Coupled Graph Convolutional Neural Networks for Text-oriented Clinical Diagnosis Inference

Ning Liu[1], Wei Zhang[2], Xiuxing Li[1], Haitao Yuan[1], and Jianyong Wang[1]

[1] Tsinghua University
[2] East China Normal University
{victorliucs,zhangwei.thu2011}@gmail.com
{lixx16,yht16}@mails.tsinghua.edu.cn
jianyong@mail.tsinghua.edu.cn

Abstract. Text-oriented clinical diagnosis inference is to predict a set of diagnoses for a specific patient given its medical notes. Due to the great potential of automatic diagnosis inference, machine learning methods have began to be applied to this domain. However, existing approaches focus on performing either labor-intensive feature engineering or sequential modeling of each medical note separately, without considering the information sharing among similar patients, which is essential for evidence-based medicine, an emerging new diagnosis process. Motivated by this issue and the recently proposed graph convolutional network (GCN) for text classification, we propose to apply GCN for the text-oriented clinical diagnosis inference task. To encode the comorbidity of diagnoses into the GCN model and allow information sharing between patients, we devise a coupled graph convolutional neural networks (CGCN), where a note-dependent graph and a label-dependent graph are learned collaboratively with hyperplane projection to ensure they are in the same semantic space. The comprehensive results on two real datasets show that our method outperforms the state-of-art methods in text-oriented diagnosis inference.

Keywords: Medical Data Mining · Graph Neural Network · Diagnosis Inference

1 Introduction

The computer aided auto-diagnosis system can reduce the burden of clinical diagnosis inference by providing the most probable diagnostic options. Among the different types of sources, medical notes can give more descriptive information about the patients such as the past disease information, cure measurements, etc. The experts make accurate diagnoses from two aspects: one is from the patient's medical records and the other is from the disease information. However, automatic text-oriented clinical diagnosis inference can be challenging. Firstly, learning a better representation of patients via their medical documents is difficult. With the rapid development of the medicine, evidence-based medicine (EBM) [26] has been widely used in clinical decision-making process where the

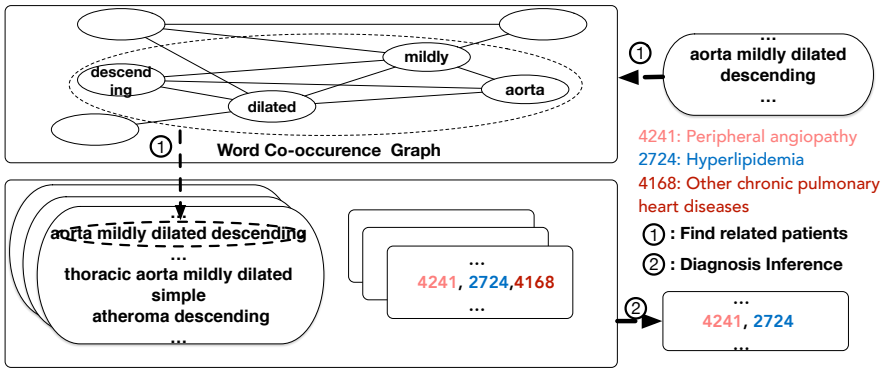

Fig. 1. The process of evidence-based medicine(left bottom part) and information sharing via word co-occurrence graph(left upper part). When diagnosing a patient, the experts firstly find the related patients in the patient corpus and give the diagnoses upon their knowledge and the existing evidences from the related patients. Relations between the two patients can be described by the common words(aorta,..,descending).

diagnoses of a patients are not only judged by the experts' knowledge but also influenced by the shared information with related patients who have similar symptoms. The simulated procedure of evidence-based medicine is shown in Figure 1. Compared with the traditional decision-making process, EBM encourages the evidences shared among similar patients to improve the quality of the healthcare [8]. Since the information sharing is eagerly stressed in the process of evidence-based medicine, the embedding of a patient should consider not only the text information describing the symptoms of the patient but also the evidences obtained from other related patients for making accurate diagnoses. Secondly, compared to traditional multilabel classification, learning a good embedding of diagnoses considering diagnosis information is quite important. Among the disease information, disease comorbidity is an essential factor and the disease comorbidity is quite common in many diseases such as Epilepsy [13], Rheumatoid Arthritis [25], Cluster headache [19] and so on.

Recently, deep learning methods have been applied to the document classification such as the convolutional neural networks [14] and recurrent neural networks [22, 2] to capture the word sequence information hidden in the texts. And LEAM [29] uses attention mechanism to jointly learn both word embeddings and label embeddings. These methods can capture semantic and syntactic information in local consecutive word sequences well. And a condensed memory neural network is proposed for clinical diagnosis inference which uses outer knowledge such as Wikipedia [24]. However, they represent text as a vector and ignore the information sharing among the patient corpus. In the previous work [34], the texts are represented as a graph to learn a text graph neural networks which shows great improvements on multi-class text classifications due to the information transfer among the text corpus. Yet it is not applied to the clinical diagnosis inference problem. Therefore, in the previous works, those methods

either not explicitly model the disease embeddings with the disease information or unable to combine the information from other related patients' medical documents.

To address the above challenges, we propose a novel Coupled Graph Convolutional Network (CGCN) and model the text-oriented clinical diagnosis inference in two aspects: one is from the aspects of patients and the other is from the aspects of diagnoses. In order to allow information sharing from the related patients' medical documents, we build a large hetergeneous two-level text corpus graph where the upper level node is the medical document node and the bottom level node is the word node where the relations between the medical documents can be linked via the common words in the global word co-occurrence graph (See left upper part of Figure 1). And then, we formulate the process of patient's embedding learning as the learning process of document node embedding learning in a graph and apply Graph Convolution Neural Network on the text corpus graph to gather high order neighbor information. Similarly, we construct diagnosis comorbidity graph and consider diagnoses as the nodes and apply Graph Convolutional Neural Network to update the embeddings of diagnoses via the comorbidity information. After that, we use a hyperplane project method to project the embeddings of diagnoses to the related hyperplane of the patients' embedding space and compute the probability of each diagosis to each patients and make the final predictions.

To summarize, our contributions are listed as the followings:

- We propose a novel Coupled Graph Neural Network (CGCN) in clinical diagnosis inference. To the best of our knowledge, we are the first to apply graph convolutional neural networks in the clinical diagnosis inference problem.
- The CGCN uses two seperate Graph Convontional Networks to joinly learn embeddings of patients and diagnoses considering the realtions between the patients and the relations between the diagnoses.
- In order to compute the match score in the same hyperplane, we use hyperplane projection to project the diagnosis embeddings into the related hyperplane of the patient embeddings.
- Experiments on two medical datasets show the CGCN outperforms the state-of-art methods. Besides, we develop an inductive learning framework for CGCN and Text GCN and external experiments show that our model still outperforms the state-of-art methods on the inductive settings.

2 Related Work

2.1 Clinicial Diagnosis Inference

The problem of clinical diagnosis inference has been studied for over twenty five years. Binaghi et al. use an an artificial neural network in the diagnosis of acute coronary occlusion [3]. And rule based methods are performed on clinical datasets [4]. In the previous studies, deep learning technologies show great

strengths over traditional methods and have been widely used in clinical diagnosis inference. Recurrent neural networks are used for modeling the time dependency in medical data [17, 6] and hiearchical attention neural networks are used to model long medical texts [2]. Besides, medical knowledge is further used to improve the performance of deep learning models. For example, C-MemNN [24] uses Wikepedia as external knowledge in predicting patients' diagnosis. However, these methods have several drawbacks. Firstly, these methods are either based on structured features or sequential representation of notes which cannot meet the requirements of evidence-based medicine with the information sharing among the patients. Secondly, the relations between diagnoses are not modelled.

2.2 Text Classification

Text Classification studies mainly focus on obtaining better text representations and can be categorized into two groups. One group focus on feature engineering. Manevitz et al.[20] uses different features, including term frequency representation and term frequency-inverse document frequency (TF-IDF) to represent texts. In [31], support vector machines and navies Bayes are used with word-level features as well. Some other works focus on indirect features learned from other models. For example, Ghassemi et.al.[9] uses topic distributions learned from LDA to make mortality predictions.

In the past few years, deep learning has been widely used in document classification and achieved remarkable success. Convolutional neural networks are used for sentence classification [14]. Wang et al. [30] used convolutional neural networks which combine knowledge and character level features for classifying short text. Despite convolutional networks, recurrent neural networks [33, 2] and recursive neural networks [11] also find their applications in text classification. Zhang et al. [35] learned label embeddings via multitask framework while LEAM [29] uses attention mechanism in learning the word and label embeddings. Further, Liu et al. [18] uses convolutional neural networks for the text based mortality prediciton. These deep learning based methods consider the text as a sequential representation and cannot capture advanced text representations. In order to capture the global word co-occurrence information, Yao et.al. [34] exploited graph convolutional neural networks in text based multi-class classification and achieve the state-of-art performance. Xue et al. [32] developed adversarial mutual learning to address the text classification under the unsupervised domain adaptation setting. However, they are not used in the clinical diagnosis inference problem and the diagnosis comorbidity is not modelled.

2.3 Graph Neural Networks

Graph neural networks are designed to handle graph structure data. While the traditional neural networks are capable of handling with structured data such as text sequences or image, they cannot handle the semi-structured data such as graphs, trees and so on, which drives the studies of graph neural networks [36, 28]. Traditional convolutional neural networks with the properties of local

connection, shared weights have gained success in various applications such as image classification, text classification and motivate the research on convolutions on the arbitrarily graphs [28]. The spectral-based graph convolutional networks [5, 10, 16] are based on spectral therory in the graph signal processing and use the orthonormal basis formed by eigenvectors of normalized graph Laplacian to transformer the input signals and have been applied into many domains due to their information sharing among the neighbors. Recently, GCNs are widely used to encode advanced graph structures in various tasks such as computational drug development and discovery [28], relation classification [27], machine translation [1], multi-class text classification [7, 16, 23, 34].

3 Coupled Graph Neural Networks

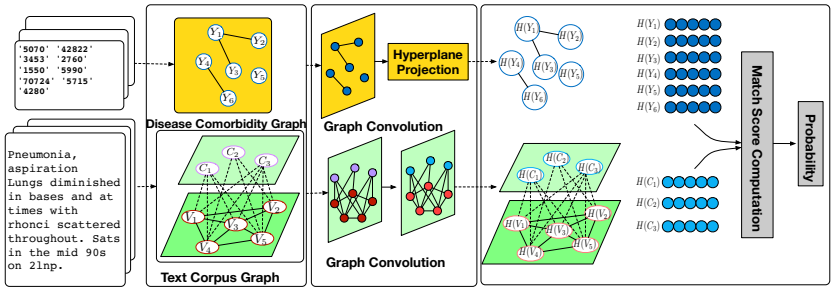

Fig. 2. The overal structure of Coupled Graph Convolutional Neural Network. Y_i indicates the diagnosis, C_i indicates the medical notes indexed by i and $H(.)$ indicates the representations of words, medical documents or diagnoses.

In this section, we will give a detailed information of Coupled Graph Convolutional Networks(CGCN). To begin with, we formulate the problem. And then, we introduce the process of constructing text corpus graph and diagnosis comorbidity graph. After that, we give a brief introduction of graph convolutional neural networks. Finally, we give a detailed overview of our model structure.

3.1 Problem Formulation

Formally, the basic elements can be defined as $(\mathcal{V}, \mathcal{C}, \mathcal{T}, \mathcal{Y}, \mathcal{G}_\mathcal{C}, \mathcal{G}_\mathcal{T})$ where $\mathcal{V}$ is the vocabulary set, $\mathcal{C}$ is the text corpus set, $\mathcal{T}$ is the diagnosis set, Y_i is the diganoses of the document i, $\mathcal{G}_\mathcal{C}$ is the text corpus graph and $\mathcal{G}_\mathcal{T}$ is the diagnosis comorbidity graph. Therefore, the goal of our method is to learn the function $\mathcal{F}$ to predict the set of diagnoses given the text $c \in \mathcal{C}$, $\mathcal{G}_\mathcal{C}$ and $\mathcal{G}_\mathcal{T}$:

$$\mathcal{F}(c, \mathcal{G}_\mathcal{C}, \mathcal{G}_\mathcal{T}) \rightarrow Y_i \tag{1}$$

3.2 Graph Construction

In this section, we will give the detailed information of constructing text corpus graph $\mathcal{G}_{\mathcal{C}}$ and diagnosis comorbidity graph $\mathcal{G}_{\mathcal{T}}$.

For text corpus graph construction, we build a large and heterogeneous text graph which consider texts and words as nodes of the graph. The graph is built in two levels: one is from the bottom word co-occurrence information and the other is built from word to text relations.

For word graph construction, we use sliding window based methods to compute the co-occurrence of words. To get the global word co-occurrence information, we use a fix size of sliding window over the text corpus $\mathcal{C}$ and get a window set defined as W and then the word co-occurrence graph between the words are computed based on the window set. We use the point-wise mutual information to define the relations between words. For each word $v \in \mathcal{V}$, we define the probability of occurrence using the sliding window based methods, that as:

$$p(v_i) = \frac{W(v_i)}{\mid W \mid} \tag{2}$$

where $W(v_i)$ is the number of windows that contain v_i. Similarly, we define the probability of word co-occurrence between word v_i and v_j as:

$$p(v_i, v_j) = \frac{W(v_i, v_j)}{\mid W \mid} \tag{3}$$

Given the probability definitions above, we can get the weight of word v_i and word v_j :

$$R(v_i, v_j) = log \frac{p(v_i, v_j)}{p(v_i)p(v_j)} \tag{4}$$

Follow the above constructions, we can get the word graph as $G_w = (V, E_w)$, where

$$E_w = \{(i, j, R(i, j)) \mid i, j \in \mathcal{V}, R(i, j) > 0\} \tag{5}$$

For word-document graph construction, some methods such as bag of words(BOW), term frequency-inverse document frequency(TF-IDF) can be used to describe the relations between words and corresponding texts. In our method, we use TF-IDF to model the relations between the words and medical documents because TF-IDF will assign a higher weight to those more descriptive words to ensure the correct information sharing between words and medical documents. Therefore, the word to document graph G_d can be defined as $G_d = (N, E_d)$ where $N = \mathcal{C} \cup \mathcal{V}$ and E_d is defined as:

$$E_d = \{(i, j, TFIDF(i, j)) \mid i \in \mathcal{V}, j \in \mathcal{C} \mid\mid i \in \mathcal{C}, j \in \mathcal{V}\} \tag{6}$$

Follow the definitions of G_w and G_d, we get the text corpus graph $\mathcal{G}_{\mathcal{C}} = (N, E_C)$ where $N = \mathcal{C} \cup \mathcal{V}$ and $E_C = E_d \cup E_w \cup \{(i, i, 1) \mid i \in N\}$

For the diagnosis comorbidity graph construction, as the number of diagnoses is further less than the number of documents and words, we compute the

frequency of any two diagnoses in the diagnosis set and construct the diagnosis comorbidity graph. And the comorbidity score between diagnosis i and j is defined as:

$$s(i,j) = \frac{No(i,j)}{\mid \mathcal{C} \mid} \tag{7}$$

where $No(i,j)$ indicates the number of patients are diagnosed with disease i and j. Therefore, the diagnosis comorbidity graph can be defined as $\mathcal{G}_{\mathcal{T}} = (\mathcal{T}, E_T)$ where

$$E_T = \{s(i,j) \mid i,j \in \mathcal{T}\} \tag{8}$$

From the above definitions, we can get text corpus graph $\mathcal{G}_{\mathcal{C}}$ and diagnosis comorbidity graph $\mathcal{G}_{\mathcal{T}}$. Although there are some knowledge graph built for diagnoses, such as the relations in ICD-10[3], we find that the disease hierarchical relation matrix is very sparse so that we use the statistics from the dataset directly.

3.3 Graph Convolutional Neural Network

In formal, the graph can be defined as $G = (V, E)$ where V is the set of nodes and E is the set of edges and the node features are defined as a matrix defined as $X \in R^{|V| \times d}$ where d is the number of node features. For computation efficiency, the graph G is represented in an adjacency matrix $A \in R^{|V| \times |V|}$ and we define D as the degree matrix of A where D is a diagonal matrix and $D_{ii} = \sum_j A_{ij}$. Graph Convolutional Neural Networks(GCN) take the node features X and graph structure A as its input and update the embeddings of the nodes using the information from the neighbors. In the spectral graph convolution theory, the graph convolution of a filter g_θ and graph G in is defined as:

$$X *_G g_\theta = U g_\theta U^T X \tag{9}$$

where $L = I_n - D^{-\frac{1}{2}} A D^{-\frac{1}{2}} = U M U^T$ is a normalized graph Laplacian matrix and U is the eigenvectors ordered by eigenvalues. Following the above definitions, Kipf and Welling et al. [16] provide a simple approximation for graph convolution computation. And the convolution over graph is defined as the following:

$$x *_G g_\theta \approx \theta(I_n + D^{-\frac{1}{2}} A D^{-\frac{1}{2}} x) \tag{10}$$

And then, Equation 10 is further modified to a compositional layer which is defined as :

$$H = f(\tilde{A} X \Theta)) \tag{11}$$

where $\tilde{A} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}} + I_n$ and $\Theta \in R^{|V| \times m}$ is a weight matrix and f is the activation function. In order to reduce the numerical instability, GCN introduces a normalised trick and the graph convolution can be defined as the followings:

$$H = f(\hat{A} X \Theta)) \tag{12}$$

[3] https://en.wikipedia.org/wiki/ICD-10

where $\hat{A} = \hat{D}^{-\frac{1}{2}}(A + I_n)\hat{D}^{-\frac{1}{2}}$ with $\hat{D} = \sum_j (A + I_n)_{ij}$

Following the above definitions, we can define one layer graph convolution as the following:

$$H_0 = f(\hat{A}X\Theta_0)) \tag{13}$$

And the stacked multi-layer graph convolution which can obtain higher order information in graphs can be defined in a recursive way:

$$H_j = f(\hat{A}H_{j-1}\Theta_j) \tag{14}$$

3.4 Overall Structure

The overall structure of our proposed model is illustrated in Figure 2.

Embedding Learning With the text corpus graph $\mathcal{G}_\mathcal{C}$ and diagnosis comorbidity graph $\mathcal{G}_\mathcal{T}$ constructed, we build a novel neural network upon the graph data which takes the related information of medical text corpus and diagnosis comorbidity into account. We explicitly model the word embeddings, document embeddings and diagnosis embeddings in a coupled way. One is for document and word embedding learning and the other is for diagnosis embedding learning.

For document and word embedding learning, we use a two graph convolution layers to model the text corpus graph. Though the text corpus graph does not model the document relations, the two layer graph convolution allow the message passing between documents. We define the adjacency matrix of the text corpus graph $\mathcal{G}_\mathcal{C}$ as $A^\mathcal{C}$. Then the word and document embeddings of one layer graph convolution can be obtained by:

$$H_0^\mathcal{C} = f(\hat{A}^\mathcal{C}X^\mathcal{C}\Theta_0^\mathcal{C}) \tag{15}$$

where $\hat{A}^\mathcal{C} = \hat{D}^{-\frac{1}{2}}A^\mathcal{C}\hat{D}^{-\frac{1}{2}}$ is the normalised matrix. Then the embeddings of words and documents after two graph convolution layers can be obtained by:

$$H_1^\mathcal{C} = f(\hat{A}^\mathcal{C}H_0^\mathcal{C}\Theta_1^\mathcal{C}) \tag{16}$$

where $H_1^\mathcal{C} \in R^{(|\mathcal{V}|+|\mathcal{C}|)\times k}$ and k is the hidden size.

For diagnosis embedding learning, we use a one layer graph convolution to get information from the neighbors. In this part, we can get the hidden representation of diagnoses by:

$$H_0^\mathcal{T} = f(\hat{A}^\mathcal{T}X^\mathcal{T}\Theta_0^\mathcal{T}) \tag{17}$$

where $\hat{A}^\mathcal{T}$ is the normalized matrix of $A^\mathcal{T}$ and $A^\mathcal{T}$ is the adjacency matrix of diagnosis comorbidity graph $\mathcal{G}_\mathcal{T}$.

In our model, we use Tanh function as the activation function and get the hidden representations of words, documents and diagnosis which hold the information from the graph structures to compute the similarities. The Tanh function can be defined as the followings:

$$f = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{18}$$

Match Score Computation From the above learning process, we get the hidden representations of words, documents and diagnoses. In this section, we will describe how to compute the similarities between the document representations and diagnosis representations.

It is clear that we can apply matrix dot to compute the scores. However, this can lead to inconsistency of the vector spaces between the embeddings of documents and diagnoses. In order to solve the above issue, we simply apply a hyperplane projection over the diagnosis space by using an orthogonal matrix and transfer the diagnosis representations into the related hyperplane of the document embeddings. In formal, the projection process can be defined as the followings:

$$\hat{H}^{\mathcal{T}} = H_0^{\mathcal{T}} - w_p^T H_0^{\mathcal{T}} w_p \tag{19}$$

where w_p is a matrix related to the relavance hyperplane of the document embeddings. Then, the hidden representations $\hat{H}^{\mathcal{T}}$ contain the diagnosis information and the hidden state $H^{\mathcal{C}}$ contain the embeddings of words and documents. Therefore, we use a matrix dot operation to compute the score between the document $i \in \mathcal{C}$ and the diagnosis $t \in \mathcal{T}$:

$$s(t, i) = H_i^{\mathcal{C}} \cdot (\hat{H}_t^{\mathcal{T}})^T \tag{20}$$

where $H_i^{\mathcal{C}}$ indicates the embeddings of the document i and $\hat{H}_t^{\mathcal{T}}$ is the embedding of diagnosis t.

Then, we can compute the probability of document $i \in \mathcal{C}$ is diagnosed with the diagnosis $j \in \mathcal{T}$ given the disease comorbidity graph $\mathcal{G}_{\mathcal{T}}$ and text corpus graph $\mathcal{G}_{\mathcal{C}}$ via a sigmoid function.

$$p(1 \mid i, j, \mathcal{G}_{\mathcal{T}}, \mathcal{G}_{\mathcal{C}}) = \frac{1}{1 + e^{-s(j,i)}} \tag{21}$$

where $s(j \mid i)$ is defined in Equation 20.

Loss Function With the probabilities of diagnoses given the documents learned, we use a multilabel binary cross-entropy loss [22].

$$\begin{aligned} \mathcal{L} = &- \frac{1}{N} \sum_{i,j} y_{ij} log(p(1 \mid i, j, \mathcal{G}_{\mathcal{T}}, \mathcal{G}_{\mathcal{C}})) \\ &- \frac{1}{N} \sum_{i,j} (1 - y_{ij}) log(p(0 \mid i, j, \mathcal{G}_{\mathcal{T}}, \mathcal{G}_{\mathcal{C}})) \end{aligned} \tag{22}$$

where $p(1 \mid i, j, \mathcal{G}_{\mathcal{T}}, \mathcal{G}_{\mathcal{C}})$ is defined in Equation 21 and y_i is an indicator vector which records the real diagnoses of the patient i.

4 Experiments

4.1 Dataset

In this paper, we use two datasets in our final experiments. As illustrated in [24], the distribution of diagnoses in the dataset has a very long tail and some

diagnoses only lie in few documents. Therefore, we use the 50 most common labels in our final datasets.

We extract the medical notes with the most recent records of patients from the MIMIC III dataset [12] and use two subcategories of the medical documents in the dataset: one is from the discharge summary and the other is from the nursing. And we only use the most recent records of the patients and we remove the documents whose diagnoses is not among the most 50 common diagnoses to get the final datasets. Table 1 gives the basic information of the datasets used in the paper.

Table 1. Basic Description of the Nursing dataset

Dataset	No. Samples	Average length
Nursing#50	6900	131.7
Dischargesummary# 50	38238	763.7

Data Processing For data prepossessing, we apply the traditional text data processing procedure. Vocabularies for each medical document are generated by first tokenizing the free text and lemmatizing the words in the texts and then removing stop words. In order to find the most representative words in the text corpus, we further analyze the tokenized text corpus and restrict the vocabulary size to 10000 for the Dischargesummary dataset and 5000 for the Nursing dataset.

Model Configuration We use Adam [15] to train our CGCN model with the learning rate set to 0.02. We randomly split the dataset into the training dataset, validation dataset and test dataset with the ratio 8:1:1. We find the best score on the validation dataset and report the performance on the test dataset. Our model is trained on a Ubuntu Server(16.04) with a Titan XP GPU(12 G) and we use pytorch to implement our model.

4.2 Models

In this section, we introduce the methods we used for comparison as the followings:

- **MLP** We use the average of word embeddings as the document representation and feed them into a two layer linear layer with Tanh as the activation. The embeddings of words are learned during the training.
- **CNN** Convolutional Neural Network. We use 50 filters with kernel size set to 2 and 3 to capture bigram information and trigram information of text sequences and then the feature map is fed into a max pooling layer to extract the most important information of a document. In our results, CNN_2 is the

Convolutional Neural Networks with the kernel size set to 2 and CNN_3 is the Convolutional Neural Networks whose kernel size is 3. For the multilabel settings, we use the sigmoid as the final activation function.

- **HA_GRU** Hierarchical attention GRU. It is first introduced in [33] which first encodes sentence to a sentence representation and then use sentence representations to learn a document representation with a hierarchical attention mechanism.
- **KV** Key Value Memory Network. The model is proposed in [21] and keeps a long term memory which can be retrieved in multiple hops. In our experiments, we use 3-hop Key Value Memory(KV_3) and 5-hop Key Value Memory Networks(KV_5).
- **C_MM** Condensed Memory Networks. The model is proposed in [24] and uses a condensed way for compressing the information learned from the long term memory. And in our experiments, we use 3-hop Condensed Memory Network(C_MM_3) and 5-hop Condensed Memory Network(C_MM_5).
- **LEAM** The model is first introduced in [29] and use label-wise attention to jointly learn the word embeddings and label embeddings in the multiclass text classification and the label descriptions are used in the classification task. For the clinical diagnosis inference problem, we change the last layer of activation function to the sigmoid function.
- **Text GCN** The model is first addressed in [34] and explores power of graph neural networks in the multiclass text classification. We use the same settings as in [34] and change the activation of the predictive layer to the sigmoid function.
- **CGCN-base** The model is a simple version of our CGCN where we remove the hyperplane projection part of the proposed CGCN.

4.3 Test Performance

Metrics In the experimental part, we report auc(macro), presicion@5(pre@5), recall@5(rec@5) and f1_score@5(f1@5) in our final results. For real applications, precision@5 indicates the ratio of relevance labels in the top 5 predictions and the recall@5 indicates the ratio of relevance labels of real diagnoses. Then the f1_score@5 can be computed by:

$$f1@5 = \frac{2 * pre@5 * rec@5}{pre@5 + rec@5} \tag{23}$$

Results In both of the two datasets, our proposed CGCN outperforms the baselines proposed in Table 2 and Table 3. The Text GCN performs the best among the baselines while our CGCN outperforms the Text GCN about 1.5% higher on Nursing#50 dataset and 1% higher than the Text CGN on Discharge-summary#50 dataset. And for f_score@5, our CGCN still outperforms the Text GCN about 1% higher on both datasets.

In order to illustrate the predictive power of our model with less train data, we randomly sample different ratios of the train data on the Nursing#50 dataset for

Table 2. Results on Nursing #50

methods	auc(macro)	rec@5	pre@5	f1@5
MLP	0.6734	0.3580	0.3380	0.3477
CNN_2	0.6729	0.3587	0.3467	0.3526
CNN_3	0.6642	0.3617	0.3455	0.3534
HA_GRU	0.6250	0.3365	0.3244	0.3303
KV_3	0.6729	0.3587	0.3528	0.3557
KV_5	0.6731	0.3594	0.3536	0.3568
C_MM_3	0.6590	0.3599	0.3412	0.3503
C_MM_5	0.6598	0.3504	0.3397	0.3450
LEAM	0.6788	0.3544	0.3344	0.3441
Text GCN	0.6842	0.3790	0.3579	0.3681
CGCN-base	0.6875	0.3731	0.3513	0.3619
CGCN	**0.6982**	**0.3909**	**0.3684**	**0.3793**

Table 3. Results on Dischargesummary #50

methods	auc(macro)	rec@5	pre@5	f1@5
MLP	0.8263	0.5394	0.4219	0.4729
CNN_2	0.8074	0.5362	0.4204	0.4713
CNN_3	0.8142	0.5446	0.4287	0.4798
HA_GRU	0.8154	0.5299	0.4212	0.4693
KV_3	0.8171	0.5126	0.4034	0.4515
KV_5	0.8164	0.5123	0.4025	0.4508
C_MM_3	0.8259	0.5359	0.4184	0.4693
C_MM_5	0.8258	0.5330	0.4191	0.4692
LEAM	0.8377	0.5806	0.4517	0.5081
Text GCN	0.8634	0.5980	0.4639	0.5225
CGCN-base	0.8718	0.6000	0.4696	0.5267
CGCN	**0.8730**	**0.6023**	**0.4726**	**0.5296**

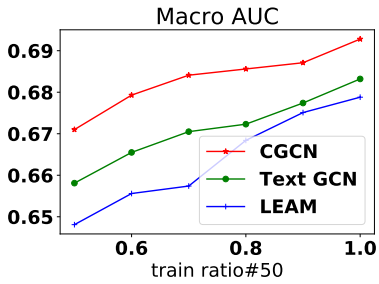

Fig. 3. Performance on varying ratio of training data

training and test on the whole test samples. From Figure 3, among the baselines, Text GCN outperforms the other baselines. What is more, even with the 60% of the training data for training, the performance of our CGCN is still competitive.

4.4 Effect of graph convolution layers

In this section, in order to figure out the effect of graph convolution layers, we do ablation study on our model by varying the number of the graph convolution operation both on the disease level and the text corpus level. Since there are coupled graph in our proposed model, we vary the number of graph convolution layers on one graph while keeping the same structure as our CGCN on the other graph. And then, we conduct experiments on the Nursing dataset and report the auc(macro) as the performance in Figure 4.

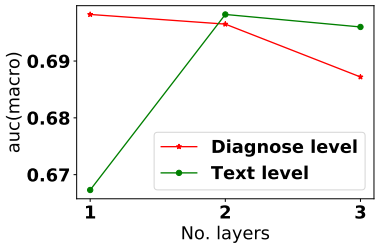

Fig. 4. Performance by varying the number of graph convolutions.

In Figure 4, we can conclude that the performance(auc(macro)) dose not benefit when adding the number of graph convolutions. As more graph convolution layers are appended on the diagnosis level, the embeddings of diagnoses may not be distinctive, causing the performance down. Therefore, we only use one layer of graph convolution on the diagnose level. Besides, we vary the number of graph convolutions on the text level while setting the number of the graph convolution on the diagnose level to 1. From the Figure 4, with the growing number of graph convolution layers on the text level, the performance achieve the best when the number of graph convolutions is set to 2 and adding more layers dose not gain any benefits. Since the text level graph is a heterogeneous graph where the upper node is the document node and the bottom node is the word node, one layer of graph convolution cannot transfer the information among the documents. With more layers of graph convolution, more information from high order neighbors are collected to the center document node.

4.5 Discussion

In this section, we discuss the performance of CGCN on the inductive setting. This is especially useful when we need to perform classification on a new patient. To this end, we develop the inductive learning algorithm for CGCN described in algorithm 2.

In the training phase, we construct the training graph containing the training dataset, validation dataset and train the CGCN in the transductive mode. In the

Algorithm 1: GraphAugmentation

1: **procedure** GRAPHAUGMENTATION(d,X_G,$\mathcal{G}$)
2: $V \leftarrow \mathcal{G}.V, E \leftarrow \mathcal{G}.E$;
3: Compute set $m = \{(i, j, TFIDF(i, j))|i \in \mathcal{V}, j = d||i = d, j \in \mathcal{V}\}$;
4: Compute $\mathcal{G}_{new} = (V \cup \{d\}, E \cup m)$;
5: Append the representation of d to X_G;
6: **return** $X_G, \mathcal{G}_{new}$
7: **end procedure**

Algorithm 2: Inductive learning For CGCN

1: Construct the text corpus $\mathcal{G}_{train}$ on training and validation dataset;
2: Train the CGCN with node features X_{train} and text corpus graph $\mathcal{G}_{train}$;
3: For a new patient d, $X_{test}, \mathcal{G}_{test} \leftarrow GraphAugmentation(d, X_{train}, \mathcal{G}_{train})$;
4: Feed the $X_{test}, \mathcal{G}_{test}$ to the CGCN

test phase, for each new patient, we generate a graph generated by Algorithm 1 and feed the graph to the CGCN without re-training. We use the TF-IDF metric as the document node features and the one-hot representations as the vocabulary node features. The results are listed in Table 4:

Cost Analysis The cost of applying new patients to the CGCN can be divided into two parts. One comes from the graph augmentation, and the other comes from the procedure of forward stage. With the given feature of a new patient (e.g. TF-IDF metric), we do not need to re-compute the word graph so that the cost of graph augmentation depends on the procedure of adding a new test node and linking the node to the current word graph (see Algorithm 1(3-4)). The process of appending a new node depends on the number of words in the test node. In the graph propagation stage, we use the same parameters learned in the training stage. As such, we can reduce the cost of dealing with a new patient.

Table 4. Performance on inductive settings

methods	Nursing #50	Dischargesummary #50
Text GCN	0.6747	0.8402
CGCN	0.6961	0.8562

From Table 4, we can conclude that, under the inductive setting, our model still outperforms the baselines (e.g., LEAM with 0.6788 on Nursing#50 dataset and Text GCN with 0.8402 on Dischargesummary#50 dataset) and suggests that the new patient can learn some knowledge form the existing patients in the training dataset.

5 Conclusion

In this paper, we propose a novel coupled graph neural network (CGCN) in the clinical diagnosis inference. We emphasize the challenges of the text based clinical diagnosis inference and take the information sharing and disease comorbidity into the consideration and use a hyperplane projection method to project the diagnosis embeddings into the related hyperplane of the patient embeddings. The CGCN outperforms the baselines on several datasets. We have shown that the information sharing and disease comorbidity are essential in the clinical diagnosis inference and need to be further studied in the computer aided diagnosis inference systems.

Acknowledge This work was supported in part by National Natural Science Foundation of China under Grant No. 61532010 and 61521002, and Beijing Academy of Artificial Intelligence (BAAI).

References

1. Bastings, J., Titov, I., Aziz, W., Marcheggiani, D., Sima'an, K.: Graph convolutional encoders for syntax-aware neural machine translation. arXiv (2017)
2. 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 (2018)
3. Baxt, W.G.: Use of an artificial neural network for data analysis in clinical decision-making: the diagnosis of acute coronary occlusion. Neural computation **2**(4), 480–489 (1990)
4. Binaghi, E.: A fuzzy logic inference model for a rule-based system in medical diagnosis. Expert Systems **7**(3), 134–141 (1990)
5. Bruna, J., Zaremba, W., Szlam, A., Lecun, Y.: Spectral networks and locally connected networks on graphs (2014)
6. Choi, E., Bahadori, M.T., Schuetz, A., Stewart, W.F., Sun, J.: Doctor ai: Predicting clinical events via recurrent neural networks. In: MLHC. pp. 301–318 (2016)
7. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. In: NIPS. pp. 3844–3852 (2016)
8. Djulbegovic, B., Guyatt, G.H.: Progress in evidence-based medicine: a quarter century on. The Lancet **390**(10092), 415–423 (2017)
9. Ghassemi, M., Naumann, T., Doshi-Velez, F., Brimmer, N., Joshi, R., Rumshisky, A., Szolovits, P.: Unfolding physiological state: Mortality modelling in intensive care units. In: SIGKDD. pp. 75–84. ACM (2014)
10. Henaff, M., Bruna, J., Lecun, Y.: Deep convolutional networks on graph-structured data. arXiv (2015)
11. Iyyer, M., Enns, P., Boyd-Graber, J., Resnik, P.: Political ideology detection using recursive neural networks. In: ACL. vol. 1, pp. 1113–1122 (2014)
12. Johnson, A.E., Pollard, T.J., Shen, L., Li-wei, H.L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L.A., Mark, R.G.: Mimic-iii, a freely accessible critical care database. Scientific data **3**, 160035 (2016)
13. Keezer, M.R., Sisodiya, S.M., Sander, J.W.: Comorbidities of epilepsy: current concepts and future perspectives. The Lancet Neurology **15**(1), 106–115 (2016)

14. Kim, Y.: Convolutional neural networks for sentence classification. arXiv (2014)
15. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv (2014)
16. Kipf, T., Welling, M.: Semi-supervised classification with graph convolutional networks (2017)
17. Lipton, Z.C., Kale, D.C., Elkan, C., Wetzel, R.: Learning to Diagnose with LSTM Recurrent Neural Networks. arXiv (2015)
18. Liu, N., Lu, P., Zhang, W., Wang, J.: Knowledge-aware deep dual networks for text-based mortality prediction. In: ICDE. pp. 1406–1417. IEEE (2019)
19. Lund, N., Petersen, A., Snoer, A., Jensen, R.H., Barloese, M.: Cluster headache is associated with unhealthy lifestyle and lifestyle-related comorbid diseases: Results from the danish cluster headache survey. Cephalalgia **39**(2), 254–263 (2019)
20. Manevitz, L.M., Yousef, M.: One-class svms for document classification. JMLR **2**(Dec), 139–154 (2001)
21. Miller, A., Fisch, A., Dodge, J., Karimi, A.H., Bordes, A., Weston, J.: Key-value memory networks for directly reading documents. arXiv (2016)
22. Nam, J., Kim, J., Loza Mencía, E., Gurevych, I., Fürnkranz, J.: Large-scale Multi-label Text Classification - Revisiting Neural Networks. arXiv (2013)
23. Peng, H., Li, J., He, Y., Liu, Y., Bao, M., Wang, L., Song, Y., Yang, Q.: Large-scale hierarchical text classification with recursively regularized deep graph-cnn. In: WWW. pp. 1063–1072 (2018)
24. Prakash, A., Zhao, S., Hasan, S.A., Datla, V., Lee, K., Qadir, A., Liu, J., Farri, O.: Condensed memory networks for clinical diagnostic inferencing. In: AAAI (2017)
25. Ramos, A.L., Redeker, I., Hoffmann, F., Callhoff, J., Zink, A., Albrecht, K.: Comorbidities in patients with rheumatoid arthritis and their association with patient-reported outcomes: results of claims data linked to questionnaire survey. The Journal of rheumatology **46**(6), 564–571 (2019)
26. Sackett, D.L., Rosenberg, W.M., Gray, J.M., Haynes, R.B., Richardson, W.S.: Evidence based medicine: what it is and what it isn't (1996)
27. Schlichtkrull, M.S., Kipf, T., Bloem, P., Den Berg, R.V., Titov, I., Welling, M.: Modeling relational data with graph convolutional networks pp. 593–607 (2018)
28. Sun, M., Zhao, S., Gilvary, C., Elemento, O., Zhou, J., Wang, F.: Graph convolutional networks for computational drug development and discovery. Briefings in Bioinformatics (2019)
29. 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. arXiv (2018)
30. Wang, J., Wang, Z., Zhang, D., Yan, J.: Combining knowledge with deep convolutional neural networks for short text classification. In: IJCAI. pp. 2915–2921. AAAI Press (2017)
31. Wang, S., Manning, C.D.: Baselines and bigrams: Simple, good sentiment and topic classification. In: ACL. pp. 90–94. ACL (2012)
32. Xue, Q., Zhang, W., Zha, H.: Multi-label classification of patient notes: case study on icd code assignment. In: AAAI (2020)
33. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E.: Hierarchical attention networks for document classification. In: NAACL. pp. 1480–1489 (2016)
34. Yao, L., Mao, C., Luo, Y.: Graph convolutional networks for text classification. In: AAAI. vol. 33, pp. 7370–7377 (2019)
35. Zhang, H., Xiao, L., Chen, W., Wang, Y., Jin, Y.: Multi-task label embedding for text classification. arXiv (2017)
36. Zhou, J., Cui, G., Zhang, Z., Yang, C., Liu, Z., Sun, M.: Graph neural networks: A review of methods and applications. arXiv (2018)