Bengio, Y.; et al. A Closer Look at Memorization in Deep Networks. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; Volume 70, pp. 233–242.
9. Shorten, C.; Khoshgoftaar, T.M. A survey on Image Data Augmentation for Deep Learning. *J. Big Data* **2019**, *6*, 60. [CrossRef]
10. Krogh, A.; Hertz, J.A. A single weight decay can imrpove generalization. In Proceedings of the Advances in Neural Information Processing Systems, Denver, CO, USA, 30 November–3 December 1992; pp. 950–957.
11. Srivastava, N.; Hinton, G.E.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
12. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; Volume 37, pp. 448–456.
13. Bellare, K.; McCallum, A. Learning extractors from unlabeled text using relevant databases. In Proceedings of the International Workshop on Information Integration on the Web, Linz, Austria, 29 November–1 December 2007.
14. Mayhew, S.; Chaturvedi, S.; Tsai, C.; Roth, D. Named Entity Recognition with Partially Annotated Training Data. In Proceedings of the Conference on Natural Language Learning, Hong Kong, China, 3–4 November 2019; pp. 645–655.
15. Shang, J.; Liu, L.; Gu, X.; Ren, X.; Ren, T.; Han, J. Learning Named Entity Tagger using Domain-Specific Dictionary. In Proceedings of the Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 2054–2064.
16. Zheng, S.; Wu, P.; Goswami, A.; Goswami, M.; Metaxas, D.N.; Chen, C. Error-Bounded Correction of Noisy Labels. In Proceedings of the International Conference on Machine Learning, Virtual, 13–18 July 2020; Volume 119, pp. 11447–11457.
17. Rau, L.F. Extracting company names from text. In Proceedings of the ICAISA, Miami Beach, FL, USA, 24–28 February 1991; pp. 29–32.
18. Zhou, G.; Su, J. Named Entity Recognition using an HMM-based Chunk Tagger. In Proceedings of the Association for Computational Linguistics, Philadelphia, PA, USA, 7–12 July 2002; pp. 473–480.
19. Malouf, R. Markov Models for Language-independent Named Entity Recognition. In Proceedings of the Conference on Natural Language Learning, Taipei, Taiwan, 24 August–1 September 2002.
20. Li, Y.; Bontcheva, K.; Cunningham, H. SVM Based Learning System for Information Extraction. In Proceedings of the Deterministic and Statistical Methods in Machine Learning, First International Workshop, Sheffield, UK, 7–10 September 2004; Volume 3635, pp. 319–339.
21. Ratinov, L.; Roth, D. Design Challenges and Misconceptions in Named Entity Recognition. In Proceedings of the Conference on Natural Language Learning, Boulder, CO, USA, 4–5 June 2009; pp. 147–155.
22. Jie, Z.; Xie, P.; Lu, W.; Ding, R.; Li, L. Better Modeling of Incomplete Annotations for Named Entity Recognition. In Proceedings of the North American Chapter of the Association for Computational Linguistics, Minneapolis, MN, USA, 2–7 June 2019; pp. 729–734.
23. Cao, Y.; Hu, Z.; Chua, T.; Liu, Z.; Ji, H. Low-Resource Name Tagging Learned with Weakly Labeled Data. In Proceedings of the Empirical Methods in Natural Language Processing-International Joint Conference on Natural Language Processing, Hong Kong, China, 3–7 November 2019; pp. 261–270.
24. Yang, Y.; Chen, W.; Li, Z.; He, Z.; Zhang, M. Distantly Supervised NER with Partial Annotation Learning and Reinforcement Learning. In Proceedings of the International Conference on Computational Liguistics, Santa Fe, New Mexico, 20–26 August 2018; pp. 2159–2169.
25. Peng, M.; Xing, X.; Zhang, Q.; Fu, J.; Huang, X. Distantly Supervised Named Entity Recognition using Positive-Unlabeled Learning. In Proceedings of the Association for Computational Linguistics, Florence, Italy, 4–13 October 2019; pp. 2409–2419.
26. Zhang, H.; Long, D.; Xu, G.; Zhu, M.; Xie, P.; Huang, F.; Wang, J. Learning with Noise: Improving Distantly-Supervised Fine-grained Entity Typing via Automatic Relabeling. In Proceedings of the International Joint Conference on Artificial Intelligence, Yokohama, Japan, 7–15 January 2020; pp. 3808–3815.
27. Liu, K.; Fu, Y.; Tan, C.; Chen, M.; Zhang, N.; Huang, S.; Gao, S. Noisy-Labeled NER with Confidence Estimation. In Proceedings of the North American Chapter of the Association for Computational Linguistics, Virtual, 6–11 June 2021; pp. 3437–3445.
28. Zhang, X.; Yu, B.; Liu, T.; Zhang, Z.; Sheng, J.; Xue, M.; Xu, H. Improving Distantly-Supervised Named Entity Recognition with Self-Collaborative Denoising Learning. In Proceedings of the Empirical Methods in Natural Language Processing, Punta Cana, Dominican Republic, 7–11 November 2021.
29. Huang, J.; Qu, L.; Jia, R.; Zhao, B. O2u-net: A simple noisy label detection approach for deep neural networks. In Proceedings of the International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 3325–3333.
30. Sang, E.F.T.K.; Meulder, F.D. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In Proceedings of the North American Chapter of the Association for Computational Linguistics, Mexico City, Mexico, 16 February 2003; pp. 142–147.
31. Godin, F.; Vandersmissen, B.; Neve, W.D.; de Walle, R.V. Multimedia Lab @ ACL WNUT NER Shared Task: Named Entity Recognition for Twitter Microposts using Distributed Word Representations. In Proceedings of the Workshop on the Association for Computational Linguistics, Beijing, China, 30–31 July 2015; pp. 146–153.
32. Balasuriya, D.; Ringland, N.; Nothman, J.; Murphy, T.; Curran, J.R. Named Entity Recognition in Wikipedia. In Proceedings of the Workshop on The People's Web Meets, Singapore, 7 August 2009; pp. 10–18.

33. Peters, M.E.; Neumann, M.; Logan, R.L.; Schwartz, R.; Joshi, V.; Singh, S.; Smith, N.A. Knowledge Enhanced Contextual Word Representations. In Proceedings of the Empirical Methods in Natural Language Processing-International Joint Conference on Natural Language Processing, Hong Kong, China, 3–7 November 2019; pp. 43–54.
34. Cao, N.D.; Izacard, G.; Riedel, S.; Petroni, F. Autoregressive Entity Retrieval. In Proceedings of the International Conference on Learning Representations, Virtual, 3–7 May 2021.
35. Ma, X.; Hovy, E.H. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In Proceedings of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016.
36. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**, arXiv:1907.11692.