BMC Medical Informatics and
Decision Making

RESEARCH

Open Access

Medical code prediction via capsule networks and ICD knowledge

Weidong Bao[1,2], Hongfei Lin[2], Yijia Zhang[2*], Jian Wang[2] and Shaowu Zhang[2]

From International Conference on Health Big Data and Artificial Intelligence 2020 Guangzhou, China. 29 October - 1 November 2020

Abstract

Background: Clinical notes record the health status, clinical manifestations and other detailed information of each patient. The International Classification of Diseases (ICD) codes are important labels for electronic health records. Automatic medical codes assignment to clinical notes through the deep learning model can not only improve work efficiency and accelerate the development of medical informatization but also facilitate the resolution of many issues related to medical insurance. Recently, neural network-based methods have been proposed for the automatic medical code assignment. However, in the medical field, clinical notes are usually long documents and contain many complex sentences, most of the current methods cannot effective in learning the representation of potential features from document text.

Methods: In this paper, we propose a hybrid capsule network model. Specifically, we use bi-directional LSTM (Bi-LSTM) with forwarding and backward directions to merge the information from both sides of the sequence. The label embedding framework embeds the text and labels together to leverage the label information. We then use a dynamic routing algorithm in the capsule network to extract valuable features for medical code prediction task.

Results: We applied our model to the task of automatic medical codes assignment to clinical notes and conducted a series of experiments based on MIMIC-III data. The experimental results show that our method achieves a micro F1-score of 67.5% on MIMIC-III dataset, which outperforms the other state-of-the-art methods.

Conclusions: The proposed model employed the dynamic routing algorithm and label embedding framework can effectively capture the important features across sentences. Both Capsule networks and domain knowledge are helpful for medical code prediction task.

Keywords: Clinical notes, Medical code prediction, Capsule network, Domain knowledge

Background

Clinical notes are written by the professional medical staff and record various information of patients in the hospital such as their medical history, lifestyle, symptoms, treatment, and medications. Clinical notes facilitate quick access to the patients' information by the medical staff and significantly contribute to the treatment process of the patients [1]. A medical code refers to a process of converting the clinical notes from the medical ontology into a group of medical codes. The most popular medical codes are the international disease codes ICD 9 and ICD 10. The ICD codes standardize and format the disease names which is the basis for applying

*Correspondence: zhyj@dlut.edu.cn
[2] College of Computer Science and Technology, Dalian University of Technology, Dalian, China
Full list of author information is available at the end of the article

clinical information systems such as medical information and hospital information management [2]. The ICD code enables the most comprehensive sharing of disease information, reflecting the nation's health status. It is also very helpful for medical research and teaching. The ICD is the management basis of hospital medical treatments and administration. Disease classification is also one of the vital basis of medical expenses control which is conducive to cost management and insurance reimbursements. The international disease code ICD 9 approximately includes 138,000 codes.

The process of assigning the medical codes to the clinical notes is often performed by trained professionals. Clinical notes, however, contain complex medical terminology and often include spelling errors and abbreviations. Therefore, reading the notes and assigning the corresponding codes is usually challenging for the professionals and requires a significant amount of time, human and financial resources. An automatic medical code assignment system can significantly improve this process and reduce human errors in ICD code assignment [3].

There exist several methods for automatic medical codes assignment to clinical notes. These methods are categorized into rule-based methods, machine learning methods, and neural network methods. Instances are K-nearest neighbours, correlation feedback, support vector machines [4], Bayesian ridge regression [5], and various neural network models [6, 7]. In [8], the author developed a method based on the hierarchical tree structure of the ICD-9 ontology. Larkey et al. [9] also used classifiers to obtain the candidate ICD labels, but the classifiers need to adjust all the parameters artificially. Koopman et al. [10] used SVM for automatic ICD-10 classification of cancer. Perotte et al. [11] mention a hierarchy based SVM model to predict the ICD codes for the discharge summaries. Lita et al. Further in [5] the ICD code classification performance of the SVM was investigated. In general, machine learning methods are time-consuming to design the specific features, that are also significantly dependent on professional competence and human resources.

With the development of artificial intelligence, deep learning methods have demonstrated incredible potential for automatic medical codes assignment and achieving a higher level of performance in the NLP tasks [12]. Neural networks with attention mechanisms [13] have achieved high performance in multi-label classification tasks [14]. Attention mechanisms [15] use one or more layers of the neurons to identify specific words and assign them different weights to obtain more accurate classification results. However, these methods are not effective at dealing with long texts in document classification tasks [16]. In general, the document text is much longer than sentence text, especially as some document texts contain many complex sentences. How to learn potential feature representations from document text is still considered as an open problem in the NLP domain.

Using label embedding has proven effective in various domains and tasks such as image classification in computer vision [17], multi-modal learning between images and text [18], and text recognition in images [19]. Label embedding also shows excellent performance in zero-shot learning tasks [20], where certain classes are not visible and capturing label correlation in the embedding space can improve predictive performance. Label embedding for the text multi-label classification has been studied in the heterogeneous networks [21]. Word embedding is an efficient intermediate representation for capturing semantic rules between the words for learning text sequence representations. In 2017, Hinton proposed the capsule network architecture [22], modifying the traditional CNN that can be used for performance improvement of the ICD codes assignment. Therefore, the label embedding framework and capsule network are both useful for automatic medical codes assignment tasks.

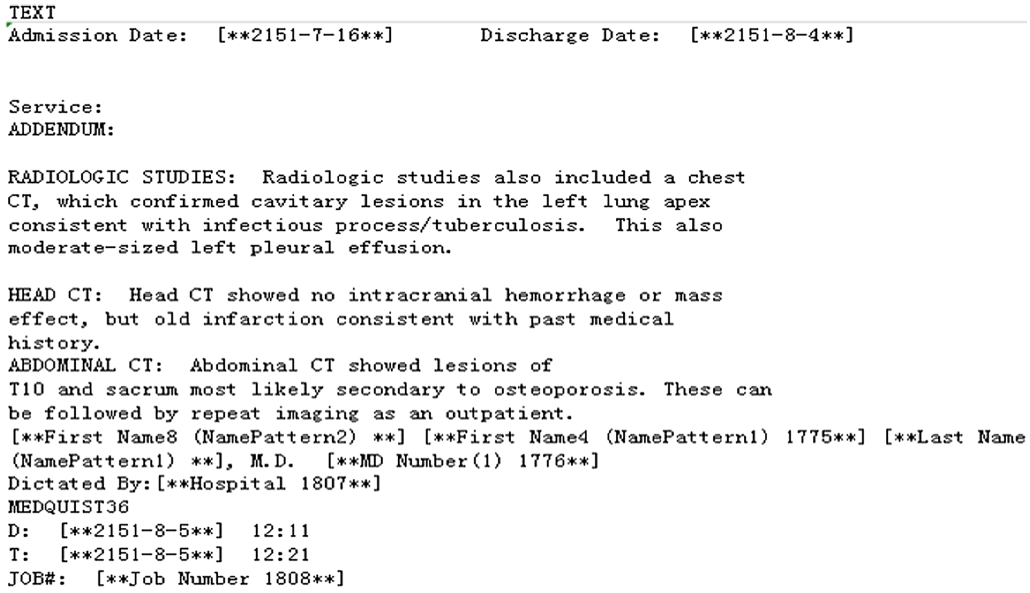
In this paper, we propose a hybrid model based on the capsule network and label embedding framework to predict the ICD codes from the clinical notes. The model effectively captures the important syntactic features between the sentences and learns a comprehensive representation of the potential features throughout the document text. Our proposed model uses a dynamic routing algorithm to capture more specific features in the clinical notes to improve accuracy, instead of filtering the features using a pooling method that may lose important information. A label embedding framework is also proposed to incorporate more information from the labels. Our approach is evaluated on the MIMIC-III public corpus and the experimental results confirm that the model achieves state-of-the-art performance.

The rest of the paper is organized as follows. We briefly describe the medical code assignment task followed by the details of our proposed model in method section. Then in results and discussions section, we show the experimental results and discussion of the MIMIC-III corpus. Finally, in conclusions section we summarize our work and conclude.

Method

Medical codes assignment

The automatic medical codes assignment is a long-term and challenging area of research. In recent years, many researchers explore the use of textual data for automatic ICD codes assignment [23], intended to automatically and accurately predict the corresponding medical



```
TEXT
Admission Date:  [**2151-7-16**]       Discharge Date:  [**2151-8-4**]


Service:
ADDENDUM:

RADIOLOGIC STUDIES:  Radiologic studies also included a chest
CT, which confirmed cavitary lesions in the left lung apex
consistent with infectious process/tuberculosis.  This also
moderate-sized left pleural effusion.

HEAD CT:  Head CT showed no intracranial hemorrhage or mass
effect, but old infarction consistent with past medical
history.
ABDOMINAL CT:  Abdominal CT showed lesions of
T10 and sacrum most likely secondary to osteoporosis. These can
be followed by repeat imaging as an outpatient.
[**First Name8 (NamePattern2) **] [**First Name4 (NamePattern1) 1775**] [**Last Name
(NamePattern1) **], M.D.  [**MD Number(1) 1776**]
Dictated By:[**Hospital 1807**]
MEDQUIST36
D:  [**2151-8-5**]  12:11
T:  [**2151-8-5**]  12:21
JOB#:  [**Job Number 1808**]
```

Fig. 1 Note events in MIMIC-III corpus

Table 1 DIAGNOSES_ICD in MIMIC-III corpus

ROW_ID	SUBJECT_ID	HADM_ID	SEQ_NUM	ICD9_CODE
1297	109	172,335	1	40,301
1298	109	172,335	2	486
1299	109	172,335	3	58,281
1300	109	172,335	4	5855
1301	109	172,335	5	4254
1302	109	172,335	6	2762
1303	109	172,335	7	45,829
1304	109	172,335	8	2875

Table 2 Example of 3-digit ICD9 description

ICD code	Description
001	Cholera
002	Typhoid and paratyphoid fevers
003	Other salmonella inflections
004	Shigellosis
005	Other food poisoning (bacterial)
006	Amebiasis
007	Other protozoal intestinal diseases

code from the clinical notes. In this paper, we choose the MIMIC-III corpus for our experimental study.

The Medical Information Mart for Intensive Care (MIMIC-III) is a large, single-centre database comprising information relating to the patients admitted to critical care units at a large tertiary care hospital [2]. Figure 1 and Table 1 provide examples of the MIMIC-III corpus. Note events contain the patient's clinical notes. DIAGNOSE_ICD contains the ICD codes corresponding to the patient's clinical notes. MIMIC-III contains 59,652 discharge records from the medical centre. There are 942 unique 3-digit ICD-9 codes in the dataset. However, the distribution of the codes is highly unbalanced. Common codes account for 26% of all codes and rare codes only account for 1%. On average, each MIMIC-III discharge note contains about 1500 words.

Our dataset is a collection of clinical notes. Each clinical note X consists of many words, and each X has a set of medical codes M corresponding to that note. Besides, there is an external knowledge source L (Table 2), which contains a basic description of each ICD code. Our goal is to add the label embedding framework to the hybrid model to predict the relevant medical codes M. The entire experiment is a multi-label text classification task.

Our model

A schematic overview of our model architecture is shown in Fig. 2. Our model consists of three main parts, including the Bi-LSTM [24] layer, the label embedding framework, and the capsule network (CapsNet) layer.

The inputs of our model are text sequences. The word embedding layer generates a distributed representation vector containing semantic information for each word. We use bi-directional LSTM (Bi-LSTM) with forwarding and backward directions to merge the information

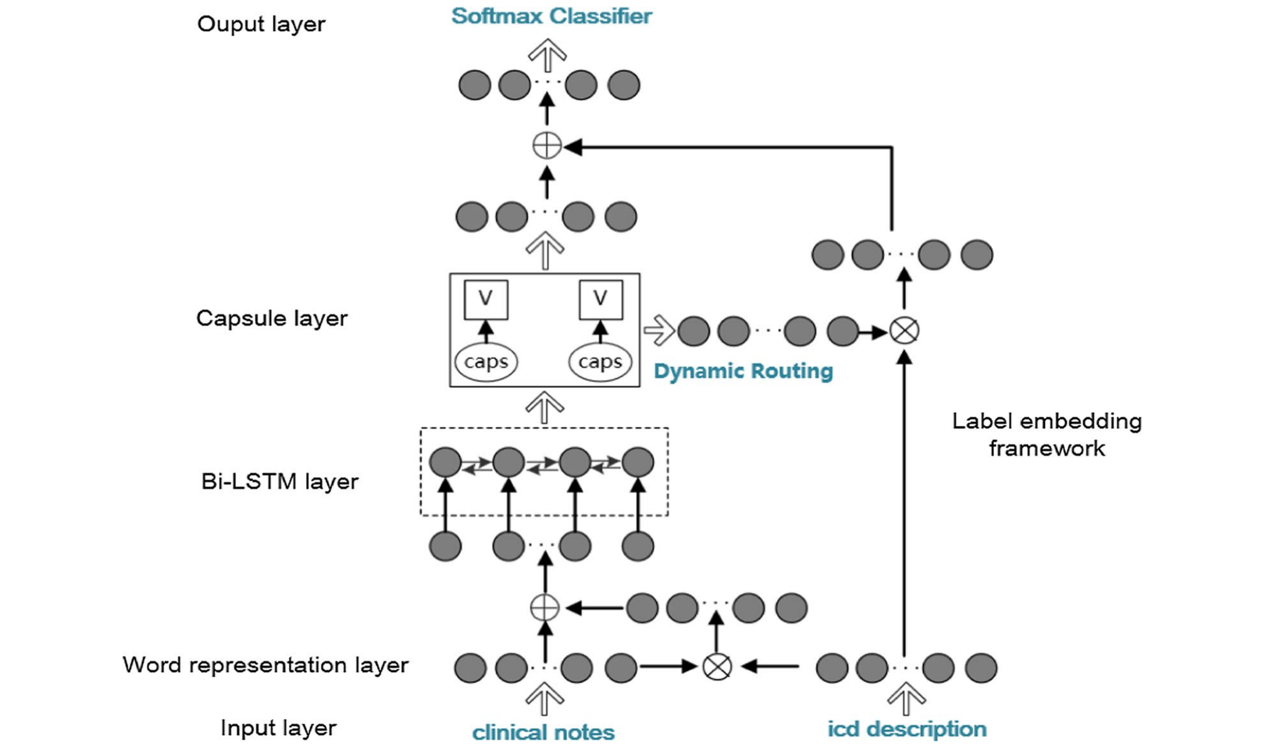


Fig. 2 Schematic overview of our model

from both sides of the sequence. It can capture the contextual information of each word to obtain a comprehensive sequence of the output vectors. The label embedding framework embeds the text and labels together to leverage the label information. We then use a dynamic routing algorithm in the capsule network to extract more valuable features. Finally, we use a fully connected layer and SoftMax function to implement the text classification. The details of our model are presented in the following sections.

Word representations

The distributed representation method, also known as word embedding [25, 26], is based on the premise that semantically similar words have similar semantics. Word embedding has been widely used in the BioNLP domain to effectively capture the underlying semantic information of the words. Since the clinical notes are usually written by the medical professionals, we adopt the distributed representation method to obtain a word vector that approximates the objective word meaning more closely. In this paper, the pre-trained word embeddings are downloaded from nlplab, using the word2vec tool to train the word vectors from PubMed and PMC texts.

Bi-LSTM

In a sentence, the semantics of the preceding and following words are related. Recurrent Neural Network (RNN) is one of the neural network models [27] which aims to capture the sequential patterns presented in data. It has however limited due to the gradient vanishing problem. The LSTM is a variant of RNN [28], which addresses the issue of long-term dependencies by introducing the gate mechanism and the memory cell. It can keep the long dependent information and conquer the gradient vanishing issue. The LSTM model overcomes the vanishing gradient problem by introducing gating mechanisms. Therefore, it is suitable to capture the long-term dependency feature.

The LSTM unit consists of three components: the input gate $\mathrm{i_t}$, the forget gate $\mathrm{f_t}$ and the output gate o_t. At the time step t, the LSTM unit utilizes the input word $\mathrm{x_t}$, the previous hidden state $\mathrm{h_{t-1}}$ and the previous cell state $\mathrm{c_{t-1}}$ to calculate the current hidden state $\mathrm{h_t}$ and cell state c_t [24]. The equations are as follows:

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \quad (1)$$

$$g_t = \tanh\left(\mathrm{W_g x_t + U_g h_{t-1} + b_g}\right) \quad (2)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{3}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{4}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{5}$$

$$h_t = o_t \odot \tanh(\mathrm{c_t}) \tag{6}$$

where W, U is weight, b is bias term, $\odot$ denote element-wise multiplication.

Figure 3 shows the architecture of the CapsNet hybrid model. In this paper, we use Bi-directional LSTM (Bi-LSTM) to integrate the information from both sides of the sequence in a forward and backward manner. It takes the word representation sequence of the text as the input and then outputs a new word representation sequence that captures the contextual information of each word in the text.

Label embedding framework

In this section, we use $\circ$ for function composition, $S = \{(X_n, Y_n)\}_{n=1}^{N}$ represents a training set, X_i represents the sequence of the clinical notes and Y_i is the corresponding ICD codes. The objective of the label embedding framework hybrid model is to learn a mapping from X to Y that makes the formula $min\frac{1}{N}\sum_{n=1}^{N}\delta\left(Y_n, f(X_n)\right)$ minimal, where δ is the loss function that measures the loss between $f(X_\mathrm{i})$ and Y_i.

Text classification is seen as a combination of three functions $f = f_0^{\circ} f_1^{\circ} f_2$. Figure 4 shows the common methods of text classification, where f_0 is a word embedding function, f_1 is a function to aggregate word embedding to get text representation, and f_2 is a function for classifying using text representation. There are two broad categories of approaches to designing f_1. One is to treat the process as a "black box" that learns mappings using various deep learning models [29, 30], while the other approaches use simple max pooling or mean pooling method. As it is seen, only f_2 used the label information, hence the impact of the label is indirect for both f_0 and f_1. Here, we propose adding the label information in each process.

Figure 5 shows how we joint label embedding framework. In f_0, the model learns the embedding of labels as the key point to influence the word embedding. In f_1, the model uses the correlation between the labels and the words for word embedding splicing. Specifically, C_i represents the i -th category of label embedding as the matrix of label embedding. We embed both words and labels into a joint space. The calculation equations are shown as follows:

$$G = C \otimes V \tag{7}$$

$$Z = f_1(G \oplus V) \tag{8}$$

$$D = Z \otimes C \tag{9}$$

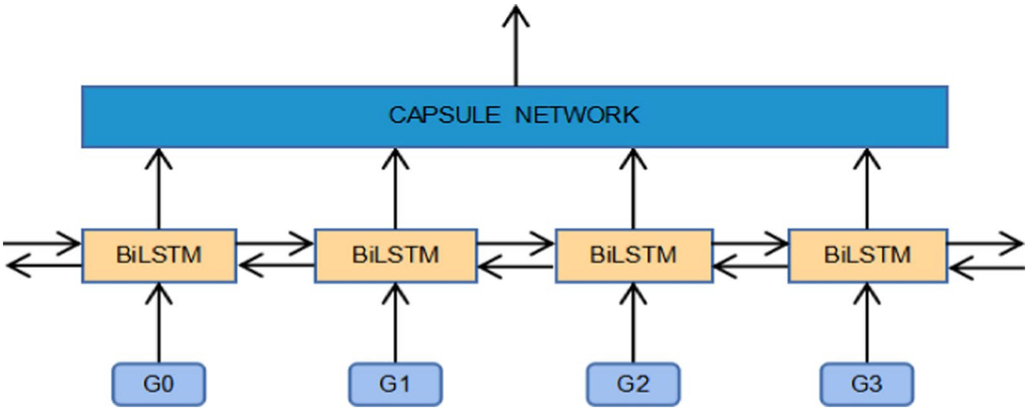


Fig. 3 The architecture of CapsNet with Bi-LSTM

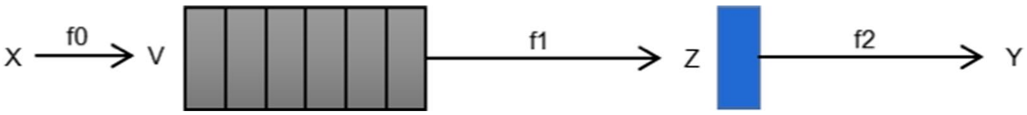


Fig. 4 Common methods to text classification

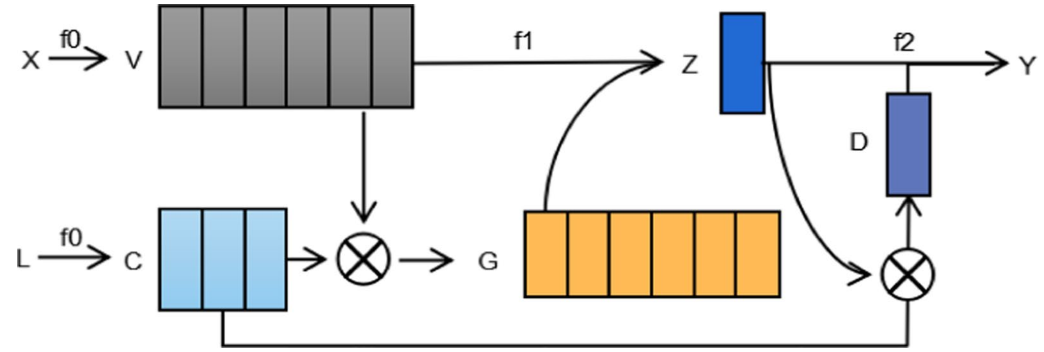


Fig. 5 Joint label embedding method

$$Y = f_2(Z \oplus D) \tag{10}$$

where ⊗ represents cosine similarity between vectors, ⊕ denotes concatenate among G and V, Z and D. In the above C and V similarities are concatenated to word embedding and fed into Bi-LSTM. The capsule network is then used to extract the features and output the [N_CAP, CAP_DIM] features, the EMB_DIM encoded sentences are combined with the label embedding to calculate the cosine similarity. The similarity is then added to the classification results.

Capsule network

In deep learning models, spatial patterns are aggregated at a lower level which helps to represent higher-level concepts. For example, CNN builds a convolutional feature detector to extract patterns from the local sequence windows and uses max pooling to select the features. The CNN extracts feature patterns at different levels in a hierarchical manner. However, since the convolution operator in the CNN is a weighted representation of the lower layers, it is difficult to represent the features of a complex object where it enters the upper layer. In this work, capsule networks are employed to learn the potential feature representation. Our capsule network architecture is shown in Fig. 6.

In the convolution layer of the capsule network, extracting N-gram features at different positions of a sentence through various convolutional filters. The N-gram size K_1 slide on the sentence to detect the features at

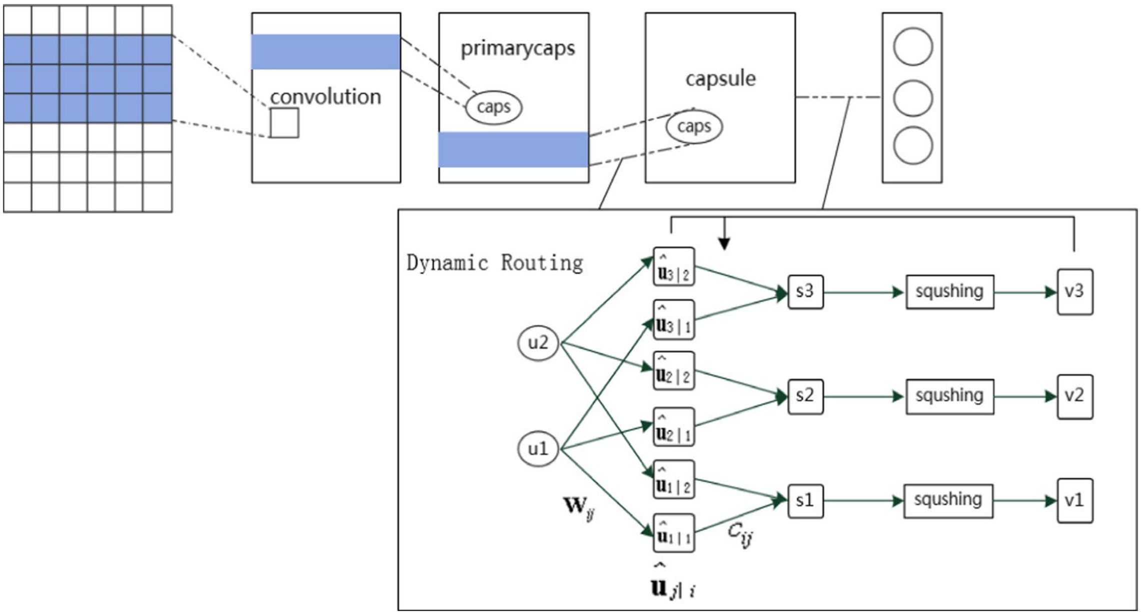


Fig. 6 capsule network model architecture

different positions. The filter of the convolution operation $W^a \in R^{K_1 \times e}$ is then convolved with the word window $\mathbf{X}_i$ at every position to generate a column feature map $\mathbf{m}^a$, $m_i^a \in R$ and $\mathbf{M}$ can be calculated as follows:

$$m_i^a = f\left(X_i^{\circ} W^a + \mathbf{b}_0\right) \tag{11}$$

$$\mathbf{M} = [\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_\mathrm{N}] \tag{12}$$

where ∘ denotes element-wise multiplication, $\mathbf{b}_0$ is a bias term, and f is a nonlinear activate function.

At the primary capsule layer, the capsule replaces the CNNs' scalar output feature detector with a vector-output capsule, it can retain more characteristic information such as the local order of words and the semantic representation of words. Let $p_i \in R^d$ denote the capsule parameter information, d be the dimension of the capsule, $W^b \in R^{N \times d}$ be the filter shared among different sliding windows. A window slides over each N-gram vector for each matrix multiplication, denoted as $\mathbf{M}_i \in R^N$. The filter W^b multiplies each N-gram vector in $\mathbf{M}_i$ to produce a column-list of the capsules $\mathbf{p}$. Each capsule $p_i \in R^d$ and $\mathbf{p}$ are calculated as follows:

$$p_i = g\left(W^b \mathbf{M}_i + \mathbf{b}_1\right) \tag{13}$$

$$\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_\mathrm{C}] \tag{14}$$

where g is a nonlinear squash function, and $\mathbf{b}_1$ is a bias term.

The capsule network uses a nonlinear function called "squashing". This nonlinear function ensures that the length of the short vector can be shortened to almost zero, while the length of the long vector is compressed to close to but no more than 1. The following is the expression for this nonlinear function:

$$\mathbf{v}_j = \frac{\left\|\mathbf{s}_j\right\|^2}{1 + \left\|\mathbf{s}_j\right\|^2} \frac{\mathbf{s}_j}{\left\|\mathbf{s}_j\right\|} \tag{15}$$

where $\mathbf{v}_j$ is the vector output of capsule j, and $\mathbf{s}_j$ is the input.

The total input for all capsules except the first layer capsule is a weighted sum of all vectors from the below layer of capsules generated by multiplying the output of the capsule layer by a weight matrix, as follows:

$$\mathbf{s}_j = \sum_i c_{ij} \widehat{\mathbf{u}}_{j|i} \tag{16}$$

$$\widehat{\mathbf{u}}_{j|i} = \mathbf{W}_{ij} \mathbf{u}_i \tag{17}$$

where c_{ij} is the coupling coefficient and is iteratively updated and determined by the dynamic routing process. The calculation method of c_{ij} is shown as follows:

$$c_{ij} = \frac{\exp\left(\mathrm{b_{ij}}\right)}{\sum_k \exp(\mathrm{b_{ik}})} \tag{18}$$

$$b_{ij} \leftarrow b_{ij} + \widehat{\mathbf{u}}_{j|i} \cdot \mathbf{v}_j \tag{19}$$

The sum of the coupling coefficients between all capsules is 1 and b_{ij} is initialized to 0, where $i \in [1, a]$, $j \in [1, k]$, and k is the number of classes.

Results and discussions

Datasets and evaluation metrics

We choose MIMIC-III dataset to evaluate our model. The total number of discharge summaries is 52,722, with an average of 1500 words per discharge summary. We preprocessed the data to remove the discontinued words, punctuation marks, and low-frequency words. The top three digits of the ICD codes for all patient visits were also aggregated then selected 344 codes from them. ICD 9 codes descriptions contain the ICD codes and the corresponding disease name descriptions. For 344 ICD codes, we found their corresponding descriptions and we used Micro F1, Macro F1, Test Loss, and Top-10 Recall to measure the performance of the model for the automatic code assignment tasks on the MIMIC-III dataset. Top-10 Recall is defined as the number of correct medical codes ranked in the top max(10, |M|) of the predicted results divided by |M|, where |M| is the number of medical codes for the note, and averaged over all test notes. The F1 score is the harmonic mean between the precision and the recall, in Micro-averaging, each forecast has the same weight and calculates the average metric over all instances. Therefore, the final result is dominated by medical codes with higher frequencies. Micro F1 gives a good overall indication of the model's performance. The macro-average metric calculates the value for each medical code separately and then takes the average of all codes. Since all the labels are given equal weights, the macro-average metric places emphasis on the prediction of the rare medical codes. However, the distribution of medical codes in MIMIC-III is not even, with the most common 10 codes occurring 26% of all codes and the least common 437 codes occurring only 1% of all codes. Therefore, the Macro F1 value in the experimental results is much lower than that of the Micro F1 value.

Experimental settings

In our experiments, our model is implemented by keras with TensorFlow backend. To accommodate the computational complexity, we initially set the model dimension in word embedding to 200, the Bidirectional LSTM hidden units to 300, and the batch size to 16. In the capsule network, the capsule dimension is 50, the dynamic routing iteration number is 3, and the learning rate is set 0.01. All parameters of the model are optimized using Adam [31] to minimize the loss of categorical cross-entropy. To ensure that other experimental results can be reproduced, we perform experiments using the same dataset and the same splitting method. In the previous study [32], the dataset was divided into a training set, a validation set, and a test set of 0.7, 0.1, and 0.2, respectively. In our experiment, we keep the same splitting of dataset and compared with the previous studies. An early stop mechanism is also used to decide where to stop the training.

Experimental results

Baseline methods

We selected three baseline models for comparison including CNN, LSTM [27], and capsule network. The model uses word embedding trained by the Word2Vec tool as the input. Here we briefly describe the selected baseline models:

Convolutional Neural Network (CNN): CNN has been successfully applied in NLP to extract the features which is a traditional method for text classification [33]. CNN is composed of an input layer, convolutional layer, activation function, pooling layer, and fully connected layer. In our work, we use three convolutional layers, the kernel size is 3, 4, and 5, respectively. Hidden dim is set to 200 and the dropout probability is set to 0.2.

Long Short-Term Memory (LSTM): LSTM is a special RNN which is mainly used to solve the problem of gradient vanishing and gradient explosion in the training process of the long sequences. Compared with the general CNN, LSTM performs better on the longer sequences. An LSTM unit has three gates including forget gate, input gate, and output gate. In our experiments, we set the hidden dim to 200, the batch size is set to 32, and the dropout is set to 0.2.

Capsule Network (CapsNet): CapsNet was originally used in the imaging field and showed excellent performance. In our work, we use a CNN-based capsule network to process text, where the scalar output of the traditional neural network is replaced with the vector output of the capsule network, and dynamic routing algorithm is used instead of the traditional pooling layer to train the neural network. The capsule vector dimension is set to 50 and the dynamic routing iteration is set to 3. The capsule network mainly consists of the convolutional layer, primary capsule layer, capsule layer and fully connected capsule layer.

We compared our proposed BiCapsNetLE method with CNN, LSTM, and capsule network models [11]. The performance of each model on the MIMIC-III dataset is shown in Table 3. The performance of the CNN model in the data set is general, with a Micro F1 score of 62.6%, the capsule network effectively improves the Micro F1 score from 62.6 to 64.7%. The results show that the capsule network's dynamic routing algorithm can capture more features from the textual information. The LSTM model relies on three gating units to have much better results for processing long text, with a Micro F1 value of 65.3%. Our proposed model is superior to the other three models in terms of all micro-measurement indicators. The value of Micro F1 is increased by 4.9% compared to CNN, 2.2% compared to LSTM, and 2.8% compared to capsule network. Macro F1 is 4.7% higher than CapsNet, 5.1% higher than CNN, 6.1% higher than LSTM. The result indicates that our model has a better improvement on rare coding prediction, and it achieves the smallest loss value among all models. This means the misclassification rate of our model is the lowest.

To visually demonstrate the impact of introducing the external knowledge sources on the experimental results, we divided medical codes into four groups according to the frequency of their occurrence in the dataset, i.e., [1, 10], [11,50], [51,100], and [100, + ∞). We further calculate the Macro AUC for all models on each group, where the Macro AUC is the unweighted average AUC over all labels. The results in Fig. 7 indicate that our model can improve the AUC of low-frequency labels in the dataset. In particular, since the medical codes with higher frequency have more examples in the learning process, they benefit less from the external sources. Conversely, because the rare codes have few examples, they can learn more from the external knowledge sources.

Comparison with related work

We compared our model with several state-of-the-art medical code assignment methods. The results are shown in Table 4. In [32], Bai et al. presented a model that incorporates the external resources of Wikipedia knowledge framework (KSI) into the model to predict ICD codes and implements the optimal performance on the MIMIC dataset. In [16], Mullenbach et al. presented

Table 3 The performance of our model and the baseline models on the MIMIC-III dataset

Methods	Macro F1 (%)	Micro F1 (%)	Test loss (%)	Top-10 recall (%)
CNN	21.4	62.6	4.0	75.3
LSTM	20.4	65.3	3.2	77.2
CapsNet	21.8	64.7	3.5	76.1
BiCapsNetLE	26.5	67.5	2.9	82.3

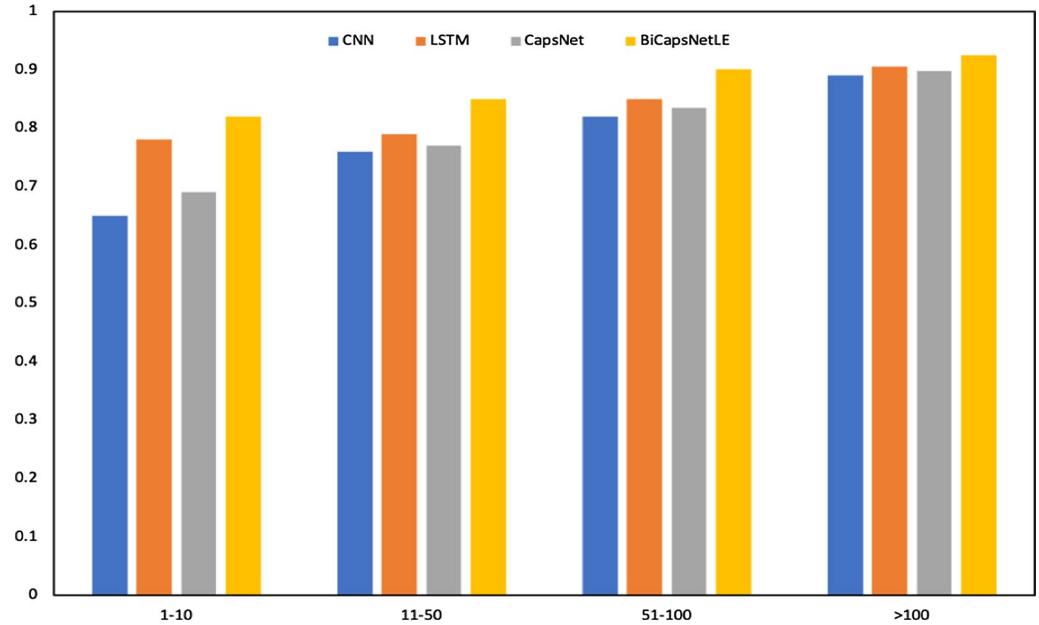


Fig. 7 Macro AUC by label frequency groups for different models. X-axis represents the label frequency groups and y-axis represents the Macro AUC

Table 4 Comparison with related works

Methods	Macro F1 (%)	Micro F1 (%)	Test loss (%)	Top-10 recall (%)
LRKSI [32]	19.6	55.7	5.5	73.8
RNNKSI [32]	24.4	66.2	3.0	79.8
CNNKSI [32]	23.7	63.7	3.9	77.5
CAML [16]	25.7	65.6	3.2	80.6
LEAM [13]	24.9	64.9	–	–
BiCapsNetLE	26.5	67.5	2.9	82.3

Table 5 Ablation studies for our model

Methods	Macro F1 (%)	Micro F1 (%)	Test loss (%)	Top-10 recall (%)
CapsNet	21.8	64.7	3.5	76.1
BiCapsNet	23.1	67.0	3.1	81.9
BiCapsNetLE	26.5	67.5	2.9	82.3

the Convolutional Attention for Multi-Label classification (CAML) model which combines the attention mechanism with CNN to achieve state-of-the-art performance in medical codes prediction. In [13], Wang et al. presented a label embedding attentive model (LEAM) to improve the text classification.

Compared to other deep neural network-based approaches, our method achieves the highest Micro F1 and Macro F1 scores of 67.5% and 26.5% for the MIMIC-III dataset. Our proposed model also outperforms other competitive models in all measurements. Comparing the results from other experiments, we improved Micro F1 by 11.8% compared to LRKSI, 1.3% compared to RNNKSI, 3.8% compared to CNNKSI, 1.9% compared to CAML, 2.6% compared to LEAM, and Macro F1 is increased by 6.9% compared to LRKSI, 2.1% compared to RNNKSI, 2.8% compared to CNNKSI, 0.8% compared to CAML, 1.6% compared to LEAM. Our proposed model also achieved the smallest loss value amongst the investigated models. In addition to using label embedding framework to acquire more important features, we use the Bi-LSTM and capsule network to automatically and efficiently capture the valuable features from documents. More importantly, our model is relatively simple, unlike the integrated model which requires the construction of a large number of neural network models that requires more time and effort to be integrated.

Ablation study

To examine the contribution of each component of the model, we conducted an ablation study. The experimental results are shown in Table 5.

It is seen that removing the Bi-LSTM layer or the label embedding framework results in reducing the model's performance. This suggests that both layers can learn effective features. By adding the Bi-LSTM layer to the capsule network model, the Micro F1 value is increased by 2.3%, and by adding the label embedding framework, the Micro F1 is increased by another 0.5%.

Using the capsule network alone also leads to the smallest F1 score and the largest loss. This is mainly due to its intrinsic shortcomings in dealing with long texts. In the MIMIC-III dataset, the average length of the discharge summaries exceeds 1000 words. The capsule network cannot remember features of long text sequences well. This may lead to poor performance. In particular, it is seen that when the capsule network is combined with the Bi-LSTM layer, the performance has been greatly improved. This indicates the effectiveness of Bi-LSTM and shows that the hybrid model can learn more long text features. The label embedding framework can significantly improve the scores of other test results, the results illustrate that the label embedding framework can provide more useful label information to add to the model label prediction task, and also effectively improves the sparsity of rare ICD codes distribution.

Relational visualization

To show the strength of the connection between the capsule layers, we use the primary capsule layer to directly connect the last fully connected capsule layer, where the primary capsule represents the N-gram words in the form of a capsule. The connection strength can show the importance of each primary capsule for the text category, like attention mechanism. This allows the capsule network to recognize multiple medical codes in the clinical notes.

In Table 6, we have manually annotated several words related to the ICD disease labels (e.g., "acid" and "atrium"), highlighted in bold for reference. We used WordCloud to visualize 3-g words for the ICD 276 and ICD 427 categories. The stronger the connection strength, the larger the font size. In Table 7, from the results, we observe that the capsule network correctly identifies and clusters the important words about the text category. Histograms are also used to show the strength of the connections between the primary capsule and the fully connected capsule.

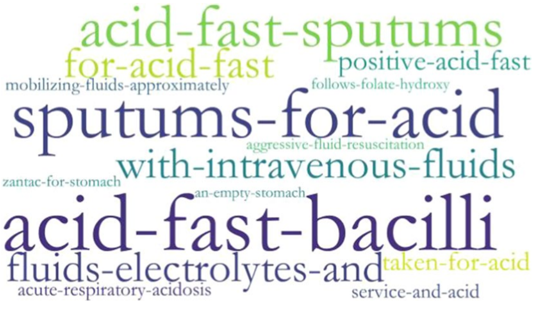
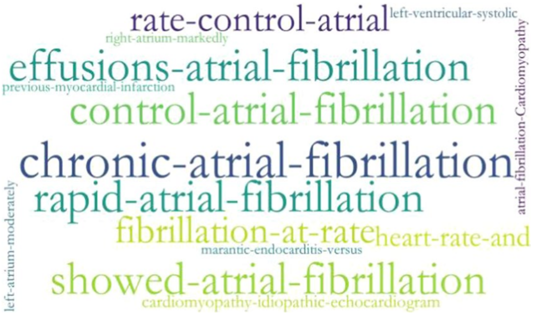
Error analysis

To better understand our proposed model, we performed an error analysis of the final output. There are two main types of errors: False-positive (FP) errors and True-negative (TN) errors. We list some examples to analyze these errors and find that most of the FP occur in similar ICD codes. Prediction performance heavily depends on the frequency of code occurrences in the data. This means that the higher the frequency of ICD code, the higher the AUC values. In contrast, low-frequency labels exhibit random performance. For two similar codes, if one occurs more frequently than the other, the low-frequency code may be easily predicted to be a high-frequency code, the least common 437 codes in the MIMIC-III

Table 6 Partial clinical note

Clinical note
Fluids, **electrolytes and** nutrition—Initial dehydration was treated **with intravenous fluids**. She tolerated p.o. throughout her placement and was placed on a thin pureed diet with supplemental Boost at breakfast, lunch and dinner. This was following swallow and nutrition evaluations for malnutrition. **Acid fast sputums** times three were taken for **acid fast bacilli**. On hospital day #4 sputum [**Doctor Last Name 1770**] from hospital day #2 showed positive **acid fast bacilli**. Infectious Disease consult was called. Cardiovascular—As noted above, the patient was in **rapid atrial fibrillation** initially. By hospital day #2 and throughout course the patient remained in rate control atrial fibrillation. **Chronic atrial fibrillation**. Intermittent left bundle branch block

Table 7 WordCloud visualization

Disorders of fluid electrolyte and acid–base balance (ICD 276)	Atrial fibrillation and flutter (ICD 427)
	

dataset accounted for only 1% of the occurrences. For example, "007" (protozoal intestinal disease, frequency 19) was incorrectly predicted as "008" (pathogenic intestinal infection, frequency 579). The reason why the actual positive prediction was negative is that some strong positive keywords were missing, or positive indicators did not appear in the article. In cases where there are few (or no) expressions in the training set, it is extremely difficult to accurately classify real positive labels. For instance, a certain clinical note describes amoebiasis, but common discriminating keywords, such as "ameba", rarely appear in the training set. On the contrary, other disease-related words appear frequently. Our model incorrectly classifies positive instances as negative ones. In the future, we will consider better pre-processing and post-processing technologies to address the abovementioned issues.

Conclusions

Prediction of medical codes for clinical notes is one of the pivotal tasks of biomedical NLP. The capsule network is a highly promising neural network model which has its unique advantages in biomedical text classification. In this paper, we propose the application of capsule network in the field of medical code prediction studies of clinical notes and propose the inclusion of the label embedding framework. We also compare our proposed model with various models and demonstrate that the capsule network hybrid model is indeed useful for medical code prediction. Besides, we propose a more efficient label embedding framework, which not only alleviates the problem of sparse distribution of rare medical codes but also leads to higher classification accuracy. Experimental results show that our hybrid model can effectively combine the advantages of capsule network and label embedding frameworks, and our hybrid model achieves state-of-the-art performance on the automatic medical code assignment task.

Current state-of-the-art methods in ICD code prediction are mainly based on supervised machine learning and a key challenge in the field is to reduce the model's reliance on the tag training data. As a future research direction, we would try to employ semi-supervised or migration learning in medical code prediction task and integrate biomedical knowledge to further improve the performance of our experiments.

Abbreviations

ICD: The International Classification of Diseases; SVM: Support vector machine; NLP: Natural language processing; BioNLP: Biomedical natural language processing; CNN: Convolutional neural network; RNN: Recurrent neural network; LSTM: Long short-term memory; Bi-LSTM: Bi-directional long short-term memory; CapsNet: Capsule network; KSI: Knowledge source integration; LRKSI: Logistic regression with knowledge source integration; RNNKSI: Recurrent neural network with knowledge source integration; CNNKSI: Convolutional neural network with knowledge source integration; CAML: Convolutional attention for multi-label classification; LEAM: Label embedding attentive model; BiCapsNet: Bi-directional Long Short-Term Memory Capsule Network; BiCapsNetLE: Bi-directional Long Short-Term Memory Capsule Network with Label Embedding; AUC: Area under the curve; TN: True-negative; FP: False-positive.

Acknowledgements

The authors would like to thank our group members for their valuable suggestions.

About this supplement

This article has been published as part of BMC Medical Informatics and Decision Making Volume 21, Supplement 2 2021: Health Big Data and Artificial Intelligence. The full contents of the supplement are available at https://bmcmedinformdecismak.biomedcentral.com/articles/supplements/volume-21-supplement-2.

Authors' contributions

Both ZYJ and LHF designed the method and experiments. BWD and WJ performed the experiments and analyzed the results. BWD and ZSW wrote the manuscript. All authors read and approved the final manuscript.

Funding

The publication costs were funded by the National Natural Science Foundation of China No. 62072070.

Availability of data and materials

The MIMIC III datasets are available at https://mimic.physionet.org/. The pre-trained word embeddings are available at http://bio.nlplab.org/.

Ethics approval and consent to participate

No ethics approval was required for the study.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

[1] School of Information Engineering, Dalian Ocean University, Dalian, China.
[2] College of Computer Science and Technology, Dalian University of Technology, Dalian, China.

Received: 4 February 2021 Accepted: 8 February 2021
Published: 30 July 2021

References

1. Hebdomadaire whojwerré: International classification of diseases-ninth revision (ICD-9). 1988, 63(45):343–4.
2. Johnson AE, Pollard TJ, Shen L, Li-Wei HL, Feng M, Ghassemi M, Moody B, Szolovits P, Celi LA, Mark RG. MIMIC-III, a freely accessible critical care database. Sci Data. 2016;3(1):1–9.
3. de Lima LR, Laender AH, Ribeiro-Neto BA. A hierarchical approach to the automatic categorization of medical documents. In: *Proceedings of the seventh international conference on Information and knowledge management*; 1998. p. 132–9.
4. Farkas R, Szarvas G. Automatic construction of rule-based ICD-9-CM coding systems. BMC Bioinform. 2008;9(3):S10.
5. Lita LV, Yu S, Niculescu S, Bi J. Large scale diagnostic code classification for medical patient records. In: *Proceedings of the third international joint conference on natural language processing*: volume-II; *2008*.
6. Choi E, Bahadori MT, Schuetz A, Stewart WF, Sun J. Doctor ai: predicting clinical events via recurrent neural networks. In: *Machine learning for healthcare conference*; 2016. p. 301–18.
7. Miotto R, Li L, Kidd BA, Dudley JT. Deep patient: an unsupervised representation to predict the future of patients from the electronic health records. Sci Rep. 2016;6(1):1–10.

8. Perotte A, Wood F, Elhadad N, Bartlett N. Hierarchically supervised latent Dirichlet allocation. Adv Neural Inf Process Syst. 2011;24:2609–17.
9. Larkey LS, Croft WB. Combining classifiers in text categorization. In: *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval; Zurich, Switzerland*. Association for Computing Machinery; 1996. p. 289–97.
10. Koopman B, Zuccon G, Nguyen A, Bergheim A, Grayson N. Automatic ICD-10 classification of cancers from free-text death certificates. Int J Med Inform. 2015;84(11):956–65.
11. Perotte A, Pivovarov R, Natarajan K, Weiskopf N, Wood F, Elhadad N. Diagnosis code assignment: models and evaluation metrics. J Am Med Inform Assoc. 2014;21(2):231–7.
12. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436–44.
13. Wang G, Li C, Wang W, Zhang Y, Shen D, Zhang X, Henao R, Carin L. Joint embedding of words and labels for text classification. In: *Proceedings of the 56th annual meeting of the association for computational linguistics.* volume 1: Long Papers. 2018; 2018:2321–31.
14. Baumel T, Nassour-Kassis J, Cohen R, Elhadad M, Elhadad N. Multi-label classification of patient notes: case study on ICD code assignment. In: *AAAI conference on artificial intelligence workshops*. 2018; New Orleans, Louisiana.
15. Shi H, Xie P, Hu Z, Zhang M, Xing EP. Towards automated ICD coding using deep learning. 2017. arXiv:1711.04075.
16. Mullenbach J, Wiegreffe S, Duke J, Sun J, Eisenstein J. Explainable prediction of medical codes from clinical text. In: *Proceedings of the 2018 conference of the North American Chapter of the association for computational linguistics: human language technologies*, volume 1: Long Papers: 2018;2018:1101–11.
17. Akata Z, Perronnin F, Harchaoui Z, Schmid C. Label-embedding for image classification. IEEE Trans Pattern Anal Mach Intell. 2015;38(7):1425–38.
18. Frome A, Corrado GS, Shlens J, Bengio S, Dean J, Ranzato MA, Mikolov T. Devise: A deep visual-semantic embedding model. In: *Advances in neural information processing systems*; 2013. p. 2121–9.
19. Akata Z, Perronnin F, Harchaoui Z, Schmid C. Label-embedding for attribute-based classification. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*; 2013. p. 819–26.
20. Palatucci M, Pomerleau D, Hinton GE, Mitchell TM. Zero-shot learning with semantic output codes. In: *Advances in neural information processing systems*; 2009. p. 1410–8.
21. Tang J, Qu M, Mei Q. Pte: predictive text embedding through large-scale heterogeneous text networks. In: *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*; 2015. p. 1165–74.
22. Sabour S, Frosst N, Hinton GE: Dynamic routing between capsules. In: *Advances in neural information processing systems: 2017*. 3856–3866.
23. Crammer K, Dredze M, Ganchev K, Talukdar P, Carroll S. Automatic code assignment to medical text. In: *Biological, translational, and clinical language processing: 2007*. p. 129–36.
24. Schuster M, Paliwal KK. Bidirectional recurrent neural networks. IEEE Trans Signal Process. 1997;45(11):2673–81.
25. Pennington J, Socher R, Manning CD. Glove: Global vectors for word representation. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014;2014:1532–43.
26. Mikolov T, Sutskever I, Chen K, Corrado G, Dean J. Distributed representations of words and phrases and their compositionality. In: *Proceedings of the 26th international conference on neural information processing systems* - Volume 2. Lake Tahoe, Nevada: Curran Associates Inc.; 2013:3111–19.
27. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735–80.
28. Ma F, Chitta R, Zhou J, You Q, Sun T, Gao J. Dipole: Diagnosis prediction in healthcare via attention-based bidirectional recurrent neural networks. In: *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining: 2017*. 1903–1911.
29. Minaee S, Kalchbrenner N, Cambria E, Nikzad N, Chenaghlu M, Gao J. Deep learning based text classification: a comprehensive review. 2020.
30. Zhou Y. A review of text classification based on deep learning. In: *Proceedings of the 2020 3rd international conference on geoinformatics and data analysis*. 2020. p. 132–6.
31. Kingma DP, Ba J: Adam: A method for stochastic optimization. In: *Proceedings of the 3rd International Conference on Learning Representations*, ICLR 2015, San Diego, CA, USA, May 7-9, 2015.
32. Bai T, Vucetic S. Improving medical code prediction from clinical text via incorporating online knowledge sources. In: *The World Wide Web conference*; 2019. p. 72–82.
33. Hughes M, Li I, Kotoulas S, Suzumura T. Medical text classification using convolutional neural networks. Stud Health Technol Inform. 2017;235:246–50.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.