Inheritance-Guided Hierarchical Assignment for Clinical Automatic Diagnosis

Yichao Du[1], Pengfei Luo[1], Xudong Hong[2], Tong Xu[1(✉)], Zhe Zhang[1], Chao Ren[1], Yi Zheng[3], and Enhong Chen[1]

[1] School of Computer Science and Technology, University of Science and Technology of China, Hefei, China
{duyichao,pfluo,renchao,pb142090}@mail.ustc.edu.cn, {tongxu,cheneh}@ustc.edu.cn
[2] Institution of Smart City Research (WuHu), University of Science and Technology of China, Wuhu, China
xdhong@ahut.edu.cn
[3] HUAWEI Technologies, Hangzhou, China
zhengyi29@huawei.com

Abstract. Clinical diagnosis, which aims to assign diagnosis codes for a patient based on the clinical note, plays an essential role in clinical decision-making. Considering that manual diagnosis could be error-prone and time-consuming, many intelligent approaches based on clinical text mining have been proposed to perform automatic diagnosis. However, these methods may not achieve satisfactory results due to the following challenges. First, most of the diagnosis codes are rare, and the distribution is extremely unbalanced. Second, existing methods are challenging to capture the correlation between diagnosis codes. Third, the lengthy clinical note leads to the excessive dispersion of key information related to codes. To tackle these challenges, we propose a novel framework to combine the inheritance-guided hierarchical assignment and co-occurrence graph propagation for clinical automatic diagnosis. Specifically, we propose a hierarchical joint prediction strategy to address the challenge of unbalanced codes distribution. Then, we utilize graph convolutional neural networks to obtain the correlation and semantic representations of medical ontology. Furthermore, we introduce multi attention mechanisms to extract crucial information. Finally, extensive experiments on MIMIC-III dataset clearly validate the effectiveness of our method.

Keywords: Clinical automatic diagnosis · Hierarchical assignment · Co-occurrence graph · Graph Convolutional Network

1 Introduction

The clinical note is an essential part of Electronic Health Record (EHR), which contains lengthy and terminological text records about medical history, chief

C. S. Jensen et al. (Eds.): DASFAA 2021, LNCS 12683, pp. 461–477, 2021.
https://doi.org/10.1007/978-3-030-73200-4_31

complaint, current symptoms, and laboratory test results. To avoid the redundancy and ambiguity caused by the text, the World Health Organization recommends using the diagnosis codes in the International Classification of Diseases (ICD) for each disease, symptom, and sign to represent the patient's condition. The goal of clinical diagnosis is to assign the most likely diagnosis codes for the patient based on the clinical note. Traditionally, clinical diagnosis is completed by well-trained clinical coders, which is labor-intensive and error-prone because the diagnosis codes system is vast and growing. For example, in the United States, about 20% of patients are misdiagnosed at the primary care level, and one-third of the misdiagnosis will cause later severe injury to the patients [22].

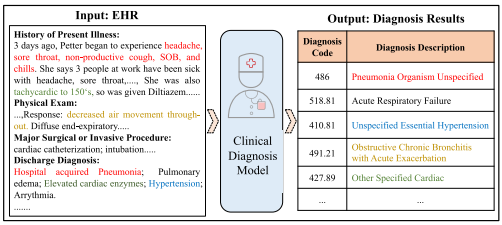

Fig. 1. Illustration of clinical automatic diagnosis task. The input and output of the model are EHR and diagnosis codes, respectively. The text related to the diagnosis code in the EHR is marked in colored font.

Consequently, the automatic clinical diagnosis based on EHR has aroused widespread attention in the industrial and academic circles [4]. Among the proposed methods, supervised machine learning methods were trained to learn shallow feature combinations for clinical note [7,19]. Recently, most deep learning models treated this task as a sequence learning problem, including used Convolutional Neural Networks [9,16] and Recurrent Neural Networks [3,21] to capture complex semantic information. On this basis, medical ontology was further introduced as auxiliary knowledge. Specifically, Bai et al. [1] incorporated the disease encyclopedia of Wikipedia into the model to enhance its predictive ability. Besides, the patient's history and demographic information could also be leveraged to enhance the prediction of future admissions [1,14,20]. Although these methods have made significant progress in automatic diagnosis, they may also fail due to the following challenges:

- **C1: The number of diagnosis codes is enormous, and the distribution is extremely unbalanced.** For example, the MIMIC-III [6] dataset, which is widely used for automatic diagnosis, contains 8,925 codes, but 4,344 appear less than five times in all data. The severe long-tail distribution makes it difficult to assign proper codes to rare diseases, which may cause irreparable damage to the patients.
- **C2: The correlations between diagnosis codes are greatly overlooked.** However, the medical relationship between diseases can help us identify diseases that are not clearly reflected by the clinical note. As shown in

Fig. 1, we can extract clues (colored fonts) from the text to assign diagnosis codes to the patient. For example, from the text "Hospital Acquired Pneumonia", we can easily infer the code "486 (Pneumonia Organism Unspecified)". Nevertheless, it is difficult to infer the code "410.81 (Acute Respiratory Failure)" only from the text. Fortunately, we can infer the code "410.81" from the relationship between it and the code "486", that is, "Pneumonia Organism Unspecified" will in all probability cause patients to have the symptom of "Acute Respiratory Failure".

– **C3: In clinical note, only a few key fragments can provide valuable information for automatic diagnosis.** For example, in the MIMIC-III dataset, clinical notes usually contain more than 1,500 tokens, but only a few tokens are related to specific diagnosis codes. Extracting crucial tokens for specific diagnosis codes is as tricky as finding a needle in a haystack.

To this end, we propose a model named **I**nheritance-guided **H**ierarchical Assignment with **C**o-occurrence-based **E**nhancement (IHCE) to address these challenges. First, for C1, we design a hierarchical assignment method based on the hierarchical inheritance structure of diagnosis codes defined by ICD, which makes assignment level by level. As shown in Fig. 2, "405.0 (Malignant renovascular hypertension)" and "405.1 (Benign secondary hypertension)" are mutually exclusive. Moreover, "405.01 (Malignant renovascular hypertension)" inherits the information of "405.0". Consequently, if we assign "405.0" at the high level, we will tend to further assign "405.01" instead of the children of "405.1". With the inheritance-guided hierarchical assignment, we can use the diagnostic results of a high level to guide the low level, which addresses the challenge of unbalanced distribution. Second, for C2, we construct a co-occurrence graph based on EHR data and use GCN to obtain the diagnosis codes' semantic representations. In this way, the representations of the diagnosis codes contain the correlation between diseases, which help us to assign codes to diseases for where it is challenging to find textual clues from the clinical note. Third, for C3, we enhance the ability to extract the tokens related to the diagnosis codes based on the attention mechanism which models the interaction between diagnosis codes' ontology representations and the clinical note. Finally, experiments on a real medical dataset show that IHCE is superior to the SOTA methods on all evaluation metrics.

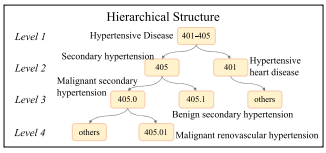

Fig. 2. An example of diagnosis codes' descriptors and their hierarchical inheritance structure based on ICD.

2 Related Work

2.1 Clinical Automatic Diagnosis

Clinical automatic diagnosis has become a research hot spot in medicine, aiming to solve manual diagnosis limitations. In recent years, deep learning technologies [9,16,21] have shown substantial advantages over traditional machine learning methods [7,19] and have been widely used for this task. Most researchers modeled this task as a multi-label text classification task based on the free text in EHR. Among them, Shi et al. [21] proposed a character-perceived LSTM network that generated written diagnosis descriptions and representations of diagnosis codes. Baumel et al. [3] proposed a hierarchical-GRU with a label-dependent attention layer to alleviate excessive text problem. Wang et al. [23] proposed a label-word joint embedding model and applied the cosine similarity to assign the codes. Moreover, some researchers incorporated external knowledge into the model [1,14,20]. For example, Knowledge Source Integration (KSI) [1] calculated the matching score between the clinical note and each knowledge document based on the intersection of clinical notes and external knowledge for this task. Our method is different from these methods, considering the hierarchy and co-occurrence relationship to achieve better performance in automatic diagnosis.

2.2 Graph Convolutional Network

In the past few years, Graph Convolutional Network (GCN) [8] has been widely used in various tasks to encode advanced graph structures, such as healthcare [11, 25], recommender systems [12], business analysis [10], machine translation [2], text classification [18,24]. Specifically, in order to promote the sharing of disease among patients, Liu et al. [11] applied GCN on text corpus to collect high-order neighbor information, and predicted for patients based on projection. Yao et al. [24] proposed Text-GCN, which was utilized to learn the representations of words and documents to improve text classification. Peng et al. [18] proposed a recursive regularized GCN to perform large-scale text classification on word co-occurrence graphs. Inspired by this, we apply GCN to obtain a good correlation between diagnosis codes and represent the medical ontology. Furthermore, we utilize the ontology representations as interactive information to improve the performance of automatic diagnosis.

3 Preliminaries

For a patient, the word sequence $S = \{w_1, w_2, ..., w_n\}$ of the patient's clinical note is included, where n is the length of S. Furthermore, a set of diagnosis codes $L = \{l_1, l_2, ..., l_{|L|}\} \in \{0,1\}^{|L|}$ are also contained to denote the diseases of the patient, where $|L|$ is the number of diagnosis codes. In addition, we also introduce hierarchical inheritance structure $\mathcal{L} = \{L^1, L^2, ..., L^{\mathcal{T}}\}$ to expand L based on external knowledge (i.e., the hierarchical inheritance structure based

on ICD in Fig. 2), where $L^t = \left\{l_1^t, l_2^t, ..., l_{|L^t|}^t\right\}$ means all diagnosis codes of the level-t, and $\mathcal{T}$ is the total number of hierarchical levels. Note that, $L^{\mathcal{T}} = L$, which means that the last hierarchical level is the same as the patient's diagnosis codes. With above description, we can define the clinical automatic diagnosis task with inheritance guidance as follows:

Definition 1. *Given the patient's clinical note sequence S and the diagnosis codes hierarchical inheritance structure $\mathcal{L}$, our goal is to predict the patient's diagnosis codes set $\hat{L}^t = \left\{\hat{l}_1^t, \hat{l}_2^t, ...\right\} \in \{0,1\}^{|\hat{L}^t|}$ level by level, and finally use the last level $\hat{L^{\mathcal{T}}}$ as the prediction of the patient's diagnosis.*

4 The Proposed Model IHCE

As shown in Fig. 3, IHCE mainly contains three components: (1) Document Encoding Layer (DEL), (2) Ontology Representation Layer (ORL), and (3) Hierarchical Prediction Layer (HPL). Specifically, we first utilize the DEL to obtain representations of the clinical note and diagnosis codes. Secondly, we apply the ORL to obtain the correlation and semantic representations of medical ontology. Finally, we design HPL to predict the patient's diagnosis codes based on hierarchical dependence and attention mechanism.

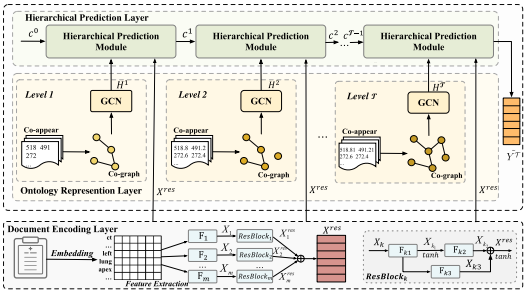

Fig. 3. The architecture of IHCE.

4.1 Document Encoding Layer

The goal of DEL is to generate unified representations for the clinical note and diagnosis codes. We first utilize the Embedding Module to encode the patient's clinical note and diagnosis codes. Then, we apply the Feature Extraction Module to enhance the semantic representation of the clinical note.

Embedding Module. First, given the word sequence $S = \{w_1, w_2, ..., w_n\}$, we use the word vector matrix $E = \left[e_1, e_2, \ldots, e_{|E|}\right] \in \mathbb{R}^{|E| \times d_e}$ to obtain the word embedding sequence $X = [x_1, x_2, \ldots, x_n] \in \mathbb{R}^{n \times d_e}$, where $|E|$ is the size of the vocabulary, and d_e is the dimension of the word vector. Similarly, we generate the diagnosis code ontology embedding for each code $l_i^t \in L^t$ via averaging the word embedding of its descriptor sequence:

$$\begin{aligned} v_i^t &= \frac{1}{|N_i^t|} \sum_{j \in N_i^t} e_j, \quad i = 1, \ldots, |L^t| \\ V^t &= \left[v_1^t, v_2^t, \ldots, v_{|L^t|}^t\right] \in \mathbb{R}^{|L^t| \times d_e} \end{aligned}, \tag{1}$$

where N_i^t is the text descriptor index set of l_i^t, and v_i^t denotes the word embedding of the l_i^t, and V^t indicates the representations of all codes of the level-t.

Feature Extraction Module. As shown in the lower part of the Fig. 3, we apply the multi-filter residual convolutional neural network [9] architecture for deep feature extraction on clinical note's embedding matrix X.

First, we utilize convolutional neural networks containing m filters to capture different length patterns of word sequence:

$$\begin{aligned} X_1 &= \mathrm{F}_1\left(X, W_1\right) = \tanh\left[\ldots, W_1^T X^{j:j+s_1-1}, \ldots\right] \\ &\ldots \\ X_m &= \mathrm{F}_m\left(X, W_m\right) = \tanh\left[\ldots, W_m^T X^{j:j+s_m-1}, \ldots\right] \end{aligned}, \text{where } j = 1, 2, ..., n, \tag{2}$$

Let us take the k-th operation as an example. $\mathrm{F}_k\left(X, W_k\right)$ denotes the convolution operation on the matrix X, where $W_k \in \mathbb{R}^{(s_k \times d_e) \times d_c}$ is the parameter matrix, and d_c indicates each convolutional layer's feature mapping dimension. $s_1, s_2, ..., s_m$ denote different convolution kernel sizes, and $X^{j:j+s_k-1} \in \mathbb{R}^{s_k \times d_e}$ is the input matrix of the j-th to the $(j + s_k - 1)$-th rows in X. Note that, we set padding and stride as $floor(s_k/2)$ and 1. Finally, the feature matrices $X_k \in \mathbb{R}^{n \times d_c}, k = 1, 2, ..., m$ can be obtained. In order to express conciseness, the bias is ignored in all the calculation formulas in this paper.

Next, we connect m parallel residual blocks after the multi-filter convolutional layer, capturing longer text features by expanding the receptive field. Taking the k-th unit as an example, the residual block is formally defined as:

$$\begin{aligned} X_{k_1} &= \mathrm{F}_{k_1}\left(X_k, W_{k_1}\right) = \tanh\left[\ldots, W_{k_1}^T X_k^{j:j+s_k-1}, \ldots\right], \\ X_{k_2} &= \mathrm{F}_{k_2}\left(X_{k_1}, W_{k_2}\right) = \left[\ldots, W_{k_2}^T X_{k_1}^{j:j+s_k-1}, \ldots,\right], \\ X_{k_3} &= \mathrm{F}_{k_3}\left(X_k, W_{k_3}\right) = \left[\ldots, W_{k_3}^T X_k^{j:j}, \ldots\right], \\ X_k^{res} &= \tanh\left(X_{k_2} + X_{k_3}\right), \end{aligned} \tag{3}$$

where $j = 1, 2, ..., n$, and W_{k_i} is the weight matrix of the k_i-th convolution layer in the residual block, specifically $W_{k_1} \in \mathbb{R}^{(s_k \times d_c) \times d_r}, W_{k_2} \in \mathbb{R}^{(s_k \times d_r) \times d_r}, W_{k_3} \in \mathbb{R}^{(1 \times d_c) \times d_r}$. The output of each residual block is $X_k^{res}, k = 1, 2, ..., m$, where d_r indicates the feature mapping dimension. Finally, we concatenate them together by rows to obtain an enhanced clinical note's representation:

$$X^{res} = \mathrm{concat}\left(X_1^{res}, \ldots, X_m^{res}\right) \in \mathbb{R}^{n \times d_{res}}, \text{where } d_{res} = (m \times d_r). \tag{4}$$

4.2 Ontology Representation Layer

Comorbidities and complications manifest the correlation between the diagnosis codes ontology and play an auxiliary role for codes that are difficult to predict based on the clinical note alone. To this end, we first use co-occurrence features at each hierarchical level to construct a co-occurrence graph (co-graph) of diagnosis codes ontology. Then, we use GCN to capture the ontology's representations, which contain the correlation between the ontology. Here we take the level-t as an example to introduce the process.

Co-graph Construction. The co-graph is represented by $G^t = (L^t, E^t)$, where L^t and E^t indicate the diagnosis codes set and edge set of the level-t, respectively. For any diagnosis code l_i^t, if there is another code l_j^t in the EHR data that co-appears, there is an edge $e(l_i^t, l_j^t)$ between them. And the corresponding weight is calculated as follows:

$$e(l_i^t, l_j^t) = \frac{\text{count}(l_i^t, l_j^t)}{\sum_{l_k^t \in L^t} \text{count}(l_i^t, l_k^t)}, \tag{5}$$

where $\text{count}(\cdot, \cdot)$ indicates the number of times the two codes co-appear in the whole EHR dataset, which can represent prior knowledge. After that, the edge set E^t can be described as follows:

$$E^t = \{e(l_i^t, l_j^t) \mid l_i^t, l_j^t \in L^t\}. \tag{6}$$

Co-graph Propagation via GCN. Now we turn to represent the diagnosis codes. First, we can obtain the feature matrix $H^{t,(0)} = V^t \in \mathbb{R}^{|L^t| \times d_e}$ of the diagnosis codes ontology by Eq. (1). For the sake of simplicity, we omit the superscript t in the rest of this subsection. Then, we apply the GCN to propagate the representations of the diagnosis codes on the co-graph G, which takes the feature matrix $H^{(l)}$ and the matrix $\tilde{A}$ as input, and update the embedding of the codes by utilizing the information of adjacent codes:

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right), \tag{7}$$

where $\tilde{A} = A + I$, A is the adjacency matrix of G, I is the identity matrix, $\tilde{D}_{ii} = \sum_i \tilde{A}_{ij}$, and $W^{(l)}$ is a layer-specific trainable weight matrix. $\sigma(\cdot)$ denotes an activation function, such as the $\text{ReLU}(\cdot) = \max(0, \cdot)$. $H^{(l)} \in \mathbb{R}^{L \times d_g}$ is the matrix of activations in the l-th layer, where d_g indicates the hidden layer size of GCN. Then the last hidden layer is used to represent the diagnosis codes ontology, i.e., $H^t = H^{t,(l+1)} \in \mathbb{R}^{|L^t| \times d_g}$.

4.3 Hierarchical Prediction Layer

To simulate human diagnosis's gradual progress from shallow to deep, we propose an inheritance-guided hierarchical joint learning mechanism. To be specific,

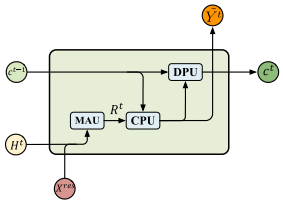

Fig. 4. Hierarchical prediction module.

according to the hierarchical structure of the codes, the patient is diagnosed progressively from coarse-grained to fine-grained.

Figure 4 shows the core module Hierarchical Prediction Module(HPM) of HPL. Specifically, HPM is mainly composed of three parts, namely Multi Attention Unit (MAU), Code Predicting Unit (CPU) and Dependency Passing Unit (DPU) respectively. For the level-t, the input of HPM includes three parts, i.e., the clinical note's representation X^{res}, the medical ontology representations H^t, and the dependency information c^{t-1} of the previous level:

$$\begin{aligned} R^t &= \text{MAU}\left(X^{res}, H^t\right), \\ Y^t &= \text{CPU}\left(c^{t-1}, R^t\right), \\ c^t &= \text{DPU}\left(c^{t-1}, \tilde{Y^t}\right). \end{aligned} \tag{8}$$

We first utilize the MAU part to obtain the correlation representation R^t between the clinical note and medical ontology. Next, the CPU part assigns the diagnosis codes $\tilde{Y^t}$ to the patient based on the R^t and c^{t-1}. Finally, the DPU part generates the level dependency information c^t for the next level based on the previous level's memory and the current level's assignment results. Note that we set c^0 to 0 since the current level is 0 and does not contain the previous level's information. Next, we introduce each unit of the HPM at level-t.

Multi Attention Unit. By the operations above, we can obtain the clinical note representation X^{res} and medical ontology representations H^t. Intuitively, the patient's clinical note is composed of a large number of lengthy text descriptions and different codes may focus on different aspects of the document. Therefore, for level-t, we need $|L^t|$ aspects to focus on different codes to represent the overall semantic of the whole clinical note. Next, we introduce the two attention mechanisms we use.

Ontology Guided Attention. For some diagnosis codes that are difficult to predict using only clinical text, we can improve it by interacting between the clinical note and medical ontology. First, we pass the document feature matrix X^{res} through a simple feed-forward neural network:

$$O'_t = \tanh(W'_t \cdot (X^{res})^T), \tag{9}$$

where $W_t' \in \mathbb{R}^{d_g \times d_{res}}$ is the transform matrix, d_g is consistent with the dimension of the columns of H^t, and $O_t' \in \mathbb{R}^{d_g \times n}$ is the intermediate result. Then, for each code $l^t \in L^t$, we can generate the attention vector guided by the ontology:

$$\alpha_{l^t} = \text{softmax}(h_{l^t} \cdot O_t'), \tag{10}$$

where $h_{l^t} \in H^t$ is the feature vector of label l^t, and softmax$(\cdot)$ is the normalized exponential function for row operations. The attention $\alpha_{l^t} \in \mathbb{R}^{1 \times n}$ is then used to compute vector representation for each label:

$$x_i^{att'} = \alpha_{l^t} \cdot X^{res}, \tag{11}$$

Finally, we concatenate the $x_i^{att'}(i = 1, .., |L^t|)$ to obtain the ontology guided document representation, denoted as $X_t^{att'} = [x_1^{att'}, x_2^{att'}, ..., x_{|L^t|}^{att'}] \in \mathbb{R}^{|L^t| \times d_{res}}$.

Code Specific Attention. Similar to ontology guided attention, the code specific attention is formalized as:

$$\begin{aligned} O_t'' &= \tanh(W_t'' \cdot (X^{res})^T), \\ A_t'' &= \text{softmax}(U_t'' \cdot O_t''), \\ X_t^{att''} &= A_t'' \cdot X^{res}, \end{aligned} \tag{12}$$

where $W_t'' \in \mathbb{R}^{d_a \times d_{res}}$ is the intermediate parameter matrix. d_a is a hyperparameter, $O_t'' \in \mathbb{R}^{d_a \times n}$ is the intermediate result matrix and $U_t'' \in \mathbb{R}^{|L^t| \times d_a}$ is the code-specific attention parameter matrix. Finally, $X_t^{att''} \in \mathbb{R}^{|L^t| \times d_{res}}$ denotes code-specific document representation.

With the above description, we apply $R^t = \text{concat}(X_t^{att'}, X_t^{att''}) \in \mathbb{R}^{|L^t| \times 2d_{res}}$ as the output of the MAU.

Code Predicting Unit. For the level-t, we combine the result R^t of MAU with the inherited information c^{t-1} of the previous level to assign diagnosis codes to the patient. Specifically, the CPU uses a linear layer following a sigmoid transformation for each code:

$$\begin{aligned} X_t^{cls} &= \text{concat}(\text{broadcast}(c^{t-1}), R^t), \\ \tilde{Y^t} &= \sigma\left(X_t^{cls} \cdot W_y^t\right), \end{aligned} \tag{13}$$

where broadcast$(\cdot)$ is the process of making matrixes with different shapes have compatible shapes for arithmetic operations, $\sigma(\cdot)$ denotes an activation function, such as the sigmoid$(x) = \frac{1}{1+e^{-x}}$, $W_y^t \in \mathbb{R}^{(2d_{res}+d_c^{t-1}) \times 1}$ is the parameter of the CPU, and $\tilde{Y^t} \in \mathbb{R}^{|L^t| \times 1}$ is the prediction results of the level-t.

Dependency Passing Unit. We aim to preserve important information while reducing the harm caused by the previous level's error transmission. Therefore, we employ the combination of a linear layer and sigmoid function to imitate the gating mechanism to filter and integrate information as follows:

$$Z = \text{concat}((\tilde{Y}^t)^T, c^{t-1}),$$
$$c^t = \sigma(Z \cdot W^t_{dpu}), \tag{14}$$

where $Z \in \mathbb{R}^{1\times(|L^t|+d_c^{t-1})}$ and $W^t_{dpu} \in \mathbb{R}^{(|L^t|+d_c^{t-1})\times d_c^t}$ is the parameter matrix. Then, we can get the inter-level dependence $c^t \in \mathbb{R}^{1\times d_c^t}$ based on the previous level's memory information and the prediction results of the current level.

4.4 Training

For training, we combine all levels of multi-label binary cross-entropy as the loss:

$$loss = \sum_t^{\mathcal{T}} loss^t = \sum_t^{\mathcal{T}} \sum_{i=1}^{L^t} \left[-y_i \log\left(\tilde{y}_i\right) - \left(1-y_i\right) \log\left(1-\tilde{y}_i\right)\right], \text{where } \tilde{y}_i \in \tilde{Y}^t, \tag{15}$$

where $loss_t$ indicates the loss function of level-t.

5 Experiments

5.1 Dataset and Evaluation Metrics

In this paper, we conduct experiments on a real-world dataset: the MIMIC-III dataset, which is widely used in clinical automatic diagnosis. Following previous studies [9,16], we use the discharge summaries as the model's input and use the full codes and the top 50 most common codes for experiments. Specifically, for the MIMIC-III full setting, it includes the 8,925 codes, 47,719, 1,631, and 3,372 discharge summaries used for training, validation, and testing, respectively. For the MIMIC-III top-50 setting, it includes 8,067, 1,574, and 1,730 discharge summaries used for training, validation, and testing, respectively. In addition, we expand the codes from fine to coarse according to the hierarchical inheritance structure of ICD because EHR data only have the finest-grained codes (i.e.the level-4 in Table 1). The specific statistical results are shown in Table 1.

The evaluation metrics used in the experiments are Precision@K (K = 5, 8, and 15), Macro-F1, Micro-F1, Macro-AUC and Micro-AUC.

5.2 Implementation Details

We utilize PyTorch [17] to implement IHCE model and train it on a server with 4 × V100 GPU. For the training setting, we use AdamW [13] for learning and set the learning rate and weight decay to 0.0001 and 0.00005, respectively. We set the dropout probability 0.4 and set the batch size to 16. We also apply an early stop mechanism, in which the training will stop if the Micro-F1 score on the validation set does not improve in 10 continuous epochs. Since our model has a number of hyperparameters, it is infeasible to search optimal values for

Table 1. The statistics of hierarchical levels.

Statistics	Full	Top-50
# codes in level-1	199	25
# codes in level-2	1,175	40
# codes in level-3	5,125	48
# codes in level-4	8,925	50
# avg codes per EHR in level-1	11.02	4.70
# avg codes per EHR in level-2	13.75	5.37
# avg codes per EHR in level-3	15.30	5.71
# avg codes per EHR in level-4	15.86	5.77

all hyperparameters. We keep the hyperparameters of the Feature Extraction Module consistent with Li [9]. Specifically, the word embedding dimension $d_e = 100$, the number of convolution kernels m in feature extraction is 6, and the size of the convolution kernels $s_1, s_2, ...s_m$ are set to "3, 5, 9, 15, 19, 25", $d_c = d_e$ and $d_r = 50$. Besides, we pre-train word embeddings on all the text in the training set using the word2vec [15] implemented by gensim[1]. The maximum length of a token sequence is 2,500, and the one that exceeds this length will be truncated. For the remaining parameters, we use the grid to search for the optimal hyperparameters. Specifically, we set the number of hidden layers to 1, and the hidden layer size $d_g = 300$ for GCN. In addition, we set d_a=300 for ORL's attention dimension, and $d_c^t = 500(t = 1, 2, ..., \mathcal{T} - 1)$ for all DPUs' parameters dimension.

5.3 Baselines

We compared IHCE with the following baselines, including machine learning and deep learning models:

- **LR:** which is a bag-of-words logistic regression model.
- **H-SVM** [19]**:** which designs a hierarchical SVM algorithm from root to leaf node by utilizing the hierarchical structure of diagnosis codes.
- **Bi-GRU** [16]**:** which employs bidirectional gated recurrent units to learn clinical note's representation for automatic diagnosis task.
- **C-MemNN** [20]**:** which combines the memory network with iterative compression memory representation to improve diagnosis accuracy.
- **C-LSTM-Att** [21]**:** which uses an LSTM-based language model to generate clinical note and diagnosis code representations as well as an attention mechanism to resolve the mismatch between notes and codes.
- **LEAM** [23]**:** which is proposed for text classification task by projecting labels and words in the same embedding space and using the cosine similarity to predict the label of text.

[1] https://radimrehurek.com/gensim/.

- **HARNNN** [5] which is initially used for multi-label text classification and considers the hierarchy of categories. We apply it to the automatic diagnosis.
- **CNN** [16]**:** which uses a single layer convolutional neural network and a max-pooling layer for automatic diagnosis task.
- **CAML and DR-CAML** [16]**:** which assign diagnosis codes based on clinical text by using CNN to aggregate information among the clinical note and attention mechanism to select the most relevant segment for each possible code. DR-CAML further uses text description as a regularization.
- **MultiResCNN** [9]**:** which utilizes multi-fliter convolutional neural networks and residual networks for automatic diagnosis and becomes the SOTA model on MIMIC-III.

5.4 Overall Performance

In this section, we compare the IHCE with existing works for clinical automatic diagnosis. Table 2 shows our overall performance on MIMIC-III full setting and MIMIC-III 50 setting. $\mathcal{T} = 3$ means that our experiment is based on the last three levels (i.e., level-2 to level-4 in Table 1) in the hierarchy. Our model IHCE surpasses all baselines on both settings. The results indicate that IHCE is able to effectively perform clinical automatic diagnosis by exploiting the hierarchy and co-occurrence structure of the medical ontology and the attention mechanism. The specific analysis is as follows:

Table 2. Overall performance on MIMIC-III, where "–" means that the baseline did not report the result of the corresponding metric.

Models	MIMIC-III full						MIMIC-III top-50				
	AUC		F1-score		P@K		AUC		F1-score		P@K
	Macro	Micro	Macro	Micro	8	15	Macro	Micro	Macro	Micro	5
LR	56.1	93.7	1.1	27.2	54.2	41.1	82.9	86.4	47.7	53.3	54.6
H-SVM	–	–	–	44.1	–	–	–	–	–	–	–
C-MemNN	–	–	–	–	–	–	83.3	–	–	–	42.0
C-LSTM-Att	–	–	–	–	–	–	–	90.0	–	53.2	–
HARNN	–	–	–	40.5	–	–	–	–	–	–	–
BiGRU	82.2	97.1	3.8	41.7	58.5	44.5	82.8	86.8	48.4	54.9	59.1
LEAM	–	–	–	–	–	–	88.1	91.2	54.0	61.9	61.2
CNN	80.6	96.9	4.2	41.9	58.1	44.3	87.6	90.7	57.6	62.5	62.0
CAML	89.5	98.6	8.8	53.9	70.9	56.1	87.5	90.9	53.2	61.4	60.9
DR-CAML	89.7	98.5	8.6	52.9	69.0	54.8	88.4	91.6	57.6	63.3	61.8
MultiResCNN	91.0	98.6	8.5	55.2	73.4	58.4	89.9	92.8	60.6	67.0	64.1
IHCE ($\mathcal{T} = 3$)	**92.9**	**98.9**	**10.4**	**57.3**	**73.5**	**58.7**	**91.0**	**93.6**	**64.7**	**69.6**	**65.2**

(1) In the MIMIC-III full setting, compared with the SOTA method MultiResCNN, the IHCE improves Macro-AUC, Macro-F1 and Micro-F1 by 2.1%, 22.4% and 3.8%, respectively. It is worth noting that all models have low Macro-F1 scores on MIMIC-III full setting because the diagnosis codes space is too large, and the distribution is extremely unbalanced. Nevertheless, what is exciting is

that our model has **18.2%** and **22.4%** improvements in this metric compared to CAML and MultiReCNN, respectively. The reason is the IHCE considers hierarchical inheritance structure and dependencies. So the IHCE can assists the processing of low-frequency codes based on high-level prediction results. Similarly, we can observe that H-SVM with a hierarchical structure is better than BiGRU without a hierarchical structure in Micro-F1. However, the performance of H-SVM is lower than that of CAML and MultiReCNN because CAML and MultiReCNN utilize a primary attention mechanism to improve the ability to retrieve critical information. Furthermore, compared to CAML and MultiResCNN, our model has multiple attention mechanisms, so our model has more robust key information retrieval capabilities and surpasses them in all metrics.

(2) In the MIMIC-III top-50 setting, compared with the SOTA method MultiResCNN, the IHCE improves Macro-F1 and Micro-F1 by 6.8% and 3.9%, respectively. Although there are only 50 diagnosis codes in MIMIC-III top-50 setting, it still shows a slight long-tail effect. The IHCE has a significant improvement on the Macro-f1, indicating that our model can employ the hierarchical structure to alleviate this problem. It is worth noting that even though DR-CAML utilize codes description as regularization to assist in the allocation of diagnosis codes that are difficult to predict, the effect is still limited compared to CNN. However, the IHCE utilizes the co-occurrence structure between codes to solve this problem better.

5.5 Ablation Study

In this section, to verify each component's effectiveness in the IHCE, we perform ablation studies. The specific results are shown in Table 3. It is observed that removing each component will cause F1 to decrease, which illustrates the effectiveness of each component of our model. (1) **HPL's effectiveness:** After removing the HPL module, the macro-average metrics drop significantly, indicating that the inheritance-guided hierarchical assignment mechanism introduced by our IHCE has a significant effect on solving the long-tail effect. (2) **ORL's effectiveness:** After ORL is removed, the overall performance of IHCE declines because the method cannot model disease co-occurrence relationships. However, this ability is beneficial for assigning diseases for which it is not easy to find textual clues in the clinical note. (3) **Attention mechanism's effectiveness:** We only retain the *Code Specific Attention* module, which expands the attention mechanism in MultiResCNN and improves almost all metrics. It shows that our attention mechanism can better extract essential information to prevent the situation of finding a needle in a haystack.

5.6 Performance at Different Levels

In the clinical automatic diagnosis task, it is important to assign the diagnosis codes of the last level to the patient. It is also essential to evaluate the performance at different levels because, in some cases, a different granularity of codes may be required. Therefore, we compared the performance of IHCE and

Table 3. Ablation study results, where "w/o" indicates without.

Models	MIMIC-III full			MIMIC-III top-50		
	Macro-AUC	Macro-F1	Micro-F1	Macro-AUC	Macro-F1	Micro-F1
MultiResCNN (SOTA)	91.0	8.5	55.2	89.9	60.6	67.0
w/o ORL& HPL	91.0	8.7	55.9	89.9	61.2	66.9
w/o HPL	92.6	9.2	56.0	89.9	62.1	67.5
w/o ORL	**93.1**	10.0	56.7	90.6	63.6	68.5
IHCE ($\mathcal{T} = 3$)	92.9	**10.4**	**57.3**	**91.0**	**64.7**	**69.6**

IHCE-DPU at each hierarchical level. Note that this comparison is based on $\mathcal{T} = 3$. The IHCE-DPU ignores the dependency between the levels by removing the DPU in the HPM. In Fig. 5, we can see that the performance of IHCE at almost all levels is better than IHCE-DPU. Moreover, we can also notice that the performance on all metrics tend to decrease when the hierarchy deepens, and the trend on Macro-F1 in MIMIC-III full setting is the most obvious. The reason is that as the level deepens, the number of codes of this level will increase rapidly (e.g., the MIMIC-III full setting has 5,125, 8,925 unique codes in level-3 and level-4 respectively, as shown in Table 1). Moreover, we can notice that IHCE reduces this negative factor compared with IHCE-DPU by modeling the dependency among different hierarchical levels.

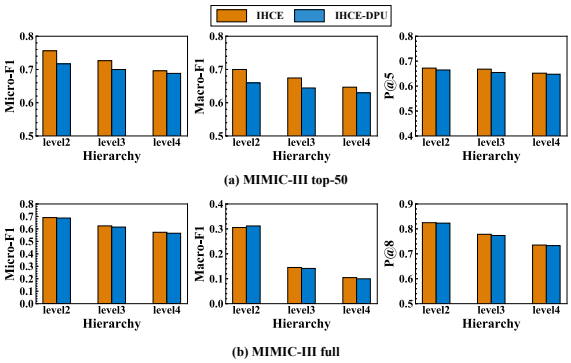

Fig. 5. Performance at different levels in hierarchy.

5.7 Effect of the Number of Hierarchical Levels

In this section, we turn to figure out the effect of the number of hierarchical levels, i.e., $\mathcal{T}$. To that end, a series of experiments are conducted to evaluate the effectiveness under different settings. Specifically, $\mathcal{T} = n$ means choosing the last n levels in Table 1. For example, $\mathcal{T} = 2$ means that we choose level-3 and level-4.

From Fig. 6, we can conclude that the models that consider hierarchical structure preform much better than models that do not. The performance rises when the number $\mathcal{T}$ of levels increases because high-level information has a guiding effect on the low level. However, the performance decreases when the $\mathcal{T}$ continuously increases. The reason is that when the number of codes between different levels is not an order of magnitude, errors caused by high-level results will still seriously affect low-level levels, although DPU has a mitigating effect. Specifically, for the MIMIC-III full setting, when $\mathcal{T} = 4$, the model will extend level-1 with only 199 diagnosis codes, which is not in the same order of magnitude as other levels. For the MIMIC-III top-50 setting, each level's magnitude is not much different, and the impact of this error will also be reduced.

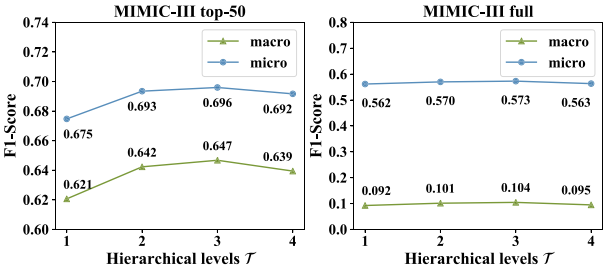

Fig. 6. Performance by varying the number of hierarchical levels.

6 Conclusion

In this paper, we proposed a novel Inheritance-guided Hierarchical Assignment with Co-occurrence-based Enhancement (IHCE) framework for clinical automatic diagnosis, which could jointly exploit code hierarchy and code co-occurrence. We utilized GCN to obtain the correlation between medical ontology. Moreover, we proposed a hierarchical joint prediction strategy based on the attention mechanism. Experimental results on real medical datasets show that our model has obtained state-of-the-art performance with substantial improvements in different evaluation metrics. We believe that our method can also be used for other tasks that require the application of hierarchical structure and label co-occurrence, such as hierarchical multi-label classification.

Acknowledgements. This research was partially supported by grants from the National Key Research and Development Program of China (Grant No. 2018YFB1402600), the National Natural Science Foundation of China (Grant No. 62072423), and the Key Research and Development Program of Anhui Province (No. 1804b06020377).

References

1. Bai, T., Vucetic, S.: Improving medical code prediction from clinical text via incorporating online knowledge sources. In: The World Wide Web Conference, pp. 72–82 (2019)
2. Bastings, J., Titov, I., Aziz, W., Marcheggiani, D., Sima'an, K.: Graph convolutional encoders for syntax-aware neural machine translation. arXiv preprint arXiv:1704.04675 (2017)
3. Baumel, T., Nassour-Kassis, J., Cohen, R., Elhadad, M., Elhadad, N.: Multi-label classification of patient notes a case study on ICD code assignment. arXiv preprint arXiv:1709.09587 (2017)
4. Esteva, A., et al.: A guide to deep learning in healthcare. Nat. Med. **25**(1), 24–29 (2019)
5. Huang, W., et al.: Hierarchical multi-label text classification: an attention-based recurrent network approach. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1051–1060 (2019)
6. Johnson, A.E., et al.: MIMIC-III, a freely accessible critical care database. Sci. Data **3**(1), 1–9 (2016)
7. Kavuluru, R., Rios, A., Lu, Y.: An empirical evaluation of supervised learning approaches in assigning diagnosis codes to electronic medical records. Artif. Intell. Med. **65**(2), 155–166 (2015)
8. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
9. Li, F., Yu, H.: ICD coding from clinical text using multi-filter residual convolutional neural network. In: AAAI, pp. 8180–8187 (2020)
10. Li, S., Zhou, J., Xu, T., Liu, H., Lu, X., Xiong, H.: Competitive analysis for points of interest. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1265–1274 (2020)
11. Liu, N., Zhang, W., Li, X., Yuan, H., Wang, J.: Coupled graph convolutional neural networks for text-oriented clinical diagnosis inference. In: Nah, Y., Cui, B., Lee, S.-W., Yu, J.X., Moon, Y.-S., Whang, S.E. (eds.) DASFAA 2020. LNCS, vol. 12112, pp. 369–385. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59410-7_26
12. Liu, Y., Li, Z., Huang, W., Xu, T., Chen, E.H.: Exploiting structural and temporal influence for dynamic social-aware recommendation. J. Comput. Sci. Technol. **35**, 281–294 (2020)
13. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
14. Ma, F., You, Q., Xiao, H., Chitta, R., Zhou, J., Gao, J.: KAME: knowledge-based attention model for diagnosis prediction in healthcare. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, pp. 743–752 (2018)
15. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
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), pp. 1101–1111 (2018)
17. Paszke, A., et al.: PyTorch: an imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems, pp. 8026–8037 (2019)

18. Peng, H., et al.: Large-scale hierarchical text classification with recursively regularized deep graph-CNN. In: Proceedings of the 2018 World Wide Web Conference, pp. 1063–1072 (2018)
19. Perotte, A., Pivovarov, R., Natarajan, K., Weiskopf, N., Wood, F., Elhadad, N.: Diagnosis code assignment: models and evaluation metrics. J. Am. Med. Inform. Assoc. **21**(2), 231–237 (2014)
20. Prakash, A., et al.: Condensed memory networks for clinical diagnostic inferencing. arXiv preprint arXiv:1612.01848 (2016)
21. Shi, H., Xie, P., Hu, Z., Zhang, M., Xing, E.P.: Towards automated ICD coding using deep learning. arXiv preprint arXiv:1711.04075 (2017)
22. Singh, H., Schiff, G.D., Graber, M.L., Onakpoya, I., Thompson, M.J.: The global burden of diagnostic errors in primary care. BMJ Qual. Saf. **26**(6), 484–494 (2017)
23. Wang, G., et al.: 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), pp. 2321–2331 (2018)
24. Yao, L., Mao, C., Luo, Y.: Graph convolutional networks for text classification. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 7370–7377 (2019)
25. Yichao, D., Tong, X., Jianhui, M., Enhong, C., Yi Zheng, T.L., Guixian, T.: An automatic ICD coding method for clinical records based on deep neural network. Big Data Res. **6**(5)