End-to-end Biomedical Entity Linking with Span-based Dictionary Matching

Shogo Ujiie[♠] **Hayate Iso**[♡*]
Shuntaro Yada[♠] **Shoko Wakamiya**[♠] **Eiji Aramaki**[♠]
[♠]Nara Institute of Science and Technology [♡]Megagon Labs
{ujiie, yada-s, wakamiya, aramaki}@is.naist.jp
hayate@magagon.ai

Abstract

Disease name recognition and normalization , which is generally called biomedical entity linking, is a fundamental process in biomedical text mining. Recently, neural joint learning of both tasks has been proposed to utilize the mutual benefits. While this approach achieves high performance, disease concepts that do not appear in the training dataset cannot be accurately predicted. This study introduces a novel end-to-end approach that combines span representations with dictionary-matching features to address this problem. Our model handles unseen concepts by referring to a dictionary while maintaining the performance of neural network-based models, in an end-to-end fashion. Experiments using two major datasets demonstrate that our model achieved competitive results with strong baselines, especially for unseen concepts during training.

1 Introduction

Identifying disease names , which is generally called biomedical entity linking, is the fundamental process of biomedical natural language processing, and it can be utilized in applications such as a literature search system (Lee et al., 2016) and a biomedical relation extraction (Xu et al., 2016). The usual system to identify disease names consists of two modules: named entity recognition (NER) and named entity normalization (NEN). NER is the task that recognizes the span of a disease name, from the start position to the end position. NEN is the post-processing of NER, normalizing a disease name into a controlled vocabulary, such as a MeSH or Online Mendelian Inheritance in Man (OMIM).

Although most previous studies have developed pipeline systems, in which the NER model first recognizs disease mentions (Lee et al., 2020; Weber et al., 2020) and the NEN model normalizes the recognized mention (Leaman et al., 2013; Ferré et al., 2020; Xu et al., 2020; Vashishth et al., 2020), a few approaches employ a joint learning architecture for these tasks (Leaman and Lu, 2016; Lou et al., 2017). These joint approaches simultaneously recognize and normalize disease names utilizing their mutual benefits. For example, Leaman et al. (2013) demonstrated that dictionary-matching features, which are commonly used for NEN, are also effective for NER. While these joint learning models achieve high performance for both NER and NEN, they predominately rely on hand-crafted features, which are difficult to construct because of the domain knowledge requirement.

Recently, a neural network (NN)-based model that does not require any hand-crafted features was applied to the joint learning of NER and NEN (Zhao et al., 2019). NER and NEN were defined as two token-level classification tasks, i.e., their model classified each token into IOB2 tags and concepts, respectively. Although their model achieved the state-of-the-art performance for both NER and NEN, a concept that does not appear in training data (i.e., zero-shot situation) can not be predicted properly.

One possible approach to handle this zero-shot situation is utilizing the dictionary-matching features. Suppose that an input sentence "Classic *polyarteritis nodosa* is a systemic vasculitis" is given, where "*polyarteritis nodosa*" is the target entity. Even if it does not appear in the training data, it can be recognized and normalized by referring to a controlled vocabulary that contains "*Polyarteritis Nodosa* (MeSH: D010488)." Combining such looking-up mechanisms with NN-based models, however, is not a trivial task; dictionary matching must be performed at the *entity*-level, whereas standard NN-based NER and NEN tasks are performed at the *token*-level (for example, Zhao et al., 2019).

To overcome this problem, we propose a novel end-to-end approach for NER and NEN that com-

*Work done while at Nara Institute of Science and Technology.

Proceedings of the BioNLP 2021 workshop, pages 162–167

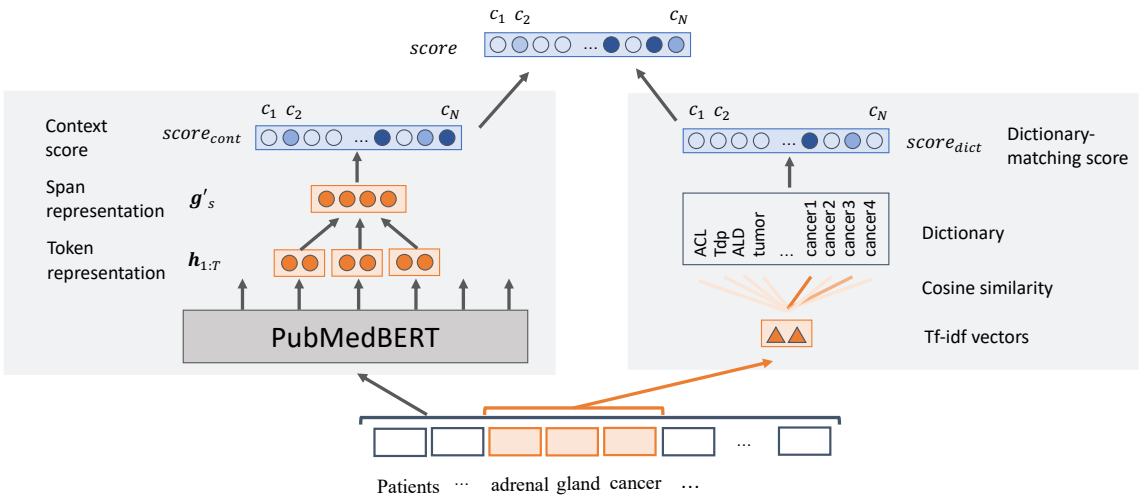

Figure 1: The overview of our model. It combines the dictionary-matching scores with the context score obtained from PubMedBERT. The red boxes are the target span and "ci" in the figure is the "i"-th concept in the dictionary.

bines dictionary-matching features with NN-based models. Based on the span-based model introduced by Lee et al. (2017), our model first computes span representations for all possible spans of the input sentence and then combines the dictionary-matching features with the span representations. Using the score obtained from both features, it directly classifies the disease concept. Thus, our model can handle the zero-shot problem by using dictionary-matching features while maintaining the performance of the NN-based models.

Our model is also effective in situations other than the zero-shot condition. Consider the following input sentence: "We report the case of a patient who developed acute *hepatitis*," where "*hepatitis*" is the target entity that should be normalized to "drug-induced hepatitis." While the longer span "acute hepatitis" also appears plausible for standalone NER models, our end-to-end architecture assigns a higher score to the correct shorter span "hepatitis" due to the existence of the normalized term ("drug-induced hepatitis") in the dictionary.

Through the experiments using two major NER and NEN corpora, we demonstrate that our model achieves competitive results for both corpora. Further analysis illustrates that the dictionary-matching features improve the performance of NEN in the zero-shot and other situations.

Our main contributions are twofold: (i) We propose a novel end-to-end model for disease name recognition and normalization that utilizes both NN-based features and dictionary-matching features; (ii) We demonstrate that combining dictionary-matching features with an NN-based model is highly effective for normalization, especially in the zero-shot situations.

2 Methods

2.1 Task Definition

Given an input sentence, which is a sequence of words $\boldsymbol{x} = \{x_1, x_2, \cdots, x_{|\boldsymbol{X}|}\}$ in the biomedical literature, let us define $\mathcal{S}$ as a set of all possible spans, and $\mathcal{L}$ as a set of concepts that contains the special label $Null$ for a non-disease span. Our goal is to predict a set of labeled spans $\boldsymbol{y} = \{\langle i, j, d \rangle_k\}_{k=1}^{|\boldsymbol{Y}|}$, where $(i, j) \in \mathcal{S}$ is the word index in the sentence, and $d \in \mathcal{L}$ is the concept of diseases.

2.2 Model Architecture

Our model predicts the concepts for each span based on the score, which is represented by the weighted sum of two factors: the context score $score_{cont}$ obtained from span representations and the dictionary-matching score $score_{dict}$. Figure 1 illustrates the overall architecture of our model. We denote the score of the span s as follows:

$$score(s, c) = score_{cont}(s, c) + \lambda score_{dict}(s, c)$$

where $c \in \mathcal{L}$ is the candidate concept and λ is the hyperparameter that balances the scores. For the concept prediction, the scores of all possible spans and concepts are calculated, and then the concept with the highest score is selected as the predicted concept for each span as follows:

$$y = \arg\max_{c \in \mathcal{L}} score(s, c)$$

Context score The context score is computed in a similar way to that of Lee et al. (2017), which is based on the span representations. To compute the representations of each span, the input tokens are first encoded into the token embeddings. We used BioBERT (Lee et al., 2020) as the encoder, which is a variation of bidirectional encoder representations from transformers (BERT) that is trained on a large amount of biomedical text. Given an input sentence containing T words, we can obtain the contextualized embeddings of each token using BioBERT as follows:

$$\mathbf{h}_{1:T} = \mathrm{BERT}(x_1, x_2, \cdots, x_T)$$

where $\mathbf{h}_{1:T}$ is the input tokens embeddings.

Span representations are obtained by concatenating several features from the token embeddings:

$$\mathbf{g}_s = [\mathbf{h}_{start(s)}, \mathbf{h}_{end(s)}, \hat{\mathbf{h}}_s, \phi(s)]$$
$$\mathbf{g'}_s = \mathrm{GELU}(\mathrm{FFNN}(\mathbf{g}_s))$$

where $\mathbf{h}_{start(s)}$ and $\mathbf{h}_{end(s)}$ are the start and end token embeddings of the span, respectively; and $\hat{\mathbf{h}}_s$ is the weighted sum of the token embeddings in the span, which is obtained using an attention mechanism (Bahdanau et al., 2015). $\phi(i)$ is the size of span s. These representations $\mathbf{g}_s$ are then fed into a simple feed-forward NN, FFNN, and a nonlinear function, GELU (Hendrycks and Gimpel, 2016).

Given a particular span representation and a candidate concept as the inputs, we formulate the context score as follows:

$$score_{cont}(s, c) = \mathbf{g}_s \cdot \mathbf{W}_c$$

where $\mathbf{W} \in \mathbb{R}^{|\mathcal{L}| \times d^{\mathbf{g}}}$ is the weight matrix associated with each concept c, and $\mathbf{W}_c$ represents the weight vector for the concept c.

Dictionary-matching score We used the cosine similarity of the TF-IDF vectors as the dictionary-matching features. Because there are several synonyms for a concept, we calculated the cosine similarity for all synonyms of the concept and used the maximum cosine similarity as the score for each concept. The TF-IDF is calculated using the character-level n-gram statistics computed for all diseases appearing in the training dataset and controlled vocabulary. For example, given the span "breast cancer," synonyms with high cosine similarity are "breast cancer (1.0)" and "male breast cancer (0.829)."

3 Experiment

3.1 Datasets

To evaluate our model, we chose two major datasets used in disease name recognition and normalization against a popular controlled vocabulary, MEDIC (Davis et al., 2012). Both datasets, the National Center for Biotechnology Information Disease (NCBID) corpus (Doğan et al., 2014) and the BioCreative V Chemical Disease Relation (BC5CDR) task corpus (Li et al., 2016), comprise of PubMed titles and abstracts annotated with disease names and their corresponding normalized term IDs (CUIs). NCBID provides 593 training, 100 development, and 100 test data splits, while BC5CDR evenly divides 1500 data into the three sets. We adopted the same version of MEDIC as TaggerOne (Leaman and Lu, 2016) used, and that we dismissed non-disease entity annotations contained in BC5CDR.

3.2 Baseline Models

We compared several baselines to evaluate our model. DNorm (Leaman et al., 2013) and NormCo (Wright et al., 2019) were used as pipeline models due to their high performance. In addition, we used the pipeline systems consisting of state-of-the-art models: BioBERT (Lee et al., 2020) for NER and BioSyn (Sung et al., 2020) for NEN.

TaggerOne (Leaman and Lu, 2016) and Transition-based model (Lou et al., 2017) are used as joint-learning models. These models outperformed the pipeline models in NCBID and BC5CDR. For the model introduced by Zhao et al. (2019), we cannot reproduce the performance reported by them. Instead, we report the performance of the simple token-level joint learning model based on the BioBERT, which referred as "joint (token)".

3.3 Implementation

We performed several preprocessing steps: splitting the text into sentences using the NLTK toolkit (Bird et al., 2009), removing punctuations, and resolving abbreviations using Ab3P (Sohn et al., 2008), a common abbreviation resolution module. We also merged disease names in each training set into a controlled vocabulary, following the methods of Lou et al. (2017).

For training, we set the learning rate to 5e-5, and mini-batch size to 32. λ was set to 0.9 using the development sets. For BC5CDR, we trained the model using both the training and development sets

Models	NCBID		BC5CDR	
	NER	NEN	NER	NEN
TaggerOne	0.829	0.807	0.826	0.837
Transition-based model	0.821	0.826	0.862	**0.876**
NormCo	0.829	0.840	0.826	0.830
pipeline	0.874	0.841	0.865	0.818
joint (token)	0.864	0.765	0.855	0.817
Ours without dictionary	0.884	0.781	0.864	0.808
Ours	**0.891**	**0.854**	**0.867**	0.851

Table 1: F1 scores of NER and NEN in NCBID and BC5CDR. Bold font represents the highest score.

following Leaman and Lu (2016). For computational efficiency, we only consider spans with up to 10 words.

3.4 Evaluation Metrics

We evaluated the recognition performance of our model using micro-F1 at the entity level. We consider the predicted spans as true positive when their spans are identical. Following the previous work (Wright et al., 2019; Leaman and Lu, 2016), the performance of NEN was evaluated using micro-F1 at the abstract level. If a predicted concept was found within the gold standard concepts in the abstract, regardless of its location, it was considered as a true positive.

4 Results & Discussions

Table 1 illustrates that our model mostly achieved the highest F1-scores in both NER and NEN, except for the NEN in BC5CDR, in which the transition-based model displays its strength as a baseline. The proposed model outperformed the pipeline model of the state-of-the-art models for both tasks, which demonstrates that the improvement is attributed not to the strength of BioBERT but the model architecture, including the end-to-end approach and combinations of dictionary-matching features.

Comparing the model variation results, adding dictionary-matching features improved the performance in NEN. The results clearly suggest that dictionary-matching features are effective for NN-based NEN models.

4.1 Contribution of Dictionary-Matching

To analyze the behavior of our model in the zero-shot situation, we investigated the NEN performance on two subsets of both corpora: disease names with concepts that appear in the training data (i.e., standard situation), and disease names with concepts that do not appear in the training data (i.e., the zero-shot situation). Table 2 shows the number of mentions and concepts in each situation. Table 3 displays the results of the zero-shot and standard situation. The proposed model with dictionary-matching features can classify disease concepts in the zero-shot situation, whereas the NN-based classification model cannot normalize the disease names.

dataset	standard		zero-shot	
	mention	concept	mention	concept
NCBID	781	135	179	56
BC5CDR	4031	461	391	179

Table 2: Number of mentions and concepts in standard and zero-shot situations.

	Methods	NCBID	BC5CDR
zero-shot	Ours without dictionary	0	0
	Ours	**0.704**	**0.597**
standard	Ours without dictionary	0.854	0.846
	Ours	**0.905**	**0.877**

Table 3: F1 scores for NEN of NCBID and BC5CDR subsets for zero-shot situation where disease concepts do not appear in training data and the standard situation where they do appear in training data.

The results of the standard situation demonstrate that combining dictionary-matching features also improves the performance even when target concepts appear in the training data. This finding implies that an NN-based model can benefit from dictionary-matching features, even if the models can learn from many training data.

4.2 Case study

We examined 100 randomly sampled sentences to determine the contributions of dictionary-matching features. There are 32 samples in which the models predicted concepts correctly by adding dictionary-matching features. Most of these samples are disease concepts that do not appear in the training set but appear in the dictionary. For example, "*pure red cell aplasis* (MeSH: D012010)" is not in the BC5CDR training set while the MEDIC contains "Pure Red-Cell Aplasias" for "D012010". In this case, a high dictionary-matching score clearly leads to a correct prediction in the zero-shot situation.

In contrast, there are 32 samples in which the dictionary-matching features cause errors. The

sources of this error type are typically general disease names in the MEDIC. For example, "Death (MeSH:D003643)" is incorrectly predicted as a disease concept in NER. Because these words are also used in the general context, our model overestimated their dictionary-matching scores.

Furthermore, in the remaining samples, our model predicted the code properly and the span incorrectly. For example, although "thoracic hematomyelia" is labeled as "MeSH: D020758" in the BC5CDR test set, our model recognized this as "hematomyelia." In this case, our model mostly relied on the dictionary-matching features and misclassifies the span because 'hematomyelia" is in the MEDIC but not in the training data.

4.3 Limitations

Our model is inferior to the transition-based model for BC5CDR. One possible reason is that the transition-based model utilizes normalized terms that co-occur within a sentence, whereas our model does not. Certain disease names that co-occur within a sentence are strongly useful for normalizing disease names. Although BERT implicitly considers the interaction between disease names via the attention mechanism, a more explicit method is preferable for normalizing diseases.

Another limitation is that our model treats the dictionary entries equally. Because certain terms in the dictionary may also be used for non-disease concepts, such as gene names, we must consider the relative importance of each concept.

5 Conclusion

We proposed a end-to-end model for disease name recognition and normalization that combines the NN-based model with the dictionary-matching features. Our model achieved highly competitive results for the NCBI disease corpus and BC5CDR corpus, demonstrating that incorporating dictionary-matching features into an NN-based model can improve its performance. Further experiments exhibited that dictionary-matching features enable our model to accurately predict the concepts in the zero-shot situation, and they are also beneficial in the other situation. While the results illustrate the effectiveness of our model, we found several areas for improvement, such as the general terms in the dictionary and the interaction between disease names within a sentence. A possible future direction to deal with general terms is to jointly train the parameters representing the importance of each synonyms.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: Analyzing text with the natural language toolkit*. O'Reilly Media, Inc.

Allan Peter Davis, Thomas C Wiegers, Michael C Rosenstein, and Carolyn J Mattingly. 2012. MEDIC: A practical disease vocabulary used at the comparative toxicogenomics database. *Database*, 2012:bar065.

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. NCBI disease corpus: A resource for disease name recognition and concept normalization. *J. Biomed. Inform.*, 47:1–10.

Arnaud Ferré, Louise Deléger, Robert Bossy, Pierre Zweigenbaum, and Claire Nédellec. 2020. C-Norm: a neural approach to few-shot entity normalization. *BMC Bioinformatics*, 21(Suppl 23):579.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*.

Robert Leaman, Rezarta Islamaj Dogan, and Zhiyong Lu. 2013. DNorm: Disease name normalization with pairwise learning to rank. *Bioinformatics*, 29(22):2909–2917.

Robert Leaman and Zhiyong Lu. 2016. TaggerOne: Joint named entity recognition and normalization with semi-markov models. *Bioinformatics*, 32(18):2839–2846.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *EMNLP*, pages 188–197.

Sunwon Lee, Donghyeon Kim, Kyubum Lee, Jaehoon Choi, Seongsoon Kim, Minji Jeon, Sangrak Lim, Donghee Choi, Sunkyu Kim, Aik-Choon Tan, and Jaewoo Kang. 2016. BEST: Next-Generation biomedical entity search tool for knowledge discovery from biomedical literature. *PLoS One*, 11(10):e0164680.

Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. 2016. BioCreative V CDR task corpus: A resource for chemical disease relation extraction. *Database*, 2016:baw068.

Yinxia Lou, Yue Zhang, Tao Qian, Fei Li, Shufeng Xiong, and Donghong Ji. 2017. A transition-based joint model for disease named entity recognition and normalization. *Bioinformatics*, 33(15):2363–2371.

Sunghwan Sohn, Donald C Comeau, Won Kim, and W John Wilbur. 2008. Abbreviation definition identification based on automatic precision estimates. *BMC Bioinformatics*, 9:402.

Mujeen Sung, Hwisang Jeon, Jinhyuk Lee, and Jaewoo Kang. 2020. Biomedical entity representations with synonym marginalization. In *ACL*, pages 3641–3650.

Shikhar Vashishth, Rishabh Joshi, Denis Newman-Griffis, Ritam Dutt, and Carolyn Rose. 2020. MedType: Improving Medical Entity Linking with Semantic Type Prediction. *arXiv preprint arXiv:2005.00460*.

Leon Weber, Jannes Münchmeyer, Tim Rocktäschel, Maryam Habibi, and Ulf Leser. 2020. HUNER: improving biomedical NER with pretraining. *Bioinformatics*, 36(1):295–302.

Dustin Wright, Yannis Katsis, Raghav Mehta, and Chun-Nan Hsu. 2019. NormCo: Deep disease normalization for biomedical knowledge base construction. In *AKBC*.

Dongfang Xu, Zeyu Zhang, and Steven Bethard. 2020. A Generate-and-Rank framework with semantic type regularization for biomedical concept normalization. In *ACL*, pages 8452–8464.

Jun Xu, Yonghui Wu, Yaoyun Zhang, Jingqi Wang, Hee-Jin Lee, and Hua Xu. 2016. CD-REST: A system for extracting chemical-induced disease relation in literature. *Database*, 2016:baw036.

Sendong Zhao, Ting Liu, Sicheng Zhao, and Fei Wang. 2019. A neural multi-task learning framework to jointly model medical named entity recognition and normalization. In *AAAI*, pages 817–824.