FRDC at the NTCIR-16 Real-MedNLP Task

Zhongguang Zheng
FRDC
China
zhengzhg@fujitsu.com

Lu Fang
FRDC
China
fanglu@fujitsu.com

Yiling Cao
FRDC
China
caoyiling@fujitsu.com

Jun Sun
FRDC
China
sunjun@fujitsu.com

ABSTRACT

In this paper, we describe the approaches of FRDC team for the Real-MedNLP task. Specially, the FRDC team participated in three sub-tasks including Subtask1-CR-EN, Subtask3-CR-EN (ADE), and Subtask3-RR-EN (CI). The Real-MedNLP task aims to promote approaches for supporting real medical services under constrained training resources. We applied pre-trained language models (PTLMs) such as BERT and BioBERT to learn sentence and document representations. For each sub-task, we designed different networks based on PTLMs. Various effective methods such data augmentation were adopted in each sub-task. In the official run, we achieved the best score for the CI sub-task, and ranked 2nd in the ADE sub-task.

KEYWORDS

Pre-trained language model, NER, classification, document clustering

TEAM NAME

FRDC

SUBTASKS

Subtask1-CR-EN
Subtask3-CR-EN (ADE)
Subtask3-RR-EN (CI)

1 INTRODUCTION

Recently, the traditional paper medical documents have been gradually replaced by the electronic medical records. This digital transformation makes it necessary to apply practical natural language processing (NLP) technologies to handle the medical records. The Real-MedNLP Task in NTCIR-16 provides a platform for developing practical NLP techniques that support various medical services.

The FRDC team participated in three English sub-tasks of the NTCIR-16 Real-MedNLP Task, including Subtask1-CR-EN, Subtask3-CR-EN (ADE), and Subtask3-RR-EN (CI). This paper reports our approaches to solve the problems and discusses the official results.

The remainder of this paper is organized as follows: In Section 2, we introduce our approaches for all the sub-tasks. Section 3 describes the official experiment results and analysis, followed by the conclusion in Section 4.

2 METHODS

2.1 Subtask1-CR-EN

We regard the problem as named entity recognition (NER) with data augmentation strategies. We employ BioBERT model [5] to learn the semantic representation of each sentence and classify each token. There are two runs for this problem denoted by FRDC-1 and FRDC-2. In the FRDC-1 run, we build a baseline system based on a pre-trained BioBERT model. In the FRDC-2 run, we build an improved system based on a pre-trained BioBERT model with token-level data augmentation strategies. Inspired by [3], we employ total 4 strategies.

Label-wise Token Replacement (LwTR): For each token, we randomly decide whether it should be replaced by a binomial distribution. If yes, we then select the token that has the same label with the original token. We should notice that the order of the label remains unchanged.

Synonym Seplacement (SR). For each token, we randomly decide whether it should be replaced by a binomial distribution. If yes, we then select one of the synonyms of the original tokens from WordNet. We should notice that the order of the label remains unchanged.

Mention replacement (MR). For each mention, we randomly replace it with another mention that has the same type with the original mention. We should notice that the order of the label could be changed accordingly.

Shuffle Within Segments (SiS). First we divide the sentence into segments that have the same labels with each other in the segment. Then we randomly select the segments to be shuffled. But the order of the label remains unchanged.

2.2 Subtask3-CR-EN (ADE)

The ADE subtask is especially designed for MdeTxt-CR, the participants are asked to predict a "ADEval" value of an entity in a given report. The "ADEval" value denotes the level of certainty that the medicine caused some ADEs or the disease was caused by some drugs, which is split into four level: 0, 1, 2 and 3. Our method is based on fine-tuning a vocabulary adapted BERT model (VART) with multi-learning mechanism, where we transform the ADE task as a classification task.

The Main Task: For each entity name n and its context c, the input sequence is formed in *[CLS] c [SEP] n* as the input of the fine-turning procedure, where *[CLS]* is the beginning of each sequence, and *[SEP]* is a special token used to separate n and c. The

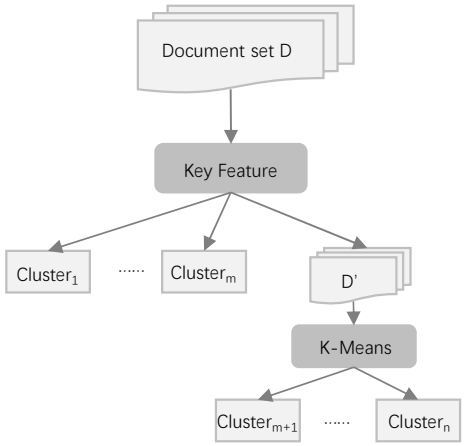

Figure 1: Overview of the two-step document clustering method for CI sub-task.

output of the final hidden state of the first word *[CLS]* in the input sequence is used to compute the probability distribution of the four classes (ADEval=0,1,2,3), and the output probability for each class is calculated using the softmax function.

The Auxiliary Task: The auxiliary task is learning to classify the type of an entity. The ADE information consists of two types: <d>-table for disease and symptom names, and <m-key>-table for medication (drug) names. We also use the final layer of the special token *[CLS]* to compute the probability distribution of binary classes, which is calculated by the softmax function.

Then the parameters are learned jointly by minimizing the overall loss function, which is the sum of the loss of the above two tasks.

2.3 Subtask3-RR-EN (CI)

The CI sub-task is designed for radiology report documents (MedTxt-RR) [7], which are the descriptions of radiology images and written by radiologists. Given a set of MedTxt-RR documents, The participants are required to group the reports diagnosing the same image. We consider the challenge as a document clustering problem and propose a two-step document clustering method which is depicted in Figure 1.

2.3.1 Key Feature Clustering. According to the introductions of the CI sub-task, each image should have 9 documents. This information motivates us to find useful features for clustering the documents. Given the training set, we first calculate the document frequency (DF) for all the words, and then we select all the words with DF equals 9 and find that each selected numeric word appears in only one cluster. From the original document, we can see that those selected numeric words are describing the size information, such as "The size of the lesion is **28** mm" and "There is a large **43** mm tumor occupying the right hilum".

Given an image, different people may have different descriptions. However, the size information is consistent. As a result, the size information can be used as key features denoted as $feat_i$=[v_i, c_j, df_i], where v_i is the numeric word representing the size information,

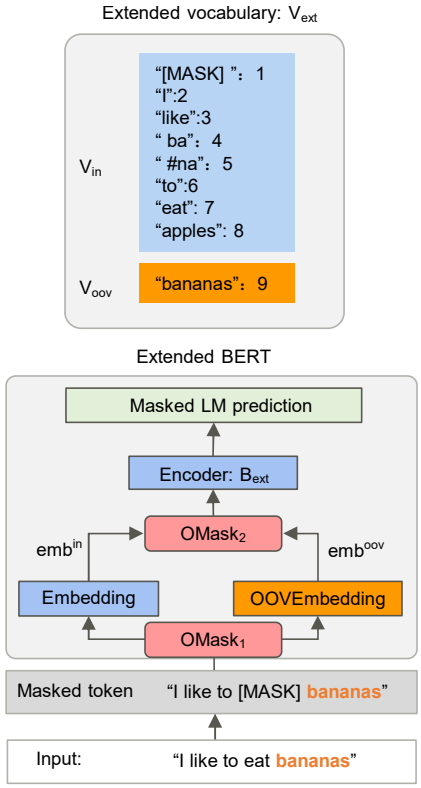

Figure 2: Overview of vocabulary extended BERT model.

c_j denotes the cluster ID, df_i is the DF of v_i and $7 \leq df_i \leq 9$. Given a document d, if d contains word v_i in $feat_i$, then the document will be clustered to c_j.

2.3.2 Document Embedding and K-means Clustering. Given the document set D, after key feature clustering, we use k-means algorithm to cluster the reset of the documents denoted by D' into $n - m$ clusters, where n is the total cluster number, and m is the cluster number generated by key features (See Figure 1).

The k-means API provided in NLTK [1] [2] toolkit is adopted in this work. We use document embeddings as the input for the k-means algorithm. Pre-trained language model (PTLM) BERT [4] and BioBERT [5] are used to encode the input document. Instead of using the hidden state of "[CLS]" token as the document representation, we use sentence-BERT [2] [6] to obtain the document embeddings.

In order to further improve the performance of PTLMs, we propose a vocabulary adapted BERT model (VART) to adapt the original BERT model to the target dataset. Given an original pre-trained BERT model B with a vocabulary V_{in} and a training dataset D_t for a specific task, e.g., CI challenge, we firstly expand V_{in} to V_{ext} with an out-of-vocabulary (OOV) list extracted from D_t, and then design an extended BERT model B_{ext}, which is inherited from B and is further pre-trained with V_{ext} on D_t. Finally, the adapted B_{ext} model will be fine-tuned for downstream tasks.

[1] https://github.com/nltk/nltk
[2] https://github.com/UKPLab/sentence-transformers

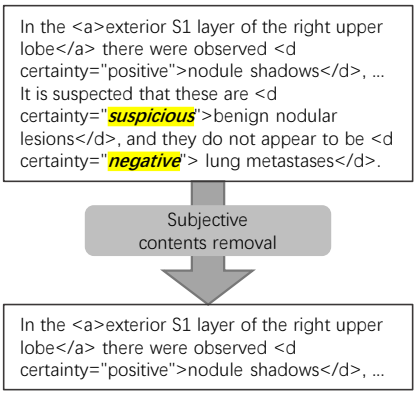

Figure 3: Remove subjective contents for CI task.

Table 1: The main hyperparameters for Subtask1-CR-EN task.

Name	Value of FRDC-1 and FRDC-2
Weight Initialization	biobert-base
Batch Size	16
Optimizer	Adam
Learning Rate	5e-5
Epoch	65

Figure 2 shows an example of VART. Suppose D_t contains one sentence "I like to eat bananas", we first extract the OOV list containing one word "bananas", and then simply complement an extra embedding layer denoted by "OOVEmbedding" to the original BERT model B while preserving the original embedding layer "Embedding" to encode the in-vocabulary words. As the example shown in Figure 2, by applying $OMask_1$, only the new word "bananas" is encoded by the OOVEmbedding layer. Afterwards, it is handy to use $OMask_2$ to combine the embedded vectors emb^{in} and emb^{oov}. This process is described in the following Equations, where w_i is id of the i-th word in the input sequence.

$$\text{emb}_i^{\text{in}} = \text{Embedding}((1 - \text{OMask}_{1,i}) \cdot w_i) \quad (1)$$

$$\text{emb}_i^{\text{oov}} = \text{OOVEmbedding}(\text{OMask}_{1,i} \cdot w_i) \quad (2)$$

$$\text{emb}_i = (1 - \text{OMask}_{2,i})\text{emb}_i^{\text{in}} + \text{OMask}_{2,i}\text{emb}_i^{\text{oov}} \quad (3)$$

The encoder of B_{ext} is initialized with the encoder in B. For the task layer in this paper, we only select the masked language model (MLM) task to further pre-train B_{ext}. Considering that the vocabulary size of V_{ext} is enlarged, the magnitude of prediction vector from the task layer should be increased to match the size of V_{ext}. Therefore, we create a new task layer with the proper dimension for further pre-training B_{ext} on D_t. Finally, the adapted B_{ext} model is used to encode the document.

2.3.3 Data Augmentation. Besides the main methods introduced above, we also propose data augmentation methods to alleviate the problem of limited training data.

Table 2: The results of our models in the evaluation set in Subtask1-CR-EN task.

Model name	P	R	F1
FRDC-1	64.19	65.39	64.78
FRDC-2	64.63	64.45	64.54

Table 3: The number of each entity in the Subtask1-CR-EN task.

Metrics	Sort of entities	FRDC-1	FRDC-2
Character-Accuracy	All target entities	78.48	79.41
Entity-Precision		42.27	47.01
Entity-Recall		35.43	35.70
Entity-F1		38.55	40.58
Entity-F1	a	51.56	49.72
	d	55.28	58.74
	m-key	64.90	65.33
	m-val	66.67	71.19
	t-key	37.12	38.58
	t-test	48.26	47.06
	t-val	45.37	45.10

For further pre-training the VART model, we use all the MedTxt-CR and MedTxt-RR datasets. Besides the original documents, we augment the datasets with documents with the stop words removed.

When encoding a document d, we firstly remove the stop words and then augment d with d', where d' is derived from d by removing the sentences containing only subjective contents. As the example shown in Figure 3, we remove the sentences containing "suspicious" and "negative" tags but no "positive" tags. The motivation is that the descriptions of one image should be relatively consistent (objective contents), such as body part, symptom descriptions. The objective contents are helpful for the clustering task. However, different people may yield different diagnoses (subjective contents). The inconsistent information, though it is important, will hinder the clustering result. Moreover, since there are not any certainty tags in the test set, we opt to train a small binary classifier with the training set to find those subjective contents in the test set.

2.3.4 Result Selection. Considering that the results generated by k-means algorithm are not consistent due to the randomly initialized centroids. We propose a result selection method based on a voting strategy to select the optimum result as much as possible.

We first run k-means algorithm multiple times, and then construct a $d \times d$ matrix A based on the results, where d is the total document number. Element $a_{i,j}$ denotes the number of times the i-th and j-th document are grouped in the same cluster. For each clustering result, we calculate a score using $score = \sum_1^h \sum_1^d a_{i,j}$, where h is the total cluster number and $a_{i,j} = 0$ if the i-th and j-th documents are not co-occurred in cluster h_i. Finally, we sort and select top N clustering results as the final outputs.

Table 4: Official results of Subtask3-CR-EN (ADE). The result of our team is marked in bold. "†" denotes the best score.

MaskedGroupID		C8	I1	F2	H1
ADEval=0	P	96.42	97.02	95.39	96.57
	R	97.79	97.63	98.10	97.95
	F	97.10	97.32	96.73	97.25
ADEval=1	P	20.00	30.00	0.00	14.29
	R	5.26	31.58	0.00	5.26
	F	8.33	30.77	0.00	7.69
ADEval=3	P	47.62	100.00	40.00	60.00
	R	52.63	26.32	42.11	63.16
	F	50.00	41.67	41.03	61.54
Report-level	P	50.00	50.00	40.00	50.00
	R	77.78	88.89	44.44	66.67
	F	**60.87**	64.00†	42.11	57.14

3 EXPERIMENTS

3.1 Subtask1-CR-EN

We used the first 100 articles of the training data, which contain 1,602 sentences, as the training set and the rest 50 articles of the training data, which contain 712 sentences, as the test set. Besides, in the FRDC-2 run, we add additional 6,408 sentences generated by data augmentation strategies introduced in Section 2.1. For each strategy, we generated 1,602 sentences. Table 1 shows the main hyperparameters in the experiments.

We use entity precision (P), entity recall (R) and entity F1 as our metrics to select the best model in the test set. Finally, we selected two models described in Table 2. We call them FRDC-1 and FRDC-2. The recall and F1 of FRDC-1 are better than those of FRDC-2, but the precision of FRDC-1 is lower than that of FRDC-2. But all the metrics of two model are quite close to each other.

The official results of our two models are listed in Table 3. The organizer use entity character accuracy, entity precision, entity recall and entity F1 as metrics. From Table 3, we can see that FRDC-2 is better than FRDC-1 in 9 metrics. We believe that this improvement is because of the data augmentation strategies.

3.2 Subtask3-CR-EN (ADE)

In the Subtask3-CR-EN (ADE) task, we need to predict a value of "ADEval", given "articleID", "tag", and "text".

We first extract the context of the entities in training dataset from corresponding report. If an entity occurs multiple times in a report, we combine the sentences which contain the entity as its context. Concerning the imbalance of the training dataset, we then augment the data by using AEDA [1] method and synonym replacement method. In synonym replacement method, we randomly choose n words from the context and replace them with their synonyms chosen randomly.

Two levels of evaluation in the entity level and the report level are applied. Table 4 shows the results of the official run on test data, where the masked group ID "C" denotes our group. Compared to the best results of other groups, our best results achieved the second best F1-score on entity level evaluation of ADEval=1,3 and also the second best F1-score on report level evaluation.

Table 5: Results on the training set of Subtask3-RR-EN (CI). The Best scores of each setting are marked in bold.

	txt	txt +rm_stop	txt +rm_stop +rm_sbj	txt +rm_stop +rm_sbj
	K-means			K-means +Feat
BERT	0.3634	0.3903	0.4427	0.7727
BioBERT	0.3396	0.4644	0.5421	0.8168
$TAPT_{BERT}$	0.3775	0.4252	0.5089	0.7995
$TAPT_{BioBERT}$	0.3924	0.4630	0.5322	0.8094
$VAPT_{BERT}$	**0.4943**	**0.5689**	**0.5909**	**0.8394**
$VART_{BioBERT}$	0.3443	0.4584	0.4952	0.8335

Table 6: Official results of Subtask3-RR-EN (CI). The result of our team is marked in bold. "†" denotes the best score.

Masked Group ID	C1	F1	I1
Result	**0.8724**†	0.2172	0.7879

3.3 Subtask3-RR-EN (CI)

Table 5 lists the results on the training set. The first row denotes the input format, where "txt" is the original document, "rm_stop" denotes the augmented input with stop words removed, and "rm_sbj" means the removal of the subjective contents. The second row represents the clustering algorithms, where "K-means+Feat" is the proposed two-step clustering method. The first column lists all the PTLMs used in our experiment. There are three settings of using PTLMs.

- **Direct Document Encoding.** We use BERT [3] and BioBERT [4] to encode the document directly.
- **Task Adaptation (TAPT).** We first further pre-train the BERT and BioBERT models on the training set, and then encode the document with the adapted PTLMs.
- **VART.** We further pre-train the BERT and BioBERT models based on the method introduced in Section 2.3.2, and then encode the document for the following clustering algorithms.

Normalized Mutual Info Score (NMI) is adopted as the evaluation metric, and each number in Table 5 is the average score of 50 runs. From the results we can see that our proposed methods are constantly improve the clustering performance. The best scores from "$VART_{BERT}$" and "$VART_{BioBERT}$" are not significantly different. However, considering that "$VART_{BERT}$" produced best scores in all the settings, we select "$VART_{BERT}$" as the final pre-trained model in the official run. Table 6 shows the results of the official run, where the masked group ID "C" denotes our results. Only the best score of each team is listed. We submitted 10 results in total and reached the best score.

[3] https://huggingface.co/bert-base-cased
[4] https://huggingface.co/dmis-lab/biobert-base-cased-v1.1

4 CONCLUSIONS

In this paper, we introduced our approaches for the Real-MedNLP Task. Specially, we presented our systems for Subtask1-CR-EN, Subtask3-CR-EN (ADE) and Subtask3-RR-EN (CI) sub-tasks, respectively. We achieved the best score in the CI task and the second best score in the ADE task in the official run. The results demonstrated the effectiveness of our approaches.

REFERENCES

[1] Leonardo Rossi Akbar Karimi and Andrea Prati. 2021. AEDA: An Easier Data Augmentation Technique for Text Classification. In *EMNLP 2021*.

[2] Bird, Steven, Edward Loper, and Ewan Klein. 2009. Natural Language Processing with Python.

[3] Xiang Dai and Heike Adel. 2020. An Analysis of Simple Data Augmentation for Named Entity Recognition. In *https://arxiv.org/pdf/2010.11683.pdf*.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *https://arxiv.org/pdf/1810.04805.pdf*.

[5] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. In *Bioinformatics*.

[6] Reimers, Nils, Gurevych, and Iryna. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 3982–3992.

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