Mathematical Biosciences and Engineering

MBE, 20(1): 1018–1036.
DOI: 10.3934/mbe.2023047
Received: 31 May 2022
Revised: 07 September 2022
Accepted: 18 September 2022
Published: 24 October 2022

http://www.aimspress.com/journal/MBE

Research article

Stacking-BERT model for Chinese medical procedure entity normalization

Luqi Li[1], Yunkai Zhai[2], Jinghong Gao[2], Linlin Wang[2], Li Hou[1,*] and Jie Zhao[2,*]

[1] Institute of Medical Information, Chinese Academy of Medical Sciences/Peking Union Medical College, Beijing, China

[2] National Engineering Laboratory for Internet Medical Systems and Applications, The First Affiliated Hospital of Zhengzhou University, Zhengzhou, China

* **Correspondence:** Email: hou.li@imicams.ac.cn, zhaojie@zzu.edu.cn.

Abstract: Medical procedure entity normalization is an important task to realize medical information sharing at the semantic level; it faces main challenges such as variety and similarity in real-world practice. Although deep learning-based methods have been successfully applied to biomedical entity normalization, they often depend on traditional context-independent word embeddings, and there is minimal research on medical entity recognition in Chinese Regarding the entity normalization task as a sentence pair classification task, we applied a three-step framework to normalize Chinese medical procedure terms, and it consists of dataset construction, candidate concept generation and candidate concept ranking. For dataset construction, external knowledge base and easy data augmentation skills were used to increase the diversity of training samples. For candidate concept generation, we implemented the BM25 retrieval method based on integrating synonym knowledge of SNOMED CT and train data. For candidate concept ranking, we designed a stacking-BERT model, including the original BERT-based and Siamese-BERT ranking models, to capture the semantic information and choose the optimal mapping pairs by the stacking mechanism. In the training process, we also added the tricks of adversarial training to improve the learning ability of the model on small-scale training data. Based on the clinical entity normalization task dataset of the 5th China Health Information Processing Conference, our stacking-BERT model achieved an accuracy of 93.1%, which outperformed the single BERT models and other traditional deep learning models. In conclusion, this paper presents an effective method for Chinese medical procedure entity normalization and validation of different BERT-based models. In addition, we found that the tricks of adversarial training and data augmentation can effectively improve the effect of the deep learning model for small samples, which might provide some useful ideas for future research.

Keywords: Chinese medical procedure entity normalization; BERT; Siamese-BERT; stacking; adversarial training

1. Introduction

Mining medical text data from electronic health records (EHRs) to generate clinical evidence has been widely applied in clinical decision-making. One fundamental problem in medical text mining is entity normalization, which aims to map entity mentions to standard concepts in a given knowledge base (KB) or controlled vocabulary. Accurate entity normalization can solve the problem of consistency in the expression of entity mentions and realize information sharing at the semantic level. In China, with increasing implementation of a healthcare payment policy by diagnostics-related groups in hospitals, a large amount of irregular writing in clinical notes need to be manually mapped to the standard concepts of the International Classification of Diseases (ICD); additionally, the entity normalization task of diagnoses and procedure has become very important, as it requires sufficiently trained staff with a good knowledge of both medicine and coding rules. In the real world, medical entity normalization tasks are time-consuming and labor-intensive; thus, this paper mainly focuses on the Chinese medical procedure entity normalization task and describes an automated and efficient method to map clinical terms into ICD codes in Chinese.

There are three major challenges to optimizing the Chinese medical procedure entity normalization task: 1) **Variety**. Due to diverse writing habits, the experience of physicians and the requirements of medical institutions, there are many different non-standard expressions in Chinese; the same concept may be linked by different entity mentions; for example, entity mentions that "Mile's", "直肠癌根治术 (Dixon)" are all linked to the normalized concept "腹会阴直肠切除术 (abdominoperineal resection of the rectum)" in Chinese control vocabulary ICD-9-CM-3. 2) **Similarity**. Chinese words have similar glyphs but different semantics, such as the two-procedure concepts "硬脊膜外病损切除术 (excision of epidural lesion)" and "硬脊膜下病损切除术 (excision of subdural lesion)" in Chinese control vocabulary ICD-9-CM-3; their similarity interferes with the exact matching of terms. 3) **Limited context information**. Mention-level entity is short text whose critical context information is limited, and the concept in ICD has no semantic relationship information available. To solve these problems, we regarded the mention-level entity normalization as a sentence-pair classification task in this study and designed a stacking-bidirectional encoder representations from transformers (BERT) fusion model to capture the semantic information of clinical entity mentions. External KB and easy data augmentation (EDA) skills were used to increase the diversity of training samples, which provided rich term variation features to model. In addition, we generated difficult negative samples to train the model to learn the subtle differences between concepts and added adversarial learning in the training process to improve the discrimination ability of the model to deal with similar samples.

The normalization task here could be referred to as entity linking in the computer science community. In the biomedical domain, many previous studies focus on the development of rule-based methods [1–3]. Their work relied on large, expert-curated vocabularies of standardized medical terminology for string matching-based approaches, with great success [4]. In recent years, deep learning-based systems have addressed the limitations of string matching and achieved good performance of entity normalization. In general, deep learning-based systems could consist of two steps [5]: (i) Candidate Concept Generation – to retrieve candidate concepts related to a given entity mention; (ii) Candidate Concept Ranking – to rank the candidate concepts and decide on the one most relevant to the given entity mention. To improve the efficiency of candidate concept generation, Vashishth et al. [6] introduced a semantic-type prediction module to alleviate the problem of the overgeneration of candidate concepts by filtering out irrelevant candidate concepts based on the predicted semantic type of a mention.

Candidate concept ranking is the key step for entity linking systems. Similarity-based methods have been proven to be effective for concept ranking. They commonly used sentence embedding as an upstream task before text classification, which adopts a vector space model to represent entity mentions and candidate concepts into a fixed length vector for semantic similarity calculations [7–9]. In recent years, deep representation learning models such as BERT [10] have been widely used to improve many natural language processing (NLP) tasks. In the medical domain, BioBERT [11] and ClinicalBERT [12] language representation models, which were pre-trained on biomedical texts and clinical notes based on BERT architecture, were introduced to advance the state-of-the-art performance on many domain-specific NLP tasks. Li et al. [13] introduced a BERT-based model named EhrBERT that was trained using 1.5 million EHRs; they proved the effectiveness of their BERT-based model on entity normalization tasks, but they treated entity normalization as a multi-classification task of a single sentence, where the size of classes depends on the vocabularies used in a corpus; the performance of this model relies on having a large amount of training data for each class, so it is not suitable for small samples. Kalyan and Sangeetha [14] proposed a medical concept normalization system based on BERT and highway layers; our experimental results show that our model outperformed all existing methods on two standard datasets. Sung et al. [15] introduced a BIOSYN system for biomedical entity representation learning that uses synonym marginalization dispensing with the explicit needs of negative training pairs; our results show that the iterative candidate selection based on our model's representations is crucial for improving the performance, together with synonym marginalization. The above studies' preliminaries proved the effectiveness of BERT on clinical entity classification tasks. In this study, we developed different sentence-pair similarity calculation models with different structures based on BERT, and stacking was performed to make full use of the advantages of BERT-based models.

Most previous studies have focused on the standardization of English entities. Up to now, there have been few studies specifically designed for Chinese-based clinical entity normalization. The real-world public datasets in Chinese related to health informatics are almost nonexistent, and this has been a bottleneck for the development of text mining in the Chinese medical entity normalization domain. Some researchers have developed algorithms based on manually annotated datasets. Xia et al. [16] proposed a multi-field indexing approach, which accomplishes the term normalization task by using an information retrieval algorithm with four level indices: word, character, pinyin and its initial. Luo et al. [8] introduced a multiview convolutional neural network to address the normalization of diagnostic and procedure names simultaneously. Likewise, Zhang et al. [17] presented an unsupervised framework to normalize the Chinese medical concept by combining disease text with comorbidity. Wang et al. [18] developed and compared several entity-linking approaches to normalize disease and procedure terms in Chinese; their results showed that the BERT-based ranking method achieved the best performance on encoding both Chinese diagnosis and procedure terms.

Based on the previous studies, the entity normalization was regarded as a sentence-pair classification task in this study; we designed different sentence-pair similarity calculation models with different structures based on BERT and propose a stacking-BERT fusion model to capture the semantic information of clinical entity mentions. There are three major contributions of this paper:

- We used an external KB and EDA skills to increase the diversity of training samples; the results show that EDA skills can provide more features of term variation for the model.
- We proposed a concept ranking model with different structures based on BERT; it is fused by a stacking mechanism to further improve the performance of the model. Our detailed experimental analysis on Chinese medical procedure entity normalization tasks realized remarkable improvements over existing methods.
- We added adversarial learning to the training process; the results show that adversarial learning can significantly enhance the robustness and generalization of the model.

2. Materials and methods

2.1. Study design

Given the medical procedure entity set $E = \{e_1, e_2, \dots, e_i, \dots e_m\}, m \in N$, which recognizes Chinese clinical text, and a controlled vocabulary $C = \{c_1, c_2, \dots, c_i, \dots c_n\}, n \in N$, which consists of a set of standard concepts, the entity normalization task of our study is to find the best corresponding concept c for each input entity e_i, as shown in Eq (1), where the score is calculated by the text matching algorithm in our model:

$$c \quad = argmax_{c \in C}\left(score(e_i, c_i)\right) \tag{1}$$

Figure 1 shows the system architecture for entity normalization used in this study, which consists of three modules: 1) dataset construction: to increase the diversity of training samples by using an external KB and EDA skills; 2) candidate concept generation: to generate a list of candidate ICD concepts for a given entity, using a simple BM25 algorithm and an extended BM25 by integrating synonym knowledge of SNOMED CT and train data; and 3) candidate concept ranking: to rank candidate ICD concepts, we propose a stacking-BERT model with different structures based on BERT, which was fused by a stacking mechanism. Detailed descriptions of these methods are given in the following sections.

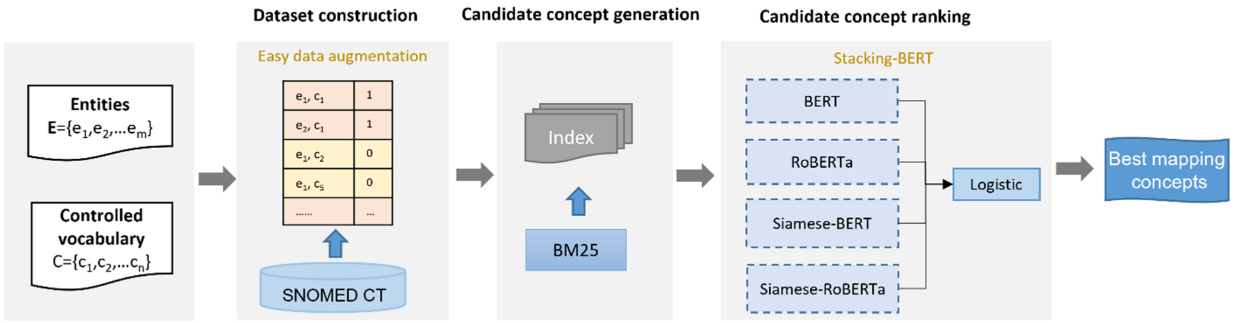


Figure 1. Overall experimental framework for this study.

2.2. Dataset construction

2.2.1. Dataset

We evaluated our approach on the clinical entity normalization task dataset of the 5th China Health Information Processing Conference (CHIP2019) [19]. The dataset provides procedure entities recognized from Chinese electronic medical records, and the controlled vocabulary is "ICD-9-CM-3 Peking union medical college hospital edition 2017", which contains 9467 different procedure concepts in Chinese, where each entity in the dataset is manually linked to one or more than one standardized concept in the controlled vocabulary. The distribution of entities in the dataset is shown in Table 1 and the examples are shown in Figure 2. The dataset has the following problems: 1) the dataset does not give negative samples with entities that do not match with concepts; 2) due to the small training set, there were 23% concepts in the test set that were not in the training set; and 3) one entity may link to more than one concept, and approximately 5% of entities in the dataset map to multiple concepts.

Table 1. Statistics of the dataset.

	Training set	Test set
Entities	5000	2000
One entity links to one concept	4751 (95.02%)	1901 (95.05%)
One entity links to more than one concept	199 (4.98%)	99 (4.95%)
Concepts of vocabulary	9467	

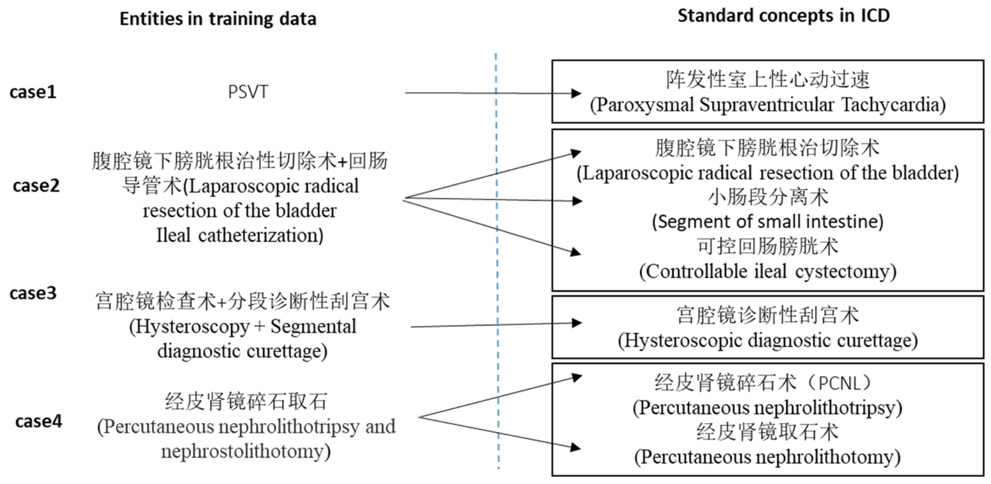

Entities in training data
Standard concepts in ICD
case1
PSVT
阵发性室上性心动过速
(Paroxysmal Supraventricular Tachycardia)
case2
腹腔镜下膀胱根治性切除术+回肠导管术(Laparoscopic radical resection of the bladder Ileal catheterization)
腹腔镜下膀胱根治切除术
(Laparoscopic radical resection of the bladder)
小肠段分离术
(Segment of small intestine)
可控回肠膀胱术
(Controllable ileal cystectomy)
case3
宫腔镜检查术+分段诊断性刮宫术
(Hysteroscopy + Segmental diagnostic curettage)
宫腔镜诊断性刮宫术
(Hysteroscopic diagnostic curettage)
case4
经皮肾镜碎石取石
(Percutaneous nephrolithotripsy and nephrostolithotomy)
经皮肾镜碎石术（PCNL）
(Percutaneous nephrolithotripsy)
经皮肾镜取石术
(Percutaneous nephrolithotomy)

Figure 2. Example of the dataset.

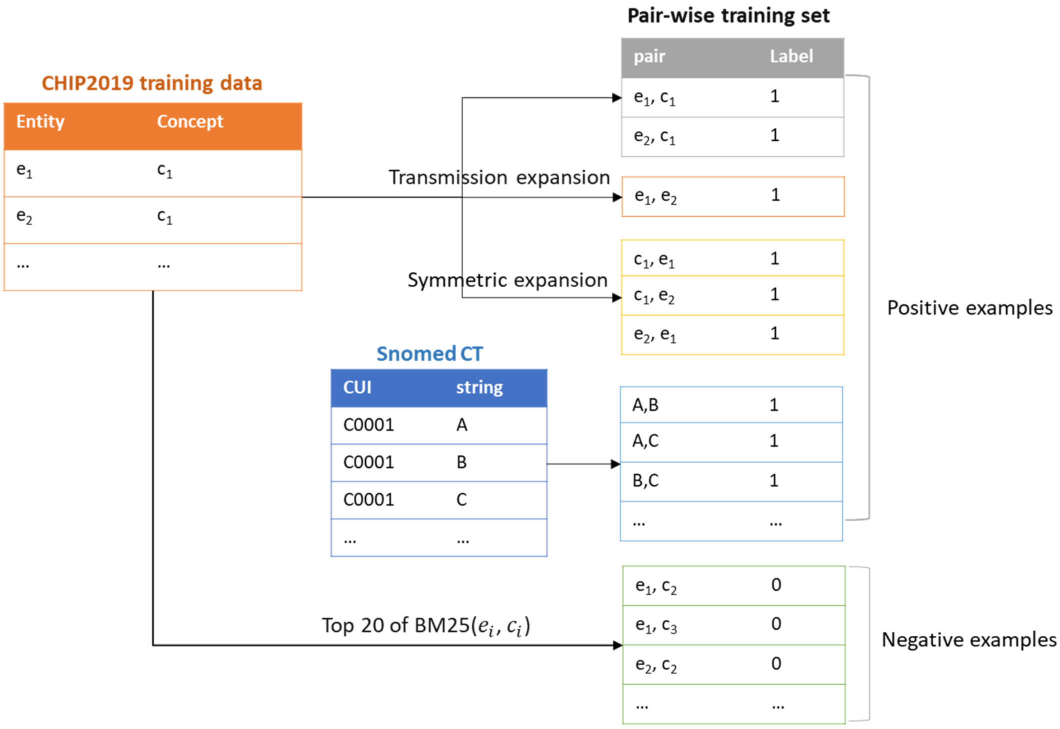
2.2.2. Easy data augmentation of training data

In order to make the model learn more semantic information, the construction of the training set is very important. We adopted EDA skills to generated new pairwise training data based on the CHIP2019 dataset (Figure 3).

Data cleaning. We cleaned the useless punctuation and content in procedure entities to match regular expressions, such as "(腹腔镜)胆囊切除术 (51.2201)" to "腹腔镜胆囊切除术 (laparoscopic cholecystectomy)". Then, English abbreviations that appear in the training set entities were extracted separately.

Positive sample extension. Three methods were used to extend positive samples in our study: (i) data transmission expansion based on the pairs of training data; (ii) data symmetric extension based on the pairs of Step (i); and (iii) positive sample supplementation based on external clinical terminology. The Systematized Nomenclature of Medicine-Clinical Terms (SNOMED CT) is a comprehensive multilingual clinical terminology guide used in EHRs and interoperability, and its components are concepts (codes), descriptions (terms) and relationships. Each concept has a unique concept ID, a fully specified name and multiple descriptions (including a preferred term and one or more synonyms); they all expressed the same semantics of one concept. We matched all descriptions in the same concept pairwise in SNOMED CT and added all synonym pairs to the training set as positive examples.

Negative sample generation. Previous studies suggested that the construction of difficult negative samples can enhance the feature-learning ability of the model and thus improve its effectiveness. We generated negative samples for each entity with the commonly used information retrieval method BM25 introduced in Section 2.3. With the exception of the manually linked concept, other top 20 concepts were retrieved for each entity in training set.



CHIP2019 training data

Entity	Concept
e_1	c_1
e_2	c_1
...	...

Snomed CT

CUI	string
C0001	A
C0001	B
C0001	C
...	...

Pair-wise training set

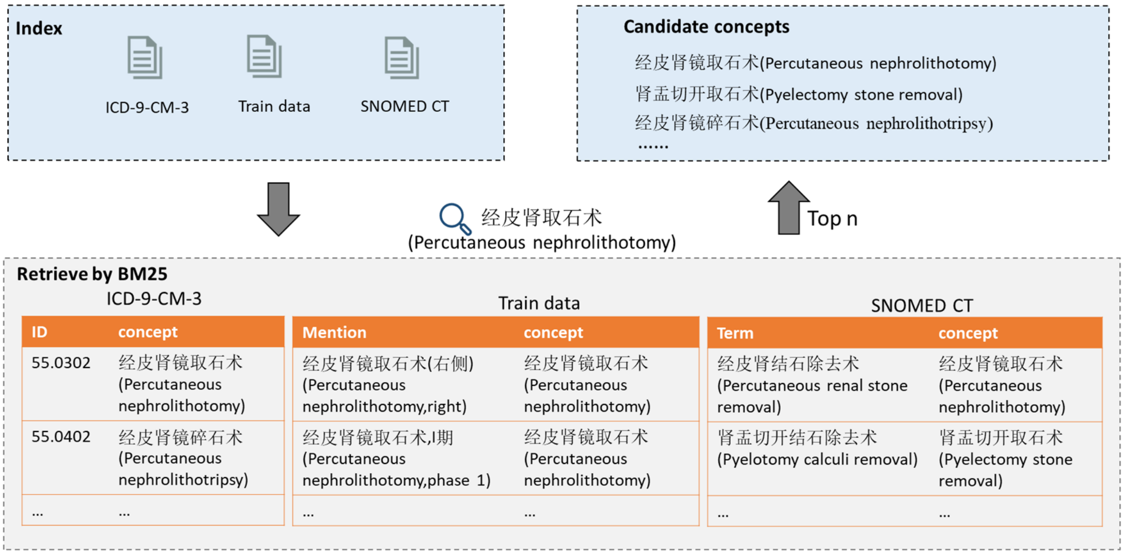
pair	Label
e_1, c_1	1
e_2, c_1	1
e_1, e_2	1
c_1, e_1	1
c_1, e_2	1
e_2, e_1	1
A,B	1
A,C	1
B,C	1
...	...
e_1, c_2	0
e_1, c_3	0
e_2, c_2	0
...	...

Figure 3. Methods of data augmentation.

2.3. Candidate concept generation

Due to the large size of the ICD-9-CM-3 data, if the whole vocabulary was used as a candidate concept set, most of the concepts are irrelevant to the entity, it will bring a great burden to the model operation. The purpose of candidate concept generation is to ensure that all possible correct concepts are added to the candidate concept set as much as possible. Common recall methods include string similarity calculations based on text features and search engine retrieval. In order to improve the efficiency of model operation and ensure the recall rate of the best corresponding concept, the candidate concept generation component consists of two steps: (1) indexing all ICD codes and their preferred concepts in Chinese by invoking the Lucene application programming interface, and (2) retrieving the top n candidate concepts $C = \{c_i\}_{i=1}^{n}$ from the index for a clinical entity e, by employing the BM25 model provided by Lucene [20].

To achieve higher recall for candidate generation, we used the Chinese characters as the basic building blocks of both indexing and retrieval without considering Chinese word segmentation. In addition to the baseline index described above, another two indexes were proposed in this section by using annotated training data and synonym terms of SNOMED CT. We established the index of SNOMED CT terms and ICD concepts by aligning the fully specified name and preferred terms in SNOMED CT with the concepts in ICD-9-CM-3 by regularization. Figure 4 shows an example of the complete candidate concept generation process.


Index
ICD-9-CM-3
Train data
SNOMED CT
Candidate concepts
经皮肾镜取石术(Percutaneous nephrolithotomy)
肾盂切开取石术(Pyelectomy stone removal)
经皮肾镜碎石术(Percutaneous nephrolithotripsy)
……
经皮肾取石术
(Percutaneous nephrolithotomy)
Top n
Retrieve by BM25

ICD-9-CM-3

ID	concept
55.0302	经皮肾镜取石术 (Percutaneous nephrolithotomy)
55.0402	经皮肾镜碎石术 (Percutaneous nephrolithotripsy)
...	...

Train data

Mention	concept
经皮肾镜取石术(右侧) (Percutaneous nephrolithotomy,right)	经皮肾镜取石术 (Percutaneous nephrolithotomy)
经皮肾镜取石术,I期 (Percutaneous nephrolithotomy,phase 1)	经皮肾镜取石术 (Percutaneous nephrolithotomy)
...	...

SNOMED CT

Term	concept
经皮肾结石除去术 (Percutaneous renal stone removal)	经皮肾镜取石术 (Percutaneous nephrolithotomy)
肾盂切开结石除去术 (Pyelotomy calculi removal)	肾盂切开取石术 (Pyelectomy stone removal)
...	...

Figure 4. Example of the candidate concept generation process.

2.4. Candidate concept ranking

This section mainly introduces the candidate concept ranking model, stacking-BERT, developed via our study. The stacking-BERT model consists of two layers, where the first layer includes four base ranking models with the different structures introduced in Sections 2.4.1 and 2.4.2, and the final layer is a simple logistic regression model. The Stacking mechanism and algorithm are introduced in Section 2.4.3.

2.4.1. BERT-based ranking model

As a sentence-pair classification task, using the BERT-based model shown in Figure 5(a), we treated the word representation from the top layer of transformers as the features for the normalization task. Similar to Ji et al. [21], in our BERT-based classification model, for each input entity m and a candidate concept c, we constructed a sequence < [CLS] e [SEP] c > as the input of the fine-tuning procedure, where [CLS] is the special word used as the representation of the whole sequence, and [SEP] is the special word used for separating e and c. After encoding 12 or 24 layers of multi-head attention transformers, the final hidden state output of the special [CLS] token $C \in R^H$ was passed to the softmax layer to compute the probability distribution of all classes, which is described as $softmax(CW^T)$, where $W \in R^{H \times K}$ is the parameter added during the fine-tuning procedure. Here, $K = 2$ means only two classifier labels in our task, the classifier $label = 1$ means that c is the mapping concept for e and $label = 0$ means that c is not the mapping concept. We employed the probability of $label = 1$ as the final score of each input pair; after ranking all scores, the top-ranking candidate concept c was found as the best mapping concept for e.

$$Score(e, c) = P(label = 1|e, c) = softmax(CW^T) \tag{2}$$

2.4.2. Siamese-BERT ranking model

The Siamese neural network architecture [22] of two towers with shared weights and a distance function at the last layer has been effective in learning similarities in domains such as text [23] and images [24] by modeling the similarity directly based on pairs of inputs. Siamese networks lend themselves well to the semantic invariance phenomena present in entity normalization. Recently, Fakhraei et al. [25] have developed a solution based on a deep Siamese neural network model (Siamese Bi-LSTM) to embed the semantic information about the entities and empirically show the effectiveness of these embeddings in bio-entity normalization datasets. Using BERT, researchers have started to input individual sentences into BERT and derive fixed-size sentence embeddings. The most commonly used approach is to average the BERT output layer (known as BERT embeddings) or use the output of the first token [CLS] [26–28]; but, Reimers and Gurevych's [29] work show that these common practices yield rather bad sentence embedding. They proposed a modification of the pretrained BERT network that uses Siamese and triple network structures to derive semantically meaningful sentence embeddings that can be compared using cosine similarity.

As shown in Figure 5(b), a Siamese-BERT network was built in this study based on the work of Reimers and Gurevych [29] to generate sentence embeddings independently for the entity mention and candidate concepts; then, they were concatenated as the input of the classification function. In the training process, candidate mapping pairs and a class label expressed as $< e, c, y >$ were fed to the Siamese-BERT network, which was composed of mapping pairs ($y = 1$) and other non-mapping pairs ($y = 0$). The aim of training is to minimize the distance in an embedding space between positive examples and maximize the distance between negative examples. We fine-tuned BERT to update the weights and produced sentence embeddings v_e and v_c, and as in Nils' work, a pooling operation was added to the output of BERT to derive the fixed-sized sentence embedding. For each candidate mapping pair (e, c), we concatenated the sentence embeddings v_e and v_c with the element-wise difference $v_e - v_c \vee$ and multiplied it with the trainable weight $W_t \in R^{3n \times K}$, where n is the dimension of the sentence embedding and K is the number of classifier labels:

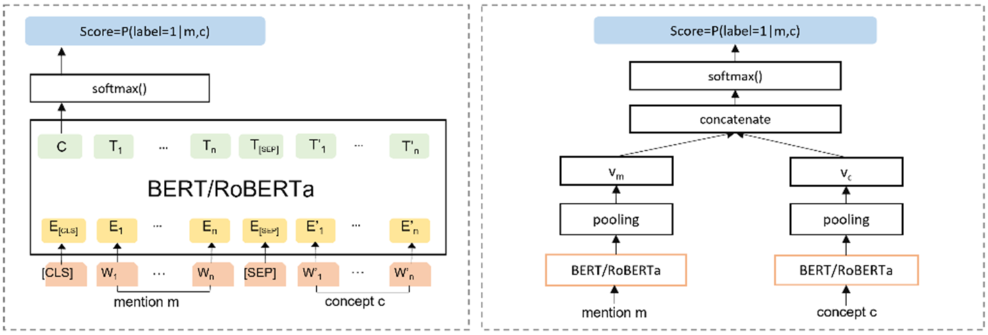
$$v = W_t \tag{3}$$

where v is a vector of the $K * 1$ dimension. Then, we computed the probability of each classifier label using the softmax function. Finally, the same with the BERT model, we computed the probability of $label = 1$ and found the top-ranking candidate concept c; the loss function of the network was set as categorical softmax loss:

$$Score(e, c) = P(i = 1|e, c) \tag{4}$$

$$L = -\sum_{j=1}^{K} y_j log\, Sj \tag{5}$$

where Sj is the prediction of the probability that this sample belongs to the jth classifier label, and y_j is the target probability the network should produce. This function makes the loss less when the prediction probability is close to the target probability, and larger when it is far away from the target probability.



(a) BERT-based model

(b) Siamese-BERT model

Figure 5. Structure of base ranking models.

2.4.3. Stacking-BERT model

Stacking is an effective ensemble learning method for classification problems, it generally use several basic classifier models to produce outputs, which are later used as features for the next stacking layer [30]. This paper presents a stacking-BERT model including two layers. Stacking models usually use several complex models for the base classifiers and a simpler combined model for the final model. Because we adopted a different language model, feature representation, network structure, corpus and adjustment strategy, the pretrained models learned different prior knowledge and performed differently in downstream tasks. In order to combine the characteristics of different pretrained models, we trained the two models introduced in the last section with the pretrained model BERT$_{\text{base-chinese}}$ [31] and RoBERTa$_{\text{large-pair}}$ [32] to generate four ranking models, i.e., the BERT-based model, RoBERTa-based model, Siamese-BERT model and Siamese-RoBERTa model. They produced the probability of $label = 1$ for each input sentence pair; then, these probability values were used as input in the logistic regression model that was a final layer. The algorithm of the stacking-BERT model is shown in Table 2. In particular, we used 5-fold cross-validation in the training process of each base ranking model.

Table 2. Algorithm of stacking-BERT.

```
Algorithm : stacking-BERT
    Input: Train dataset $\boldsymbol{D} = \{(\boldsymbol{x_1}, \boldsymbol{y_1}), (\boldsymbol{x_2}, \boldsymbol{y_2}), \ldots, (\boldsymbol{x_n}, \boldsymbol{y_n})\}$
        Base ranking model $\boldsymbol{\zeta_1}, \boldsymbol{\zeta_2}, \ldots, \boldsymbol{\zeta_T}$ ;
        combined model $\boldsymbol{\zeta}$ ;
    For $\boldsymbol{t} = \boldsymbol{1}, \boldsymbol{2}, \ldots, \boldsymbol{T}$ do
        $\boldsymbol{h_t} = \boldsymbol{\zeta_t}(\boldsymbol{D})$ ;
    End for
    $\boldsymbol{D'} = \emptyset$
    For $\boldsymbol{i} = \boldsymbol{1}, \boldsymbol{2}, \ldots, \boldsymbol{n}$ do
        For $\boldsymbol{t} = \boldsymbol{1}, \boldsymbol{2}, \ldots, \boldsymbol{T}$ do
            $\boldsymbol{z_{it}} = \boldsymbol{h_t}(\boldsymbol{x_i})$ ;
        End for
        $\boldsymbol{D'} = \boldsymbol{D'} \cup \big((\boldsymbol{z_{i1}}, \boldsymbol{z_{i2}}, \ldots, \boldsymbol{z_{iT}}), \boldsymbol{y_i}\big)$ ;
    End for
    $\boldsymbol{h'} = \boldsymbol{\zeta}(\boldsymbol{D'})$
    Output: $\boldsymbol{H}(\boldsymbol{x}) = \boldsymbol{h'}\big(\boldsymbol{h_1}(\boldsymbol{x}), \boldsymbol{h_2}(\boldsymbol{x}), \ldots, \boldsymbol{h_T}(\boldsymbol{x})\big)$
```

3. Results

3.1. Training details

3.1.1. Experimental settings

In this study, we built the experimental environment using a PyTorch 1.6 framework, using the library of Transfomers to load the pretrained models. The training set described in Section 2.2 was used to fine-tune the stacking-BERT model, wherein most model hyperparameters were the same as those saved in the pretrained model; we tuned the batch_size with 32 and fixed the max_sequence length with 128. In order to get the best result, we set learning rates of 1e-5, 2e-5 and 5e-5, respectively, for each model in the training process and tuned the number of training epochs from 1–10; finally, we saved the best performance for each model. The final hyperparameters of the four base ranking models are shown in Table 3. For the logistic regression model, we used the default parameters of sklearn [33].

Table 3. Hyperparameters of base ranking models.

Models	Learning rate	Epoch	Batch_size	Max_sequence
BERT$_{\text{base-chinese}}$	2e-5	4	32	128
RoBERTa$_{\text{large-pair}}$	1e-5	3	32	128
Siamese-BERT$_{\text{base-chinese}}$	2e-5	4	32	128
Siamese-RoBERTa$_{\text{large-pair}}$	1e-5	5	32	128

3.1.2. Adversarial training

Adversarial training is the process of training a model to correctly classify both unmodified examples and adversarial examples. Adversarial training has been widely applied and achieved good generalization performance on image classification tasks. Miyato et al. [34] extended these techniques to text classification tasks and sequence models by applying perturbations to the word embeddings in a recurrent neural network; the proposed method achieved state-of-the-art results on multiple benchmark semi-supervised and purely supervised text classification tasks. Furthermore, Madry et al. [35] proposed the projected gradient descent (PGD) method to improve the perturbations to the word embedding, their MNIST and CIFAR10 networks based on the PGD achieved good performance in response to a broad set of attacks.

To improve robustness and the generalization ability of concept ranking models, we added the adversarial training to the process of model training. Instead of interfering with the original input sample itself, adversarial training feeds the adversarial samples to the model by adding some small perturbations to the word vector of the embedded layer. Generally, the optimization function of adversarial training can be represented as follows [35]:

$$\underset{\theta}{Min} E_{(x,y)D}\left[\underset{r_{adv}\in S}{max} L(\theta, x + r_{adv}, y)\right] \tag{6}$$

The part of max () means that we need to find a set of adversarial samples that maximize loss in the sample space; the part of min () means that, when faced with the adversarial sample set of such a data distribution, we should minimize the expected loss of the model on the adversarial sample set by updating model parameter, where r_{adv} means the perturbations on input x.

PGD obtains adversarial examples by multi-step variant fast gradient sign attack (FGSM). With the initialization word embedding $x_0 = x$, the perturbed data in the *t*-th step x_t can be expressed as follows:

$$x_t = \prod\nolimits_{x+s}\left(x_{t-1} + \alpha \frac{g(x_{t-1})}{\vee g(x_{t-1})\vee_2}\right) \tag{7}$$

$$g(x_{t-1}) = \nabla_x L(x_{t-1}, y, \theta) \tag{8}$$

where $s = \{r \in R^d \colon \|r\|_2 \leq \varepsilon\}$ denotes the projection of perturbations into the set s, α is the step size, L is the loss function, the meaning of ∇_x is to take the partial derivatives. The algorithm of PGD in the training process is described as shown in Table 4.

Table 4. Adversarial training process for PGD.

```
Algorithm 1: Adversarial training process for PGD
Input: Initialization word embedding x of input data, perturbation accumulation steps $\boldsymbol{K}$
1. Compute the forward loss of $\boldsymbol{x}$, then compute the gard of backward $\boldsymbol{g}(\boldsymbol{x})$, backing up the initial embedding;
2.  for t in range($\boldsymbol{K}$): ($\boldsymbol{t}$ starts at 1)
3.     Compute adversarial perturbation $\boldsymbol{r_{adv}}$ by the grad of the embedding, add $\boldsymbol{r_{adv}}$ to the current embedding, which is represented as $\boldsymbol{x_t}$;
4.     if $\boldsymbol{t}! = \boldsymbol{K}$:
5.         Zero the grad, then compute the forward loss and of $\boldsymbol{x_t}$ in Step 3, then compute the $\boldsymbol{grad_{adv}}$ of backward;
6.     else:
7.         Restore the $\boldsymbol{g}(\boldsymbol{x_0})$ of Step1, then compute the last forward loss of $\boldsymbol{x_t}$ in Step 3, then compute the $\boldsymbol{grad_{adv}}$ of backward $\boldsymbol{g}(\boldsymbol{x_t})$ and add it to $\boldsymbol{g}(\boldsymbol{x_0})$;
8. Restore the embedding to the value of Step 1;
9. Update the parameters according to the grad of Step 7.
```

3.2. Evaluation metrics

We evaluated the performance of different entity normalization algorithms in terms of the evaluation metrics provided by the CHIP2019 organizer [19]. For each original entity $e_i, i \in (1, k)$ which has been manually annotated to N concepts in the test dataset, assuming the model outputs M concepts for e_i, N and M are a set of concepts and the score S of the model is calculated as

$$S_i = \frac{Count(N \cap M)}{Max(Count(N), Count(M))} \tag{9}$$

$$S = \frac{1}{k}\sum_1^k S_i \tag{10}$$

3.3. Evaluation results

3.3.1. Comparisons with other different models

Several unsupervised and deep learning models were selected as baseline methods in this paper:

• Metric_LCS [36] method. Longest common subsequence (LCS) finds the subsequences of two given sequences, which appear in the same order in the two sequences but need not be continuous; it is often used as the unsupervised method for text matching and to measure the literal similarity of strings. We used the Metric_LCS method to measure the literal similarity of entities and concepts, and then found the most similar concept as the standardized result.

• BM25 [20]. This is the most popular algorithm to calculate the query and document similarity score in the field of information indexing; we used the same method introduced in Section 2.2.3 and chose the top 1 candidate concept as the final result of this method.

• Bert-as-service [37]. The bert-as-service system uses BERT as a sentence encoder and hosts it as a service via ZeroMQ, mapping a variable-length sentence to a fixed-length vector using the BERT model. We used the bert-as-service system to calculate the sentence vectors of all entities and concepts, and then used cosine similarity to find the best matching concept for each entity.

• CNN-ranking model [7]. It was the best deep learning-based system to date on both the ShARe/CLEF and NCBI datasets. Since the language of data were different, we could not completely reconstruct the KBs as used but not released in Li et al.'s work; we just reimplemented the system in our data and used the same settings as described in their paper.

• Siamese Bi-LSTM model [24]. This model significantly has outperformed other models on web document retrieval tasks. Because the tasks and datasets are different, we just reimplemented the system in our data and used the same settings as described in their paper.

• BIOSYN model [15]. The BIOSYN model outperformed previous state-of-the-art models on four biomedical entity normalization datasets having three different entity types (disease, chemical, adverse reaction). We used the same method of sparse representation and the same settings described in their paper. However, the BioBERT model was replaced with the $\text{BERT}_{\text{base-chinese}}$ model because the BioBERT model was pretrained by an English corpus.

Table 5 shows the performance comparisons for different models. Compared with other methods, our stacking-BERT fusion clinical entity normalization system achieved the highest accuracy of 93.1% on the CHIP2019 test set. Respectively, all deep learning methods achieved better results than the unsupervised methods. The BIOSYN model performed better than other deep learning models. For three unsupervised models, the bert-as-service system performed better, as the accuracy was improved by at least 10% as compared to Metric_LCS. It can be seen that pretrained models based on large-scale corpora can play an important role in both supervised and unsupervised methods.

Table 5. Comparisons of different models for Chinese clinical entity normalization.

Models	Accuracy
Metric_LCS	51.24%
Bm 25	62.57%
bert-as-service$_{\text{base-chinese}}$	71.33%
CNN-ranking model [9]	86.7%
Siamese Bi-LSTM [20]	85.12%
BIOSYN	91.31%
Stacking-BERT	**93.1%**

**Note*: The bold values denote the highest values

3.3.2. Comparisons of ensemble models

In order to verify the effectiveness of the stacking model proposed in this work, we compared it with different ensemble models. Two ensemble models named Voting-BERT$_{\text{hard}}$ and Voting-BERT$_{\text{soft}}$ were obtained by fusing four BERT-based classifiers with a hard voting mechanism and soft voting mechanism, respectively [38]. From the performance shown in Table 6, we can find that 1) the ensemble model based on the stacking method performed better than voting methods; 2) compared with the single BERT-based ranking model, multi-model fusion can achieve a better result; 3) each BERT-based ranking model achieved a good result, i.e., the accuracy of each model was above 90%, and the result showed that the supervised learning model which fine-tuned with domain data was significantly better than that of unsupervised learning; 4) compared to the Siamese-BERT model with a structure of twin towers, the result of the BERT-based model was better; and 5) in the models with different structures, the pretrained model BERT$_{\text{base-chinese}}$ had achieved a better result than RoBERTa$_{\text{large-pair}}$, but the difference between the results was smaller.

Table 6. Comparison of ensemble models.

Models	Accuracy without PGD	Accuracy with PGD
Stacking-BERT	**91.73%**	**93.1%**
Voting-BERT$_{\text{soft}}$	91.66%	92.98%
Voting-BERT$_{\text{hard}}$	91.3%	92.11%
BERT-based	91.31%	92.05%
RoBERTa-based	91.15%	91.79%
Siamese-BERT	90.7%	91.51%
Siamese-RoBERTa	90.26%	91.33%

**Note*: The bold denotes the highest value; PGD: adversarial training method used in this paper

3.3.3. Effect of adversarial training

Table 6 shows that, regardless of whether it was our stacking-BERT model or other ranking models, adversarial training based on the PGD algorithm could effectively improve the effect of the model. When PGD adversarial training was added to the training process, the accuracy of the BERT-based model was even higher than that of the stacking model without PGD (92.05% vs. 91.73%).

3.3.4. Effect of EDA

Table 7 shows the results of our stacking-BERT model using different training data. D_0 refers to the 8000 positive samples in the training data, D_1, D_2 and D_3 respectively refer to the positive examples generated by the three methods introduced in Section 2.2.2. The results show that the negative examples generated by BM25 were much better than that randomly selected; in the case of identical positive samples, the accuracy improved by 22.8%. In addition, we validated the effects of different data augmentation methods through ablation experiments. By comparing the results of four experiments, it can be seen that three data augmentation methods all played a certain role in improving the effect of the model. Particularly, the positive samples supplemented by SNOMED CT were most effective, as the accuracy stabilized at more than 92% when we used the supplementary positive samples.

Table 7. Comparisons of different training data types used in our stacking-BERT model.

Training data Positive examples	Negative examples	Accuracy of stacking-BERT model
$D_0 + D_1 + D_2 + D_3$	Top 20 candidate concepts (randomly)	70.12%
D_0	Top 20 candidate concepts (bm25)	91.33%
$D_0 + D_1 + D_2$	Top 20 candidate concepts (bm25)	91.56%
$D_0 + D_1 + D_3$	Top 20 candidate concepts (bm25)	92.8%
$D_0 + D_2 + D_3$	Top 20 candidate concepts (bm25)	92.44%
$\mathbf{D_0 + D_1 + D_2 + D_3}$	**Top 20 candidate concepts (bm25)**	**93.1%**

**Note*: The bold denotes the highest value

3.3.5. Effect of candidate concept generation

As described in Section 2.3, we adopted a BM25 algorithm to generate candidate concepts. Figure 6 reports the number of candidates per entity and the rate of standard entity recall for the candidate sets that were conducted using two types of strategies. The line "(total)" means the recall of candidate concepts in all test sets, while the line "(1 to 1)" only calculates the recall rate of samples which one entity linked to one concept. For the traditional information retrieval model BM25, to which we applied three indexes, the top 20 candidates were retrieved for each entity and a recall of 99.6% was obtained for one-to-one samples. When the number of candidate concepts was the same, the recall rate of the BM25 algorithm was higher than the bert-as-service system, which proves that our method is more efficient.

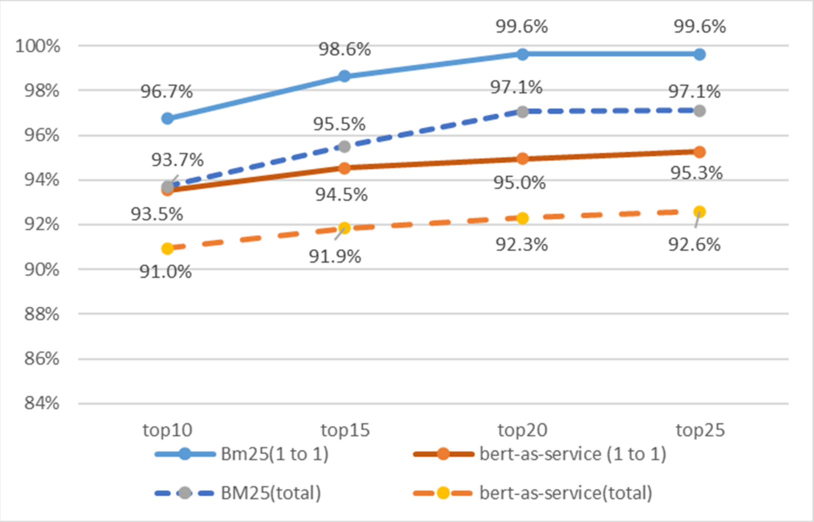


Figure 6. Comparisons of different methods of candidate concept generation.

4. Discussion

As shown in the results, the performance of the stacking-BERT model was better than that of other deep learning models. Stacking models can make full use of the learning ability of base classifiers and further improve the classification effect without increasing the complexity of a single model, or the amount of training data. The combination of classification models with different structures and pretrained models can produce better results. BERT can learn deeper semantic features through the mechanism of multi-head attention based on the transformer. At the same time, it used the task of next sentence prediction as the training goal and trained the language representation together with the mask language model. This design was used to capture the relationship between sentences, which was conducive to the application of pretrained general representation in text matching and other tasks. Second, the BERT pretraining models were all based on large-scale Chinese text corpus like wiki; they fully learned the grammatical features of Chinese words and phrases. Therefore, BERT-based models proved to be effective for Chinese clinical entity normalization tasks.

However, the differences between four ranking models were not quite as large, and the BERT-based models performed better than the Siamese-BERT models. Note that the Siamese-BERT framework was not optimal for sentence pairwise classification. It used a bi-encoder that mapped sentences independently to sentence embeddings. For classification, the classifier would take these two embeddings and derive a label. On the other side, BERT used a cross-encoder, which meant that both sentences were present at input time, and BERT compared the two inputs to derive the labels which gave much better classification results.

The quality of the training dataset had a close relationship with the results of the model. Particularly, the generation of negative examples was very important. Negative examples generated by BM25 were hard samples for the model, and more detailed differences could be learned through hard similar samples to improve the discrimination ability of the model. For the three data augmentation methods for the positive samples, the external clinical terminology supplement in the same domain was the most effective method. Using a transitive extension can make the model learn more similar information; using a symmetric extension to exchange the position of text pairs will change the position encoding so that the model can observe the similarity of the two texts from different angles.

Candidate concept generation needs to consider both the recall rate and data scale. The BM25 retrieval method based on a triple index proposed in this paper has been proved to be simple and effective. But, there is also a drawback, because our dataset had cases that one entity linked to multiple standard concepts in the CHIP2019 dataset; the candidate concept generation recall rate for the total test data did not reach 100%; thus, the concept ranking model could not find the correct concept. Deep generative models will be considered in the future to improve the recall rate of candidate concept generation.

There were a lot of concepts with high similarity and redundant components in the original words of procedure in the data, and these will cause interference in the model in the process of training and prediction. The adoption of PGD confrontation training can improve the robustness of the model response to confrontation samples. However, a PGD algorithm will increase our training time, and it was not suitable for large-scale datasets.

In our experiment, an entity in the test set may be linked to one or more concepts; the statistics show that our multi-model fusion system had a normalization accuracy of 96.48% for single mapping and 25.86% for multiple mapping. For the clinical entity standardization task of CHIP2019, the average score of all participating teams was 79.75%; the first ranked team constructed a ranking system of implication scores based on BERT and applied the best fine-tuning to the quantity prediction module, finally achieving an effective result of 94.83%; the final performance of our model was second only to the Top 1 team [19]. The analysis of the experimental results shows that our model needs to be improved in two aspects. On the one hand, our model had poor ability to predict the number of concepts; using a manual rule or deep learning model to predict the number of concepts will be the way to improve our methods in the future. On the other hand, although we have dealt with common abbreviations in the data preprocessing stage, the normalization performance of new entities with professional abbreviations was still not ideal; for example, the entity mention “VVI 心脏起搏器植入术(Cardiac pacemaker implantation)” was predicted as the normalized concept “心脏起搏器置入术(Cardiac pacemaker implantation)” by our model, but the correct concept is “单腔永久起搏器置入术”. The key to solving this problem is relying on a large number of medical professional KB.

5. Conclusions

A system that can automatically encode clinician-entered terms into ICD codes with high accuracy is of great importance to hospitals in China. It will not only save cost and time for clinical coding processes, but also improve the standardization of clinical data in China. In this paper, we proposed a stacking-BERT model for Chinese clinical entity normalization tasks which investigated the effectiveness of different BERT models. Our experiment proved that BERT-based normalization models outperformed some similarity-based methods; using the sentence-pair classification task of the original BERT architecture and the pre-trained model of Chinese can lead to satisfactory performance. In addition, we found that the tricks of adversarial training and EDA can effectively improve the effect of the deep learning model for small samples. However, our study lacks in-depth mining of Chinese clinical entity characteristics, so we are exploring the use of HowNet Sense and Lattice Graph to calculate the similarity of clinical entities.

Acknowledgments

The publication of this paper was funded by the National Engineering Laboratory for Internet Medical Systems and Applications of China (No. NELIMSA2018P02) and the program of China Knowledge Center for Engineering Sciences and Technology (Medical Knowledge Service System) (No. CKCEST-2022-1-6).

Conflict of interest

The authors declare that there is no conflict of interest.

References

1. N. Kang, B. Singh, Z. Afzal, E. M. van Mulligen, J. A. Kors, Using rule-based natural language processing to improve disease normalization in biomedical text, *J. Am. Med. Inf. Assoc.*, **20** (2013), 876–881. https://doi.org/10.1136/amiajnl-2012-001173
2. O. Ghiasvand, R. J. Kate, UWM: Disorder mention extraction from clinical text using CRFs and normalization using learned edit distance patterns, in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, (2014), 828–832. https://doi.org/10.3115/v1/S14-2147
3. O. Bodenreider, The unified medical language system (UMLS): integrating biomedical terminology, *Nucleic Acids Res.*, **32** (2004), 267–270. https://doi.org/10.1093/nar/gkh061
4. J. Jovanovi´c, E. Bagheri, Semantic annotation in biomedicine: The current landscape, *J. Biomed. Semant.*, **8** (2017), 1–18. https://doi.org/10.1186/s13326-017-0153-x
5. W. Shen, J. Wang, J. Han, Entity linking with a knowledge base: Issues, techniques, and solutions, *IEEE Trans. Knowl. Data Eng.*, **27** (2015), 443–460. https://doi.org/10.1109/TKDE.2014.2327028
6. S. Vashishth, R. Joshi, R Dutt, D. Newman-Griffis, C. Rose, MedType: improving medical entity linking with semantic type prediction, Preprint, arXiv:2005.00460. https://doi.org/10.48550/arXiv.2005.00460
7. H. Li, Q. Chen, B. Tang, X. Wang, H. Xu, B. Wang, et al., CNN-based ranking for biomedical entity normalization, *BMC Bioinf.*, **18** (2017), 385. https://doi.org/10.1186/s12859-017-1805-7
8. Y. Luo, G. Song, P. Li, Z. Qi, Multi-task medical concept normalization using multi-view convolutional neural network, in *Proceedings of the AAAI Conference on Artificial Intelligence*, **32** (2018).
9. I. Mondal, S. Purkayastha, S. Sarkar, P. Goyal, J. Pillai, A. Bhattacharyya, et al., Medical entity linking using triplet network, in *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, (2019), 95–100. https://doi.org/10.18653/v1/W19-1912
10. J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, preprint, arXiv:1810.04805. https://doi.org/10.18653/v1/N19-1423
11. J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, et al., BioBERT: a pre-trained biomedical language representation model for biomedical text mining, *Bioinformatics*, **36** (2020), 1234–1240. https://doi.org/10.1093/bioinformatics/btz682

12. K. Huang, J. Altosaar, R. Ranganath, ClinicalBERT: Modeling clinical notes and predicting hospital readmission, preprint, arXiv:1904.05342. https://doi.org/10.48550/arXiv.1904.05342
13. F. Li, Y. Jin, W. Liu, B. P. S. Rawat, P. Cai., H Yu, Fine-tuning bidirectional encoder representations from transformers (BERT)-based models on large-scale electronic health record notes: an empirical study, *JMIR Med. Inf.*, **7** (2019), e14830. https://doi.org/10.2196/14830
14. K. S. Kalyan, S. Sangeetha, BertMCN: Mapping colloquial phrases to standard medical concepts using BERT and highway network, *Artif. Intell. Med.*, **112** (2021), 102008. https://doi.org/10.1016/j.artmed.2021.102008.
15. M. Sung, H. Jeon, J. Lee, J. Kang, Biomedical entity representations with synonym marginalization, in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020. https://doi.org/10.48550/arXiv.2005.00239
16. Y. Xia, H. Zhao, K. Liu, H. Zhu, Normalization of Chinese informal medical terms based on multi-field indexing, *Commun. Comput. Inf. Sci.*, **496** (2014), 311–320. https://doi.org/10.1007/978-3-662-45924-928.
17. Y. Zhang, X. Ma, G. Song, Chinese medical concept normalization by using text and comorbidity network embedding, in *2018 IEEE International Conference on Data Mining (ICDM)*, (2018), 777–786. https://doi.org/10.1109/ICDM.2018.00093
18. Q. Wang, Z. Ji, J. Wang, S. Wu, W. Lin, W. Li, et al., A study of entity-linking methods for normalizing Chinese diagnosis and procedure terms to ICD codes, *J. Biomed. Inf.*, **105** (2020), 103418. https://doi.org/10.1016/j.jbi.2020.103418
19. CHIP 2019, Chinese Information Processing Society of China, 2021. Available from: http://www.cips-chip.org.cn/.
20. S. E. Robertson, S. Walker, S. Jones, M. Hancock-Beaulieu, M. Gatford, Okapi at TREC-3, in *Proceedings of TREC*, (1995), 109–126.
21. Z. Ji, Q. Wei, H. Xu, Bert-based ranking for biomedical entity normalization, preprint, arXiv:1908.03548. https://doi.org/10.48550/arXiv.1908.03548
22. S. Chopra, R. Hadsell, Y. Lecun, Learning a similarity metric discriminatively, with application to face verification, in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, **1** (2005), 539–546. https://doi.org/10.1109/CVPR.2005.202
23. N. Paul, M. Versteegh, M. Rotaru, Learning text similarity with siamese recurrent networks, in *Proceedings of the 1st Workshop on Representation Learning for NLP*, (2016), 149–157. https://doi.org/10.18653/v1/W16-1617
24. G. Kertész, S. Szénási, Z. Vámossy, Vehicle image matching using siamese neural networks with multi-directional image projections, in *2018 IEEE 12th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, 2018. https://doi.org/10.1109/SACI.2018.8440917
25. S. Fakhraei, J. Mathew, L. A. José, NSEEN: neural semantic embedding for entity normalization, in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Springer, Cham, (2019), 665–680. https://doi.org/10.48550/arXiv.1811.07514
26. C. May, A. Wang, S. Bordia, S. R. Bowman, R. Rudinger, On measuring social biases in sentence encoders, preprint, arXiv:1903.10561. https://doi.org/10.18653/v1/N19-1063
27. T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, BERTScore: Evaluating text generation with BERT, preprint, arXiv.1904.09675. https://doi.org/10.48550/arXiv.1904.09675

28. Y. Qiao, C. Xiong, Z. Liu, Z. Liu, Understanding the behaviors of BERT in ranking, preprint, arXiv:1904.07531. https://doi.org/10.48550/arXiv.1904.07531
29. N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, preprint, arXiv:1908.10084. https://doi.org/10.48550/arXiv.1908.10084
30. L. Shoushan, C. Huang, Chinese sentiment classification based on stacking combination method, *J. Chin. Inf. Process.*, **24** (2010), 56–61. https://doi.org/10.1109/ACCESS.2020.3007889
31. bert-base-chinese, *Hugging Face*, 2021. Available from: https://huggingface.co/bert-base-chinese/tree/main.
32. CLUEPretrainedModels, Github, 2021. Available from: https://github.com/CLUEbenchmark/CLUEPretrainedMode-ls.
33. scikit-learn: Machine Learning in Python, scikit-learn, 2022. Available from: https://scikit-learn.org/stable/.
34. T. Miyato, A. M. Dai, I. Goodfellow, Adversarial training methods for semi-supervised text classification, preprint, arXiv:1605.07725. https://doi.org/10.48550/arXiv.1605.07725
35. A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu, Towards deep learning models resistant to adversarial attacks, preprint, arXiv:1706.06083. https://doi.org/10.48550/arXiv.1706.06083
36. L. Bergroth, H. Hakonen, T. Raita, A survey of longest common subsequence algorithms, in *Proceedings Seventh International Symposium on String Processing and Information Retrieval, SPIRE 2000*, IEEE, (2000), 39–48. https://doi.org/10.1109/SPIRE.2000.878178
37. bert-as-service, Github, 2021. Available from: https://github.com/hanxiao/bert-as-service.
38. S. Sherazi, J. W. Bae, J. Y. Lee, A soft voting ensemble classifier for early prediction and diagnosis of occurrences of major adverse cardiovascular events for STEMI and NSTEMI during 2-year follow-up in patients with acute coronary syndrome, *Plos One*, **16** (2021), e0249338. https://doi.org/0.1371/journal.pone.0249338