Reweighting in Conditional Random Fields using an Expert-Domain Dictionary

Shusuke Tatsumi[1] Pontus Stenetorp[2] Yuji Matsumoto[1,3]
Graduate School of Science and Technology, NAIST[1]
Department of Computer Science, University College London[2]
RIKEN Center for Advanced Intelligence Project[3]
{tatsumi.shusuke.tk0, matsu}@is.naist.jp
p.stenetorp@cs.ucl.ac.uk

1 Introduction

For Named Entity Recognition (NER) in specialized fields, it is often the case that one lacks access to gold data. Therefore, many studies have explored techniques such as noise reduction of 'incomplete data' where some of the named entities are given false negative labels. In these studies, they assign multiple labels (all possible label candidates) to tokens with incomplete annotations. The model improves the recognition performance by learning probability distributions over the given labels. However, these models still face three main challenges. The first problem is that confirmation bias causes noise propagation. These models perform sequence labeling learning and noise reduction in the same process and with the same resources. Therefore, the noise at the beginning of learning propagates to the latter stages of learning. The second problem is the trade-off between Precision and Recall. When applying noise reduction using multiple label, Recall improves, but Precision decreases. The third problem is false positives in the model prediction, due to mismatches between the resource(s) used to create the multiple labels for training relative to the evaluation data.

For these problems, we propose the following strategies. As a strategy for problem 1, we reduce noise from different processes and resources. Specifically, we improve the weighting process. We use 'entity linking, knowledge graph embedding and clustering' as another process, and 'hierarchical dictionary' as another resource. As a strategy for problem 2, we reduce false positive and improve Precision without decreasing Recall. As a strategy for problem 3, we recognize word sets targeted by evaluation data from various word sets on the dictionary based on 'hierarchical structure of dictionary' and 'incomplete data'. We describe the contribution of this paper below.

- We proposed a method for reweighting conditional random fields (CRF) using a dictionary. This method consists of two elements, 'weighting based on entity linking' and 'weighting based on dictionary hierarchy'.
- Our method exceeded the baseline in two of the three datasets.
- 'Entity linking' was effective, but the effect of 'dictionary hierarchy' was limited.

2 Previous work

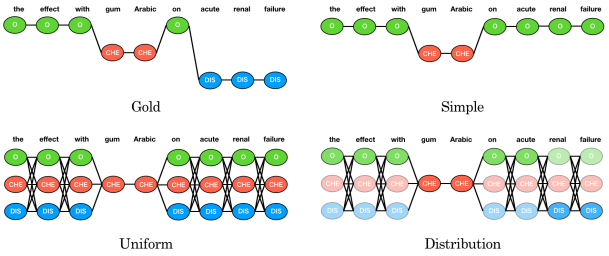

Figure 1: Label path for related research. In this example, the named entities 'acute', 'renal', and 'failure' are false negative. Gold is the gold data path. In Simple, we give the false negative an 'O' label. In Uniform, we assign probability scores equally to multiple labels. In Distribution, we assign optimal probability scores to the multiple labels, so that they are close to the Gold Pass. We indicate the level of the probability scores by the shading of the label in the figure.

NER is a sequence labeling task that takes a word sequence X as input and outputs a label sequence y such as BIOES. Linear-Chain CRF[4] is a kind of this sequence labeling model. It can be trained on fully labeled data (single-label sequences (Figure 1 $GoldandSimple$)) to minimize the loss function Eq.1.

$$L(w) = -\sum_i log\, p_w(y_i \mid X_i) \quad (1)$$

Next, we consider a method for processing $y_i^{possible}$ which is label sequence of incomplete data. In a

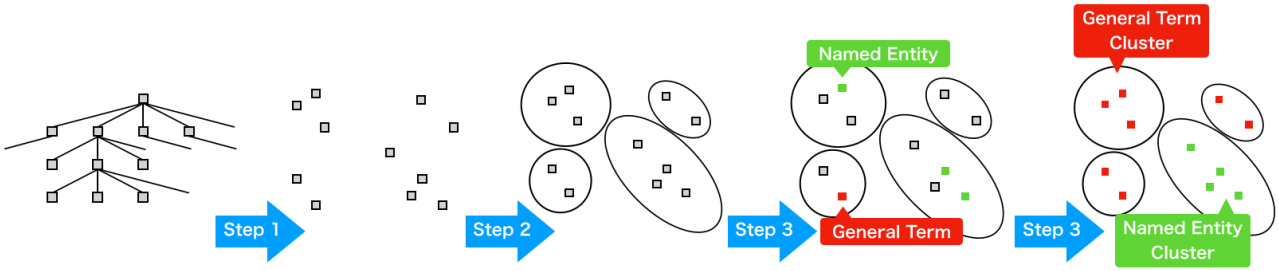

Figure 2: Discrimination based on hierarchical structure

previous study[1], as in Figure 1 *Uniform*, they assign probability scores equally to all multiple labels $C\left(y_i^{possible}\right)$. The loss function is Eq.2.

$$L\left(w\right) = -\sum_i log \sum_{y \in C\left(y_i^{possible}\right)} p_w\left(y \mid X_i\right) \quad (2)$$

On the other hand, in previous study[3], as shown in Figure 1 *Distribution*, q assigns the optimal probability scores to the multiple labels so that the path gets to be closer to the gold path. We use this model as the base model. Eq.3 describes the loss function.

$$L\left(w\right) = -\sum_i log \sum_{y \in C\left(y_i^{possible}\right)} q_D\left(y \mid X_i\right) p_w\left(y \mid X_i\right) \quad (3)$$

2.1 Estimation q

According to previous study[3], we find the probability distribution q (Eq.4) of the label y for the word X_i by k-cross validation. In the Viterbi algorithm, we estimate the score $P_{i,\,y_j}$ of X_i with y_j in Eq.5. Φ indicates the transition probability score from label y_i to label y_{i+1}. As shown in Eq.6, y_j is composed of a label set that combines a range label of the named entity and a class type of the named entity, plus a general term label.

$$q_D\left(y \mid X_i\right) = \frac{e^{s(X_i, y)}}{\sum_{y \in C(y^{possible})} e^{s(X_i, y)}} \quad (4)$$

$$s\left(X, y\right) = \sum_{i=0}^{n} \Phi_{y_i, y_{i+1}} + \sum_{i=1}^{n} P_{i, y_i}, \quad \Phi_{y_i, y_{i+1}},\ P_{i, y_i} \in R \quad (5)$$

$$y_j \in [\{B, I, E, S\} \times \{Class\ Types\} + \{O\}] \quad (6)$$

3 Proposed method

3.1 Reweighting

After calculating $P_{i,\,y_j}$ in the base model, we reweight using the dictionary. Intuitively, we adjust the weights so that when the dictionary identifies X_i as a general term, the label y_i is more likely to be predicted as 'O'. Specifically, as shown by Eq.7, we add the hyperparameter α to the score of label 'O' among all multiple labels of label y_i . Dictionary identification consists of two steps as shown in *Algorithm* 1.

First, we link the entity X_i to the dictionary. We determine that X_i, which could not be linked, is a general term. Then we reweight with Eq.7. Next, when X_i can be linked, we identify based on the hierarchical structure. Specifically, we reweight only those terms that are determined to be general terms with Eq.7.

Algorithm 1 Reweigh

Require:
$Words = [X_0,\ X_1,\ ...,\ X_n]$

```
procedure REWEIGHT(Words)
    for all Words do
        if Word.label is Incomp then
            Linked ⇐ LINK(Word.character)
            if Linked is Unlinked then
                WEIGHT(Word.labels)
            else
                Identified ⇐ IDENTIFY(Word)
                if Identified is General then
                    WEIGHT(Word.labels)
```

Also, we found that if we continue reweighting until the end of training, the recall will not improve because of the influence of the dictionary biases the model too strongly towards precision. Therefore, we reweight more early in the training and less at the end of the training. We reduce the value of α, as shown in Eq.8. η is the hyperparameter, α_{init} is the initial value of α, and *epoch_count* is the number of epochs completed.

$$P\left(X_i, y_{j='O'}\right) = P\left(X_i, y_{j='O'}\right) + \alpha, \quad P\left(X_i, y_{j='O'}\right),\ \alpha \in R \quad (7)$$

$$\alpha = max\left(0,\ \alpha_{init} - \eta \cdot epoch\_count\right) \quad (8)$$

3.2 Entity linking to dictionary

We link entities based on Levenshtein distance. We associate X_i with the dictionary word with the shortest distance among the dictionary words whose Levenshtein distance from X_i is MIN_DISTANCE or less. If we cannot find such a dictionary word, we will not associate it. Also, we do not link the word X_i, whose word length is less than MIN_CHARACTER, to the dictionary because it has too many linking candidates.

3.3 Hierarchical identification

First, we describe the preparation stage. We identify whether the word X_i is a general term or a named entity, separately from the sequence labeling model. Therefore, we prepare in advance using a hierarchical dictionary and incomplete data (4.2). It consists of three steps. In the first step, we vectorize the hierarchical structure of the dictionary using knowledge graph embedding. In the second step, we perform unsupervised clustering using all named entity vectors as instances. In the third step, we classify all clusters into general term clusters and named entity clusters using incomplete data. Specifically, we classify each cluster as a named entity cluster if the proportion of named entities in the incomplete data exceeds a threshold μ, and as a general term cluster otherwise.

Next, we describe the inference stage. We link the word X_i to the dictionary and find the cluster to which the linked word belongs. We identify X_i as a general term if the cluster is a general term cluster, and as a named entity if it is a named entity cluster.

4 Experiment settings

4.1 Datasets and Dictionary

We evaluate with the NER datasets of the life sciences, BC5CDR [1], NCBI-Disease[2] and CHEMDNER[3]. In CHEMDNER, we evaluate only TRIVIAL, SYSTEMATIC, and FAMILY among all classes, and evaluate others as general terms. We show the datasets configuration in Table 1.

We use CTD, the database of life sciences, for the expert domain dictionary. CTD Chemical vocabularies contain 172,861 words, and CTD Disease vocabularies contains 12,984 words.

[1]http://www.biocreative.org/tasks/biocreative-v/track-3-cdr/

[2]https://www.ncbi.nlm.nih.gov/research/bionlp/Data/disease/

[3]http://www.biocreative.org/resources/biocreative-iv/chemdner-corpus/

Table 1: Datasets configuration

	BC5CDR	NCBI-Disease	CHEMDNER
Domain	Biomedical	Biomedical	Biomedical
Entity Types	Disease, Chemical	Disease	Chemical
Articles	1,500	793	10,000
Mentions	15,935 Chemical 12,852 Disease	6,892 Disease	25,610 Trivial 19,138 Systematic 11,935 Family

4.2 Creating incomplete data

We randomly remove a certain number of named entities from the named entity sets of the gold data, and label them as 'O'. We express the proportion of correctly annotated named entities as 'named entity coverage ρ'. For example, if $\rho = 0.6$, we keep 60 % of all named entities, remove the remaining 40 %, and give them an 'O' label.

4.3 Configuration details

For sequence labeling, we set $k : 2$, $\alpha_{init} : 5.0$, $\eta : 0.5$. In entity linking, we set $MIN\_DISTANCE = 5$, $MIN\_CHARACTER = 7$, and are case insensitive. For the hierarchical discrimination, we set $\mu : 0.001$, use TransE[2] for knowledge graph embedding, and k-means for unsupervised clustering. We use Precision, Recall, and F1 as evaluation metrics.

4.4 Comparison method

Linear-Chain BiLSTM-CRF[5] processes all words with a single label. We give the missing entity a label 'O'. Separately, we also evaluate the performance of models trained on gold data.

Multi Label BiLSTM-CRF[3] is our base model. We assign single labels to named entities which are kept and multiple label to other words.

We compare three of our models. Ours-Linking does not use a hierarchical structure and reweights only with linking. Ours-Fixed α does not decrease α but fixes it. Ours-All uses all the features proposed. We summarize these in Table 3.

In addition, in the Multi Label BiLSTM-CRF and the our models, we initialize the label by the style shown in Figure 1 *Simple* according to previous study[3]. We use the result to initialize q.

5 Experimental result

5.1 Comparison of NER performance

We show the results of experiments with $\rho = 0.5$ in Table 2. First, in comparison with the baseline trained on incomplete data, our method outperformed the baseline in terms of F1 in two out of three datasets. Next, we compared our method to a model trained on gold data. Our method was comparable

Table 2: Comparison of NER performance

	Annotation	BC5CDR			NCBI-Disease			CHEMDNER		
		Pre.	Rec.	F1	Pre.	Rec.	F1	Pre.	Rec.	F1
Linear Chain BiLSTM-CRF	Gold	88.18	87.58	87.88	81.95	82.29	82.12	89.37	85.57	87.43
Linear Chain BiLSTM-CRF	Incomplete	78.79	74.63	76.65	**82.09**	60.62	69.74	**89.46**	66.39	76.21
Multi Label BiLSTM-CRF	Incomplete	74.03	**85.51**	79.35	72.54	**77.60**	74.79	77.06	**81.17**	79.06
Ours-Linking	Incomplete	81.06	83.80	82.41	79.46	76.15	**77.77**	80.75	78.33	**79.52**
Ours-Fixed α	Incomplete	**84.01**	78.78	81.31	80.48	63.13	70.75	83.47	72.87	77.82
Ours-All	Incomplete	80.50	84.63	**82.51**	77.42	76.77	77.09	79.63	78.21	78.91

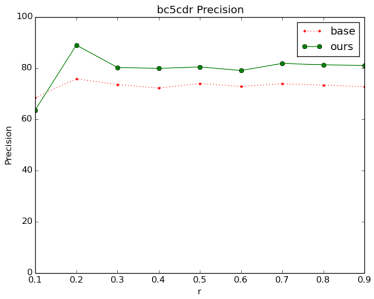

Figure 3: Precision and ρ

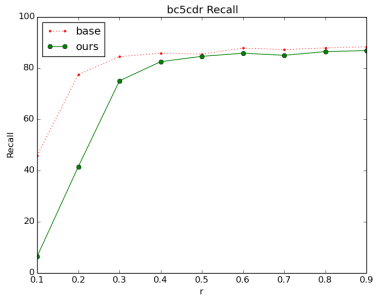

Figure 4: Recall and ρ

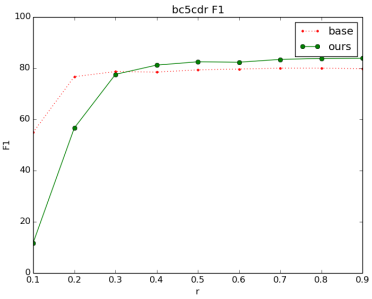

Figure 5: F1 and ρ

Table 3: Function comparison of our models

Model	Decreasing α	Entity Linking	Hierarchical Structure
Ours-Linking	○	○	×
Ours-Fixed α	×	○	○
Ours-All	○	○	○

to the learning model with gold data, despite learning with incomplete data.

5.2 Effect of decreasing α

We compared Ours-Fixed α with Ours-All and found that α reduction was effective.

5.3 Effect of hierarchical structure

We compared Ours-Linking with Ours-All and analyzed which of linking and hierarchical structure contributed more to the performance of NER. In theory, Ours-All has a higher Precision than Ours-Linking. However, our results showed that Ours-Linking was higher.

5.4 ρ and effects of the our method

We experimented with different $\rho = [0.1, 0.2, ..., 0.9]$ settings and analyzed the correlation between the effect of our method and ρ. We compared Multi Label BiLSTM-CRF and Ours-All with the data set BC5CDR. We show the results in Figures 3-5. Experimental results show that our method is effective when $\rho \geq 0.4$.

6 Conclusion

We proposed a method for reweighting CRF using an expert domain dictionary. Our method exceeded the baseline in two of the three datasets. In addition, 'Linking to the dictionary' was effective, but the effect of 'Discrimination using the hierarchical structure' was limited.

Acknowledgments

The first author worked on this study in UCL with the support of the Ministry of Education 'Tobitate'.

References

[1] K. Bellare and A. Mccallum. Learning extractors from unlabeled text using relevant databases. AAAI, 2007.

[2] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko. Translating embeddings for modeling multi-relational data. NIPS, 2013.

[3] Z. Jie, P. Xie, W. Lu, R. Ding, and L. Li. Better modeling of incomplete annotations for named entity recognition. NAACL, 2019.

[4] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. ICML, 2001.

[5] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and Chris Dyer. Neural architectures for named entity recognition. NAACL, 2016.