NTTD at the NTCIR-16 Real-MedNLP Task

Shuai Shao
NTT Data
Shuai.Shao@nttdata.com

Gongye Jin
NTT Data
Gongye.Jin@nttdata.com

Daisuke Satoh
NTT Data
Daisuke.Satoh@nttdata.com

Yuji Nomura
NTT Data
Yuji.Nomura@nttdata.com

ABSTRACT

The NTTD team participated in the Subtask1-CR-JA and Subtask1-RR-JA subtasks of the NTCIR-16 Real-MedNLP Task. This paper reports our approach to solve the NER (named entity recognition) problem when dealing with limited labeled medical documents. The documents are real Case-Report and Radiographic-Report data in Japanese. We first applied our recently developed annotation inconsistency detection tool to detect and correct inappropriate labels within the given training data. Then we applied data augmentation methods to create additional labeled data and combined the original and additional data as training data of our model. In this task, we fine-tuned Flair by the forementioned training data and acquired the results.

KEYWORDS

named entity recognition, data augmentation, synonym replacement, annotation inconsistency detection

TEAM NAME

NTTD

SUBTASKS

Subtask1-CR-JA, Subtask1-RR-JA

1 INTRODUCTION

In recent years, instead of the conventional paper format, the electronic format of medical records is becoming increasingly prevalent. Thus, extracting information from those data is becoming more important to accelerate the diagnosis and chart review process. Named entity recognition is one of the most essential information extraction tasks and there is no exception when dealing with the medical records. It is known that supervised learning is still the dominant method in the NER task, and to build a high-quality model needs large amount of labeled data. However, the labeled Japanese medical dataset is still rare and due to the required specialty, it is highly time consuming and costly if we try to label the data by ourselves.

The NTTD team participated in the Subtask1-CR-JA and Subtask1-RR-JA subtasks of the NTCIR-16 Real-MedNLP Task [1]. The dataset provided in the subtasks are relatively small which reflects the real-world situation. In Subtask1-CR-JA, the training and test datasets respectively consist of 100 case reports (CR) which can be openly accessed at CiNii. The training dataset and test dataset comprises 72 and 63 radiology reports (RR). The lack of sufficient data remains a huge challenge for building a high-quality NER model. This paper reports our approach to solve the problem and discusses the official results.

2 RELATED WORK

To overcome the challenge proposed by the small dataset, various approaches have been investigated. For example, semi supervised learning is applied when large amount of unlabeled data is available [2]. Transfer learning pretrains language representations on large amount of unlabeled data and adapts the representations to the target task [3]. Active learning extracts the least confident results from the current model and add more data based on that information [4]. With extremely limited data size, few-shot learning-like approaches are among the most common [5]. Since these subtasks mainly focus on dealing with text data from medical domain, a pre-trained model learned from the corresponding domain is optimal for achieving promising performance, which is unfortunately too costly to obtain during this competition period.

And currently the direct augmentation on labeled data is attracting more attention, since its simplicity and versatility. Data augmentation approaches are being widely used in image processing field, in which data can be easily created by changing the color, rotation or mixing multiple images intuitively. In natural language processing, data augmentation techniques are also attracting more attention recently [6].

3 METHODS

For higher efficiency we chose data augmentation approaches to deal with which automatically generate more training data by exploiting the existing ones. Before applying the data augmentation approach, we first utilized our recently developed annotation inconsistency detection tool to ensure the consistency within the given CR training dataset. In this section we explain the annotation inconsistency detection and all the data augmentation methods we have experimented with during the competition.

3.1 Annotation inconsistency detection

Our annotation inconsistency detection tool can promptly extract the expressions which have been annotated with different labels in the dataset. After the extraction, since the label sometimes can be different based on the context, instead of making one expression with the same label unconditionally, we checked the detection result and chose appropriate labels from the existing label lists to correct the inconsistency manually. During the competition period, we applied the tool to the CR training dataset, and made changes to 167 labels according to the detection results. Samples of label detected inconsistency can be found in the following Table 1. After the annotation changes, the data augmentation methods in the following sections are applied.

Table 1: Inconsistency detected by tool

Labels	Sentences
a	<a>**口腔粘膜**および口唇に広範囲に糜爛
a	<a>顔の一部と**口腔粘膜**にびらんを認めた
	豆腐と比べて**口腔粘膜**からの吸収性が良く

3.2 Simple data augmentation

We chose two approaches out of four approaches proposed by Dai and Adel [6] which are all NER-specific based on the experiment results conducted on other datasets.

Synonym replacement (SR): Different from the data augmentation approaches for text classification which assigns a label to each document, the basic idea of data augmentation for NER is to replace each token with other expressions instead of replacing the entire document in order to maintain name entity information. SR first uses a binomial distribution to randomly determine whether each token should be replaced. If a token is to be replaced, its synonyms are retrieved from WordNet. Simple rules are applied to assign IOB-labels (in which "B" is used to indicate the beginning of a named entity, "I" identifies the subsequent tokens in the named entity and "O" means that a token is outside of named entities) to multi-token synonyms to maintain label consistency in the newly created training samples.

Shuffle within segments (SiS): This approach first split the whole document into mention/non-mention segments. Therefore, under the IOB-labels scheme, consecutive tokens with a label starts with "B" or "I" are merged into the same segment and consecutive tokens with label "O" are also grouped. Each segment is then randomly decided whether to shuffle based on a binomial distribution. We maintain the label order of the segments chosen to be shuffled.

3.3 Masked language model (MLM)

Since SR replace selected target token by its synonym retrieved from a hand-crafted knowledge base which could be suffered from low coverage, we utilize masked language model proposed in BERT [7] to acquire synonyms for each target token. During synonym acquisition, the target token is masked using special symbol "[MASK]" and top *N* synonyms are predicted using the context.

Also, instead of equally assigning a binomial distribution to each token in a document, we consider the fact that different types of tokens contain different volume of information that can be useful to this task. For example, it is very natural to treat prepositions or pronouns as less informative while verbs or nouns convey more information, thus are usually more important. As a result, we measure the level of importance by simply calculating the tf-idf value for each token. Tokens which have tf-idf values larger than a threshold are chosen to be replaced. In order not to disrupt the internal structure for each mention, we avoid replacing tokens within the mention spans.

4 EXPERIMENTS

4.1 Settings

Because of the limited size of training data, we used 3-fold cross validation for better evaluation of the built models. Since the time for fine-tuning the model was very tight for us, we tested all the methods on CR training data which was randomly divided into 3 different training-test datasets by ratio of 8:2. We first converted the provided XML file to IOB format, in which mostly of modern NLP models handles NER task. And due to the limited experiment time, we omitted attribute level labels and focused on entity level performance. We experimented with augmented training data using each approach alone, along with training data augmented by all approaches. We empirically set the parameters of data augmentation approaches. Concerning SR approach for data augmentation, the parameter *p* of the binomial distribution used to decide whether each token is to replace was set to 0.2. For SiS, the parameter *p* was set to 0.5. And for MLM, the tf-fidf value was set at 0.1.

Below are the examples of one sentence augmented by different augmentation approaches.

Table 2: Sentences augmented by different approaches

Approaches	Sentences
Original	<timex3>今回</timex3>は<a>眼瞼周囲の<d>浮腫</d>, <d>紫斑と呼吸困難</d>のため<cc>緊急入院</cc>した。
SR	今回 は 目縁 周囲 の 水症 , 紫斑 と 呼吸 作用 波乱 の 悧 巧 事変 入院 し た。
SiS	今回 は 周囲 眼瞼 の 浮腫 , と 呼吸 困難 紫斑 ため の 入院 緊急 し。
MLM	今回 の 眼瞼 周囲 性 浮腫 , 紫斑 と 呼吸 困難 が ため 緊急 入院 でき 退院 。

It is obvious that SiS and MLM approaches tend to change the context around the entities, either by changing the word order or by replacing the original word with a synonym. In contrast, SR is the

only approach that can introduce new named entities, though sometimes the entity may not be appropriate.

We used *Flair* [8] to implement a neural network for NER model training. Detail specification during implementation is listed as follows. Other parameters were set as default.

Table 3: Parameters of NER model implementation using Flair

parameters	values
embeddings	"ja-forward", "ja-backword"
hidden_size	256
use_crf	True
learning_rate	0.1
mini_batch_size	32
max_epochs	150

After cross validation, same parameters were used applied for data augmentation of full training data and final model training.

4.2 Results

We first evaluate the performance of Flair without applying the data augmentation as baseline. The results were evaluated on 3 randomly created datasets divided from the provided CR training data. Then SR is applied for data augmentation, 2255(±43) sentences were added into the 3 training datasets, and we reevaluated the performance of Flair. Similarly, we applied SiS and MLM approaches, 753(±18) and 747(±14) sentences were created and merged into the training datasets.

The results are listed in Table 4. It can be observed that among all data augmentation methods, Flair achieved highest performance gain after applying SR, in terms of F1 score, which is 0.11.

Table 4: Performance of different models

	Precision	Recall	F1
Baseline	0.63(±0.02)	0.61(±0.03)	0.62(±0.03)
SR	**0.73(±0.02)**	**0.73(±0.01)**	**0.73(±0.01)**
SiS	0.72(±0.01)	0.72(±0.01)	0.72(±0.01)
MLM	0.71(±0.01)	0.71(±0.01)	0.72(±0.01)

In Table 5 it is shown that most of our models can hardly predict the label of "センチネルリンパ節" (Sentinel lymph node). Only when the dataset was augmented by SR approach, it was correctly predicted as Anatomical parts instead of Diseases and Symptoms. This may be due to that most of "リンパ節" (lymph node) was followed by "腫大" (enlargement) or "転移" (metastasis).

Table 5: Prediction of different models

Models	Results
Baseline	<d>センチネルリンパ節</d>に<d>転移</d>はなく
SR	<a>センチネルリンパ節に<d>転移</d>はなく
SiS	<d>センチネルリンパ節</d>に<d>転移</d>はなく
MLM	<d>センチネルリンパ節</d>に<d>転移</d>はなく

Therefore, we applied SR to all given CR and RR training datasets and built 2 models respectively to predict labels of given CR and RR test datasets.

The official results of CR and RR test datasets are listed in the Table 6.

Table 6: Model performance after applying SR on given test datasets

	Precision	Recall	F1
CR-test	0.62	0.62	0.62
RR-test	0.87	0.87	0.87

The results above show that with the increase of test data, the precision, recall and F1 score of given CR test datasets all decreased about 10% compared to the results on 20% CR training dataset.

Table 7 shows the detailed results of different tags on CR dataset by using SR approach. The final result on the given test data were mostly consistent with the results in our 3-fold experiments, where the model achieved highest performance on timex3 and fell unconfident when predict m-val entities. It can be inferred that the second largest entity number and relatively less variation patterns of timex entity type contribute to the result above. On the contrary, m-val entity type is rarest in the training data, which may explain the high standard deviation in our 3-fold experiment and the low accuracy in the final result. Besides, there are some inconsistences on the results of t-key and t-test entity type, which suggests that further experiments by splitting the training data according to the ratio of provided training and test may give closer result to the final model.

Table 7: Model performance on each tag types in CR dataset

Tags	F1(3-fold experiments)	F1(final result)
a	0.72(±0.03)	0.54
d	0.73(±0.02)	0.65
m-key	0.76(±0.02)	0.60
m-val	**0.72(±0.17)**	**0.48**
t-key	0.76(±0.06)	0.44
t-test	0.76(±0.02)	0.45
t-val	0.77(±0.03)	0.59
timex3	**0.86(±0.00)**	**0.77**

5 CONCLUSIONS

Our work attempt to solve the low-resource NER task by leveraging the data augmentation method. And by simply using the synonym replacement method, we achieved 10% performance increase which efficiently reduce the cost of annotation. There still remains challenges that if we can further combine different data augmentation methods or even combine data augmentation with other approaches such as transfer learning to achieve higher

performance increase. We plan to further investigate the combination of different methods and evaluate the approach on multiple datasets to verify the versatility.

REFERENCES

[1] Shuntaro Yada, Yuta Nakamura, Shoko Wakamiya, Eiji Aramaki. 2022. Real-MedNLP: Overview of REAL document-based MEDical Natural Language Processing Task, In *Proc. of the 16th NTCIR Conference on Evaluation of Information Access Technologies*.

[2] Wenhui Liao, and Veeramachaneni Sriharsha. 2009. A simple semi-supervised algorithm for named entity recognition. In *Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing*. 2009.

[3] Suchin Gururangan, Ana Marasovic, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *ACL*, pages 8342–8360, Online.

[4] Yanyao Shen, Hyokun Yun, Zachary Lipton, Yakov Kronrod, and Animashree Anandkumar. 2017. Deep Active Learning for Named Entity Recognition. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 252–256, Vancouver, Canada. Association for Computational Linguistics.

[5] Huang, Jiaxin, et al. 2020. Few-shot named entity recognition: A comprehensive study. arXiv preprint arXiv:2012.14978.

[6] Xiang Dai and Heike Adel. 2020. An Analysis of Simple Data Augmentation for Named Entity Recognition. In *Proceedings of the 28th International Conference on Computational Linguistics.*

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *North American Association for Computational Linguistics (NAACL)*.

[8] Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. Flair: An easy-to-use framework for state-of-the-art nlp. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59. aclweb.org.