Artificial Intelligence In Medicine 154 (2024) 102916

Contents lists available at ScienceDirect

Artificial Intelligence In Medicine

journal homepage: www.elsevier.com/locate/artmed

Research paper

EHR coding with hybrid attention and features propagation on disease knowledge graph

Tianhan Xu [a,b], Bin Li [a,b,*], Ling Chen [a,b], Chao Yang [a,b], Yixun Gu [c], Xiang Gu [d]

[a] School of Information Engineering, Yangzhou University, Yangzhou, 225127, Jiangsu, China
[b] Jiangsu Province Engineering Research Center of Knowledge Management and Intelligent Service, Yangzhou, 225127, Jiangsu, China
[c] Department of Oncology, Northern Jiangsu Province People Hospital of Yangzhou University, Yangzhou, 225001, Jiangsu, China
[d] Department of Cardiovascular, Northern Jiangsu Province People Hospital of Yangzhou University, Yangzhou, 225001, Jiangsu, China

ARTICLE INFO

Keywords:
EHR coding
ICD
Disease knowledge graph
Hybrid attention
Graph propagation
Explainability

ABSTRACT

And sentences associated with these attributes and relationships have been neglected. in this paper We propose an end-to-end model called **K**nowledge **G**raph **E**nhanced neural **net**work (KGENet) to address the above shortcomings. specifically We first construct a disease knowledge graph that focuses on the multi-view disease attributes of ICD codes and the disease relationships between these codes. we also use a long sequence encoder to get EHR document representation. most importantly KGENet leverages multi-view disease attributes and structured disease relationships for knowledge enhancement through hybrid attention and graph propagation Respectively. furthermore The above processes can provide attribute-aware and relationship-augmented explainability for the model prediction results based on our disease knowledge graph. experiments conducted on the MIMIC-III benchmark dataset show that KGENet outperforms state-of-the-art models in both model effectiveness and explainability Electronic health record (EHR) coding assigns International Classification of Diseases (ICD) codes to each EHR document. These standard medical codes represent diagnoses or procedures and play a critical role in medical applications. However, EHR is a long medical text that is difficult to represent, the ICD code label space is large, and the labels have an extremely unbalanced distribution. These factors pose challenges to automatic EHR coding. Previous studies have not explored the disease attributes (e.g., symptoms, tests, medications) of ICD codes and the disease relationships (e.g., causes, risk factors, comorbidities) between them. In addition, the important roles of medical

1. Introduction

In healthcare, the electronic health record (EHR) serves as a repository for various clinical patient information, including medical history, vital signs, laboratory test results, and clinical notes. EHR coding assigns International Classification of Diseases (ICD) codes to each EHR document [1]. These ICD codes enable streamlined medical data retrieval, billing, epidemiological assessment, and health management. However, it has been reported that EHR coding incurs significant costs of approximately $25 billion annually [2]. The manual process of EHR coding is not only time-consuming but also inefficient. In light of this, a number of recent studies have explored the use of algorithms to automate the EHR coding process.

Automatic EHR coding is a critical task, as shown in Fig. 1. However, it presents several challenges. (1) *The label space of EHR coding is large.* For instance, ICD-9 has 16,000 diagnosis codes while ICD-10 has increased five-fold [3]. (2) *The labels have an extremely unbalanced distribution.* Some ICD codes are common while others are rare, thereby reducing the model's recognition rate on rare labels. (3) *EHR is difficult to represent.* EHR contains many medical terminology descriptions. Poor representation of EHR significantly affects ICD code assignment. (4) *Weak explainability of automatic EHR coding results.*

Previous studies have attempted to address the above challenges. (1) *Model based on the improved encoder.* Some of these studies focus on model encoder improvements. Research [4–6] have used Convolutional Neural Network (CNN) and its variants as the encoder, while Recurrent Neural Network (RNN) was employed in [7,8] as the encoder. The transformer-based model was also used in [9–11] to encode EHR. Few-shot learning [12] and zero-shot learning [13] frameworks for EHR coding are employed to increase the detection rate of rare diseases. (2) *Model based on the structured labels.* ICD codes have a hierarchical tree structure, therefore [7,14] use the hierarchical relationship to solve the problem of large label space. Additionally, the RPGNet model [15] utilizes reinforcement learning and treats EHR coding as a path-generation task along the hierarchical structure. (3) *Model based on external knowledge.* Compared with regular text, EHR has the characteristics of **intensive medical terms**, so some recent works consider

* Corresponding author at: School of Information Engineering, Yangzhou University, Yangzhou, 225127, Jiangsu, China.
E-mail address: lb_kmis@yzu.edu.cn (B. Li).

https://doi.org/10.1016/j.artmed.2024.102916
Received 1 May 2023; Received in revised form 8 April 2024; Accepted 17 June 2024
Available online 18 June 2024

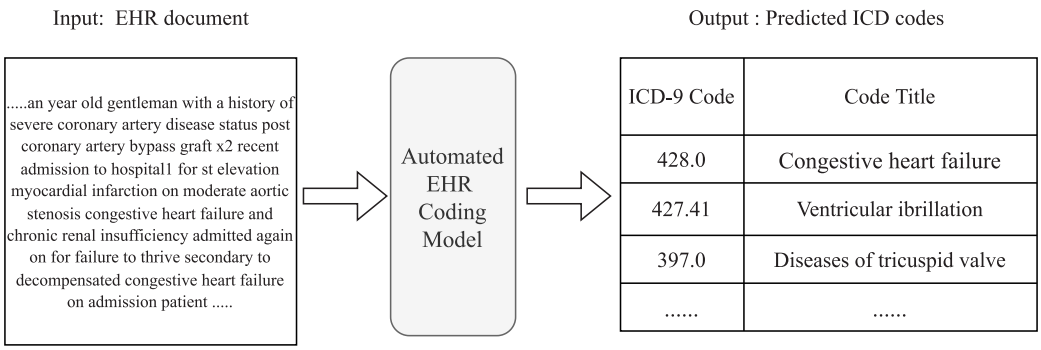

Fig. 1. The automatic EHR coding process.

the use of knowledge. MSATT-KG [16] employs a graph convolutional network to represent ICD codes using ICD titles. The work [17] extracts entities associated with ICD-9 codes via Freebase and integrates them with medical records for prediction. However, these works still have the following shortcomings that need to be addressed: **(1) Lack of ICD disease attributes.** The EHR coding task uses ICD codes as labels, as shown in Fig. 1. Prior studies have only considered the title of the ICD code and ignored other critical medical terminologies (e.g., symptoms, tests, medications). **(2) Lack of ICD disease relationships.** Previous studies have considered the hierarchical relationship of ICD codes, but not the medical associations (e.g., causes, risk factors, comorbidities) between diseases represented by ICD codes. **(3) Insufficient feature interaction between EHR text and ICD labels.** Most previous work ignores the interactions between EHR text, disease attributes, and disease relationships. Therefore, keywords and phrases related to ICD labels in the EHR cannot be well-mined. Overall, the above issues will result in lower model accuracy and a reduced level of explainability.

To overcome these shortcomings, we construct an innovative disease knowledge graph. We use the multi-view attributes and disease relationships to refine the ICD code representation, enriching the semantic information while enhancing the medical causal relationships between ICD codes. To fully leverage the disease knowledge graph for knowledge enhancement, we introduce an end-to-end model called **K**nowledge **G**raph **E**nhanced Neural **Net**work (KGENet). We propose a hybrid attention module to extract features in the EHR that are strongly associated with the multi-view disease attributes of each ICD code, enabling us to acquire more fine-grained explainable evidence. Moreover, a graph propagation module is devised to explore the correlation among the EHR features via structured disease relationships. Our resources and code are available at github repository.[1] The main innovations and contributions of this work are as follows:

- We construct a **disease knowledge graph** with **multi-view disease attributes** and **disease relationships**. This provides richer medical knowledge and better explainability for our model.
- We propose a novel knowledge-enhanced model named KGENet. KGENet uses a long sequence encoder to represent EHR long text sequences. It then leverages a **hybrid attention module** and a **graph propagation module** to extract both label-specific features and interactive features. In addition, the two modules can provide both **attribute-aware** and **relationship-augmented** explainability for automatic EHR coding.
- To evaluate the effectiveness of KGENet, we conducted extensive experiments on a benchmark dataset. The results demonstrate that the proposed model outperforms the state-of-the-art models in terms of both performance metrics and explainability.

[1] https://github.com/xutianhan/KGENet.

2. Related work

In this section, we present baseline models for EHR coding from the three perspectives listed above. All these models are based on deep learning.

2.1. Model based on the improved encoder

Some researchers use CNN-based models. The CAML model [4] is a well-established CNN-based approach for EHR coding. Furthermore, to address the issue of underrepresented codes, the authors propose a variant of the CAML model, called Description Regularized CAML (DR-CAML). Li et al. presented another CNN-based model MultiResCNN [5] which leverages multi-filter convolution and a residual network for EHR text feature extraction. In addition, Luo et al. introduce Fusion [18], a model that employs an improved CNN to compress sparse and redundant word representations into information-rich and dense word representations as features.

Some other researchers adopt the model of RNN structure. Dong et al. introduced the HLAN model [8] which encodes EHR text using Bidirectional Gated Recurrent Unit (Bi-GRU) and implements a hierarchical label-wise attention mechanism at both word and sentence levels. The LAAT model [7], on the other hand, utilizes a bidirectional Long Short-Term Memory (LSTM) network as the label extractor and incorporates a label attention layer to learn label-specific vectors for each ICD code in the EHR. Furthermore, JointLAAT [7] combines the LAAT model with the task of predicting normalized codes and appends an additional specific label vector to the output. Joint training is then performed to obtain the final prediction.

In recent years, some Transformer-based models have been proposed for EHR coding. One such model is TransICD [9], which utilizes a Transformer encoder to capture contextual word representations and implements a label-wise attention mechanism. Another study, ISD [6], adopts a bidirectional multilayer Transformer decoder to extract interactive shared representations from both clinical notes and labels. Similarly, Malte Feucht et al. proposed the use of Longformer as an EHR text encoder and experimented with various variant models [10]. Liu et al. introduced a Hierarchical Label-Wise Attention Transformer (HiLAT) model [11] aimed at enhancing the explainability of ICD coding. The model uses XL-Net as a pre-training model and uses a token-level and chunk-level hierarchical structure to handle EHR long text.

2.2. Model based on the structured labels

Because the ICD codes follow a hierarchical tree structure, some researchers have used graph neural networks (GNN) to represent ICD codes in EHR coding. For example, HyperCore [14] used a GCN [19] to derive the code representation by calculating the co-occurrence frequency of ICD codes in EHR texts, thereby improving the incomplete prediction of codes. Similarly, Du et al. [20] used the tree structure

of the ICD taxonomy to capture code dependencies. In addition, Wang et al. [21] introduced two approaches to constructing edge weights in the ICD code graph using the point-wise mutual information and the TF-IDF value.

The difference with the above work is that we use a directed graph to encode hierarchical ICD relations. The reason for using a directed graph is twofold: First, in general, child nodes inherit all attributes and HER features of their parent nodes. However, the reverse is not true. For example, the node 'heart failure' has two child nodes: 'left heart failure' and 'right heart failure'. 'Left heart failure' possesses all the attributes and HER features of 'heart failure', while also having its own features that 'right heart disease' does not have. Therefore, we focus more on the transmission from parent nodes to child nodes and ignore the transmission from child nodes to parent nodes. Second, whether it is a unidirectional or bidirectional graph, each transmission of information generates noise, and the more transmissions, the greater the impact of noise. By removing the transmission from child nodes to parent nodes, we can minimize the impact of noise as much as possible.

2.3. Model based on external knowledge

The use of knowledge-enhanced models in EHR coding and medical prediction tasks has become increasingly popular [22], as they not only enhance model accuracy but also provide more logical explanations, which is imperative for healthcare-oriented research. To this end, MSATT-KG [16] employs a graph convolutional network to represent ICD codes using ICD description knowledge. Teng and Yang et al. [17] extracted entities associated with ICD-9 codes via freebase and integrated them with medical records for the prediction task, referred to as G-coder. Furthermore, Zou et al. [23] presented a method that tackles the challenge of patient hospitalization prediction by augmenting the vector representation of EHR with information extracted from diverse knowledge graphs. Yuan et al. [24] proposed to use synonyms in UMLS to learn a more comprehensive ICD code representation and use a synonym matching network to improve code classification performance.

Compared to the above work, we consider multi-view disease attributes as well as multiple types of relationships when constructing the disease knowledge graph. In addition, we design a more appropriate model framework to utilize the above knowledge.

3. Disease knowledge graph

This section first introduces the disease knowledge graph. Then, we present the steps of its construction.

3.1. Disease knowledge graph definition

3.1.1. Entity

Since the labels of the EHR coding task are ICD codes, we use ICD codes as the entities of the disease knowledge graph. In designing ontology, we followed the specification of the National Center for Health Statistics (NCHS) for ICD-9-CM [25].

3.1.2. Attribute

Successful EHR coding requires the identification of key medical terms associated with relevant ICD codes. To this end, we created **multi-view disease attributes** for each ICD entity including descriptions, symptoms, physical signs, tests, treatments, and medicines. The format of the attributes is text that consists of multiple words or phrases. These attributes were selected due to their high frequency in EHR texts and their essential role in accurate disease diagnosis.

3.1.3. Relation

The ICD code is logically a hierarchical tree structure, and there is a parent–child relationship between disease codes. Throughout the ICD code, different digits indicate different levels. For example, in the ICD-9 coding system, code '428' represents 'heart failure' and is a parent node. Heart failure can be further subdivided into 'left heart failure' as '428.0' and 'right heart failure' as '428.1', which are child nodes. Our method is based on the MIMIC-III ICD-9 code system, which has a total of 4 layers. As the layers increase, the disease codes become more and more detailed. We call this the 'hierarchical relationships' between diseases. The ancestor of each node can be determined by its code.

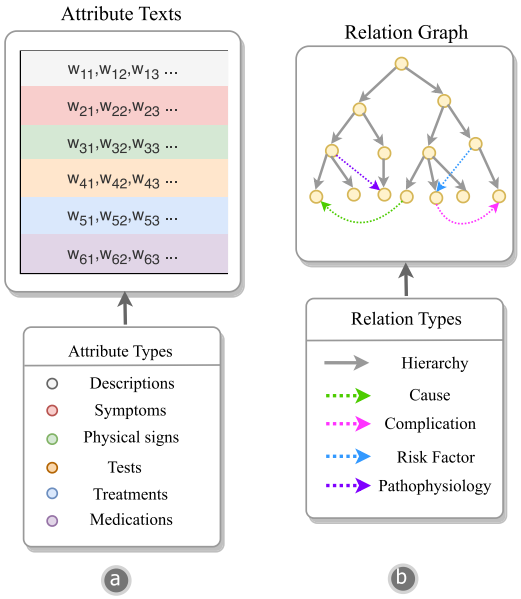

Fig. 2. Attributes and relations of the disease knowledge graph. Different dots in (a) represent different types of attribute text. The gray solid arrows in (b) represent the hierarchical relationships, and the colored dotted arrows represent different types of associated relationships. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In addition, when constructing the disease knowledge graph, we consider other disease relationships between multiple diseases. For example, hypertension causes 'heart failure', and 'heart failure' combines with 'kidney failure'. We call these 'associated relationships'. In our knowledge graph these relationships include cause, risk factor, complication, and pathophysiology [26,27].

Unlike previous work, the directed edges between the ontology graphs we construct are intended to more clearly express the dependencies between nodes, thus improving the explainability of the model. For example, if it is necessary to determine whether a patient in an EHR sample has 'left heart failure' or 'right heart failure', the EHR features associated with the 'heart failure' parent node are passed to the current node. Note that our 'heart failure' parent node contains its own disease-related attributes, which ensures that even if the attributes of the child node cannot be propagated to its parent node, the model still has enough information to determine the classification of the parent node.

Fig. 2 shows the attribute types and relationship types of the disease knowledge graph. The disease knowledge graph is the foundation of our knowledge-enhanced model and can also provide explainability.

3.2. Construction method

Entity Construction. First, we treat each ICD-9 disease code as an entity and use authoritative medical resources such as UMLS [28], Mayo Clinic [29], and Wikipedia [30] to obtain the attribute information of each entity. This attribute information includes symptoms, physical signs, laboratory tests, treatments, and medications. Specifically, we primarily obtain attribute information based on the UMLS Web

API. We then design a web crawler to crawl data from Wikipedia and Mayo Clinic and extract attribute information through rule templates to provide additional information. For diseases attributes that cannot be accurately found in authoritative resources, we will supplement them through the ChatGPT [31] API.

Relation Construction. Next, we construct the hierarchical relationship between each ICD entity based on the number of digits of the ICD code and the specific code value. We search for interactive relationships between ICD codes using the UMLS Web API and use templates to supplement the relationships by extracting unstructured data from Wikipedia. These relationships include cause, complication, risk factor, and pathophysiology.

The design of the disease knowledge graph was inspired by previous work [32–34], focusing on the attributes and relationships of the ICD entities. In addition, we also use human experts to verify and supplement the results to build a complete ICD knowledge graph.

4. The proposed model

4.1. Formal definition

We view EHR coding as a long text, multi-label classification task. The input of our model is the EHR text and the disease knowledge graph. EHR text $X = [x_1, x_2, \dots, x_n] \in \mathcal{R}^n$, where n is the length of the words in the EHR. The disease knowledge graph is denoted as $G = \{V, D, R, T\}$, where $V = [v_1, v_2, \dots, v_C]$ is a set of ICD entities (C indicates the total number of ICD entities). Each entity v_i in V has a set of text attribute descriptions D. R denotes the relationship types, and T is the set of triples. R and T form the relation graph in Fig. 2. The output binary vector $y \in \mathcal{R}^C$ is comprised of elements representing ICD codes, where a value of 1 signifies an assignment to the EHR, while 0 indicates the opposite.

4.2. Overview of model

A graphical representation of the proposed model architecture is presented in Fig. 3. The proposed framework is composed of 5 parts, including a Long sequence encoder (LSE), an attributes encoder (AE), a hybrid attention module (HAM), a graph propagation module (GPM), and a classification module (CM). LSE is used to obtain EHR long text representation. AE is used to encode the multi-view text attributes of ICD codes to obtain the attribute representation of each label. HAM first obtains the EHR label-specific features. These features are then processed through the GPM to derive EHR interactive features. Finally, the label-specific features and interactive features are integrated through CM to predict the probabilities of multiple ICD labels. Details of each module are shown in Fig. 4.

4.3. Long Sequence Encoder (LSE)

Our input EHR text comes from discharge summaries, which may contain text noise. So we first remove punctuation and stop words, including words such as 'admission', 'date', 'birth', 'patient', 'year', 'work', etc., and make all words to lowercase. To make the model pay more attention to clinical feature words, we use `en_core_sci_sm` and `en_core_med7_trf` models in Spacy [35] for Named Entity Recognition (NER). `en_core_sci_sm` is a small English model specifically used for scientific and medical texts, which has high accuracy in identifying symptoms, signs, and laboratory test entities. `en_core_med7_trf` is a transformer-based model specifically used for medical entity identification. It has high accuracy in identifying drug and treatment entities. We merge the two sets of medical entities to form the input of our model.

Next, we tokenize the medical entity text X into the word sequence $W = [w_1, w_2, \dots, w_N]^T$, where N is the maximum length of the input words. If the number of words in X is greater than N, we truncate it to N. On the contrary, we perform padding operations. Then, we map the words to the pre-trained embeddings. The word embedding matrix E_X obtained is:

$$E_X = Embed\,(Tokenizer(X)) = \left[e_1, e_2, \dots, e_N\right]^T, \tag{1}$$

where $E_X \in \mathcal{R}^{N \times d_e}$, and d_e is the dimension of embedding.

Then, we feed the word embedding matrix E_X into a transformer encoder to get the EHR representation.

$$H = TransformerEncoder(E_X) = \left[h_1, h_2, h_3, \dots, h_N\right]^T, \tag{2}$$

where $H \in \mathcal{R}^{N \times d_e}$, is the EHR hidden representation matrix.

We use the Clinical-Longformer [36] model as an encoder for the EHR. Clinical-Longformer is a model based on Longformer [37] that is specifically used to process clinical text. Its token length can reach 4096.

4.4. Attributes Encoder (AE)

In order to prevent the loss of label semantic information, we encode the attribute texts D of multiple views in the disease knowledge graph separately.

The acquisition of embedding for the ith label text D_i's specific view j is performed using the ClinicalBERT model [38] pre-trained on clinical notes.

$$E_i^j = ClinicalBERT(D_i^j) = \left[e_1, e_2, \dots, e_K\right]^T, \tag{3}$$

where K is the length of words in D_i^j. Then we get the representation of view j of the ith label q_i^j through max-pooling:

$$q_i^j = max\text{-}pooling(E_i^j). \tag{4}$$

We concatenate all view vectors and perform dimensionality reduction through a fully connected layer:

$$q_i = FC\left(\oplus_{j=1}^{M} q_i^j\right), \tag{5}$$

where M denotes the number of views in the multi-view text. As shown in Fig. 2, M = 6 in our disease knowledge graph. The obtained result $q_i \in \mathcal{R}^{d_a}$ is called multi-view attributes representation, and d_a is its dimension. $Q = [q_1, q_2, \dots, q_C]^T \in \mathcal{R}^{C \times d_a}$ represents the attribute representation matrix for all the labels.

4.5. Hybrid Attention Module (HAM)

Intuitively, we use label attention to capture text features associated with specific labels in the EHR training samples. Due to the unbalanced label distribution of the dataset, it is difficult for the model to learn the features associated with rare labels. Therefore, a cross-attention module is designed to provide prior knowledge to handle the assignment of rare ICD labels.

Label Attention (LA). In the context of EHR multi-label classification, a single EHR instance may be associated with multiple labels. To this end, we propose to use the attention mechanism [39] on the EHR representation $H \in \mathcal{R}^{N \times d_e}$, where N is the maximum word length in EHR and d_e is the dimension of EHR word embeddings.

$$T = tanh(W_1 H^T), \quad A^{(L)} = softmax(W_2 T). \tag{6}$$

Here $W_1 \in \mathcal{R}^{d_m \times d_e}$ and $W_2 \in \mathcal{R}^{1 \times d_m}$ are trainable parameters. $T \in \mathcal{R}^{d_m \times N}$ is the middle layer, and d_m is the dimensional parameters of T. $A^{(L)} \in \mathcal{R}^{C \times N}$ is the attention score matrix. Then we can obtain the label-aware EHR features $F^{(L)} \in \mathcal{R}^{C \times d_e}$ under the label attention mechanism:

$$F^{(L)} = A^{(L)} H. \tag{7}$$

Cross Attention (CA). We use the attributes representation matrix $Q \in \mathcal{R}^{C \times d_a}$ as the query, $H \in \mathcal{R}^{N \times d_e}$ as the key and value to obtain the knowledge similarity score $S^{(C)} \in \mathcal{R}^{C \times N}$. The intuitive explanation for

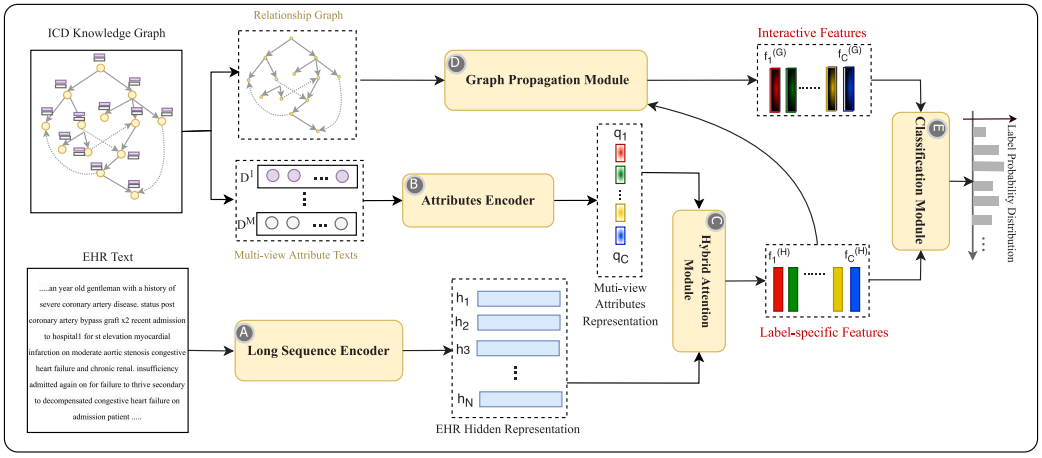

Fig. 3. Framework of the proposed model. The model framework is divided into 5 parts, namely (A) Long Sequence Encoder, (B) Attributes Encoder, (C) Hybrid Attention Module, (D) Graph Propagation Module, and (E) Classification Module.

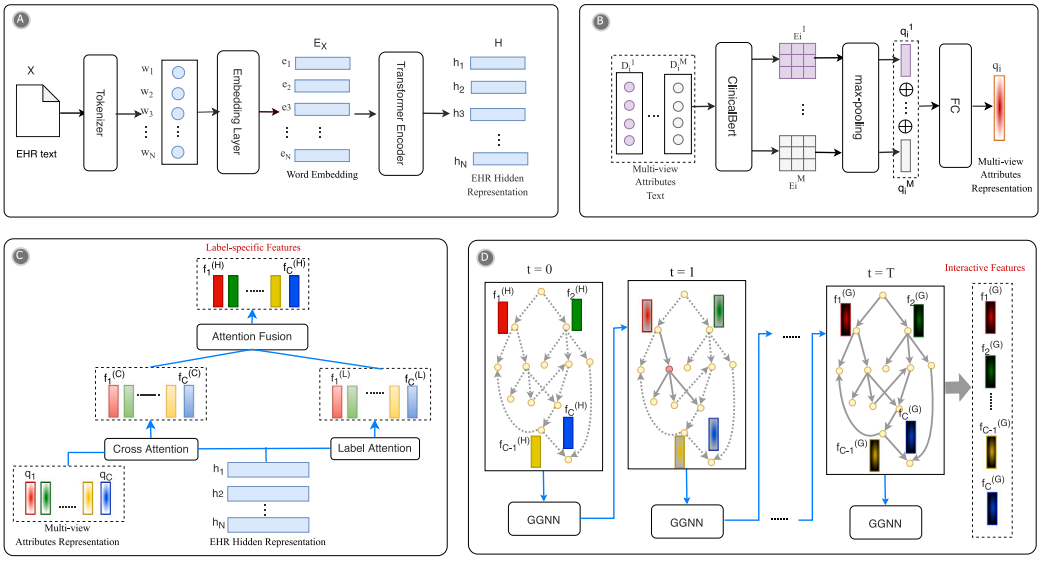

Fig. 4. Details of the proposed model. (A) Long Sequence Encoder, (B) Attributes Encoder, (C) Hybrid Attention Module, and (D) Graph Propagation Module.

this is that we use the attribute knowledge of each label to query the features of the EHR that are closest to its semantics, and the higher the similarity, the higher the matching score $S^{(C)}$.

$$S^{(C)} = QUH^T, \tag{8}$$

where $U \in \mathcal{R}^{d_a \times d_e}$ is a dimension transformation matrix between the EHR representation and the multi-view attributes representation. We then normalize the similarity score $S^{(C)} \in \mathcal{R}^{C \times N}$ to the interval [0,1] to obtain the cross attention score matrix $A^{(C)} = \left(A_i^j\right)_{i=\{1,\ldots,N\}, j=\{1,\ldots,C\}} \in \mathcal{R}^{C \times N}$ as follows:

$$A_i^j = e^{S_i^j} / \sum_{i=1}^{N} e^{S_i^j}. \tag{9}$$

Finally, similar to label attention, we obtain cross-attention features $F^{(C)} \in \mathcal{R}^{C \times d_e}$ through the following formula:

$$F^{(C)} = A^{(C)} H. \tag{10}$$

Attention Fusion. To simplify the computation, the output of label attention is concatenated with the output of cross attention along the ith label to obtain the features of the hybrid attention module.

$$F^{(H)} = [f_1, f_2, \ldots, f_C]^T, \quad f^{(H)} = f^{(L)} \oplus f^{(C)}. \tag{11}$$

Here, $\oplus$ donates concatenate operation, $i \in \{1, 2, \ldots, C\}$, and $F^{(H)} \in \mathcal{R}^{C \times d_r}, d_r = 2d_e$.

4.6. Graph Propagation Module (GPM)

The disease knowledge graph showcases the interrelation between disease labels, which can also be observed in the label-specific features of EHR. In light of this correlation, a Graph Neural Network is employed, leveraging the disease relation graph, to propagate the label-specific features of EHR. We name the resultant features interactive features.

In fact, both hierarchical and interactive relations are directional, and the aggregated messages of neighboring nodes should decrease as the number of hops increases. As such, the Gated Graph Neural Network (GGNN) [40] is a suitable network for information propagation.

This study adopts the GGNN for propagating information over T steps, utilizing a Gated Recurrent Unit (GRU) [41] based on the disease relation graph. At step t = 0, we initialize the hidden state of the ith node using the ith column vector f_i of HAM output matrix $F^{(H)}$:

$$h_i^0 = f_i. \tag{12}$$

During message passing, different weights should be assigned to different nodes. For this purpose, we use the following formula to compute

the weights:

$$P(i,j) = \frac{N_{ij}}{N_i}. \tag{13}$$

Let N_i represent the number of occurrences of the ith label, and N_{ij} represent the number of co-occurrences of the jth label when the ith label appears. However, to avoid over-smoothing, inspired by Chen et al. [42], we formulated a_{ij} as

$$a_{ij} = \begin{cases} 0 & P(i,j) < \theta, \\ P(i,j) & P(i,j) \geq \theta. \end{cases} \tag{14}$$

Here, θ is the threshold which is a hyper-parameter used to filter the propagation of node information with a weaker correlation. At timestep t, GPM aggregates messages from neighbor nodes, which can be expressed as:

$$a_i^t = \left[\sum_j (a_{ij}) h_i^{t-1}, \sum_j (a_{ji}) h_i^{t-1} \right]. \tag{15}$$

In this way, GPM aggregates incoming and outgoing messages by the association weights between neighbor nodes. Then GPM updates the hidden state using the aggregated feature vector a_i^t and the hidden state h_i^{t-1} from the previous time step.

$$\begin{aligned} z_i^t &= \sigma(W^z a_i^t + U^z h_i^{t-1}), \\ r_i^t &= \sigma(W^r a_i^t + U^r h_i^{t-1}), \\ \tilde{h}_i^t &= tanh(W a_i^t + U(r_i^t \odot h_i^{t-1})), \\ h_i^t &= (1 - z_i^t) \odot h_i^{t-1} + z_i^t \odot \tilde{h}_i^t. \end{aligned} \tag{16}$$

The forgotten and reset information is denoted by z_i^t and r_i^t, respectively. The logistic sigmoid function $\sigma(.)$ is employed to control the gating signals. Furthermore, element-wise multiplication $\odot$ is used to integrate gating signals with hidden states. This process iterates T times, where T is a hyper-parameter. We obtain the features of GPM $F^{(G)} \in \mathcal{R}^{C \times d_r}$ of all categories:

$$F^{(G)} = \left[h_1^T, h_2^T, \ldots, h_C^T \right]^T. \tag{17}$$

In this way, we can aggregate contextual features from multi-hop neighbors. We named the output as interactive features.

4.7. Classification Module (CM)

Weighted Fusion. The above $F^{(H)} \in \mathcal{R}^{C \times d_r}$ and $F^{(G)} \in \mathcal{R}^{C \times d_r}$ are the label-specific and interactive features of EHR extracted by the two modules of HAM and GPM respectively. We adaptively fuse these two pieces through a weighted strategy to get the final representation. The weights can be computed by:

$$\alpha = \sigma\left(FC_1(F^{(H)})\right), \quad \beta = \sigma\left(FC_2(F^{(G)})\right), \tag{18}$$

where σ is the sigmoid function, FC_1 and FC_2 are two fully connected layers. α_i and β_i indicate the importance of label-specific features and interactive features to the final representation along the ith label respectively. α_i and β_i are normalized to ensure their sum is 1. The final representation is:

$$f_i = \alpha_i \times f_i^{(H)} + \beta_i \times f_i^{(G)}. \tag{19}$$

where $f_i \in \mathcal{R}^{2d_r}$ is the row vector of the final EHR text representation matrix.

Classifier Layer. In order to classify an EHR based on its representation F, a linear layer is employed as the classifier. Subsequently, the classifier leverages a sigmoid transformation to compute the probability $\hat{y}_i$ for label i:

$$\hat{y}_i = \sigma(\gamma_i^T f_i + b_i). \tag{20}$$

The γ_i is a weight vector and b_i is a bias parameter. A binary output is predicted using a threshold value of 0.5. Consistent with prior research on multi-label classification, the binary cross-entropy loss function is adopted during the training process:

$$\mathcal{L}_{BCE} = -\sum_{i=1}^{C} \left[y_i log \hat{y}_i + (1 - y_i) log(1 - \hat{y}_i) \right]. \tag{21}$$

For ICD coding of full-label datasets, since its label space $C = 8686$ and the average positive label is only 16, it is a typical sparse label scenario. Inspired by negative sampling approach in Glove [43], for each sample we randomly select k labels from the remaining non-positive labels as negative labels. The predicted probability of unsampled negative labels is set to 0, which does not participate in the loss calculation, thus reducing the amount of computation. To balance model performance and computational burden, we choose $k = 1000$, which gives the model more opportunities to learn the differences between labels and reduces the computational burden to some extent without losing the performance of the result.

5. Experimental setup

5.1. Datasets

To test the performance of our method, we conduct experiments on the MIMIC-III dataset [44], a large, open-access database that represents a real-world dataset. The dataset consists of 58,976 admission records for 49,583 patients treated at Beth Israel Deaconess Medical Center between 2001 and 2012. Each EHR document in the MIMIC-III dataset consists of a clinical text that includes details on medical history, diagnostic findings, surgical procedures, and discharge instructions. In addition, the diagnoses and procedures performed during the patient's stay were coded by coders in descending order of their importance and relevance. It is important to note that the MIMIC-III dataset is the only publicly available and commonly used benchmark dataset for this particular task.

Consistent with previous research [5,7,17], the main objective of our study is to examine the discharge summaries contained in the EHR. These summaries serve as a condensed form of information that summarizes a patient's hospital stay. To improve the quality of the data, we applied a data-cleaning process to the discharge summaries. This involved the elimination of irrelevant information such as physician and hospital details. We use two datasets based on MIMIC-III to evaluate the effectiveness of our proposed approach.

- **Full label dataset**: We retain all the diagnostic ICD codes and their samples that appear on discharge summaries.
- **Top-50 label dataset**: We predict only the 50 most common ICD codes and filter the dataset to include cases that have at least one of the 50 most common codes.

For both of the above datasets, we randomly divide the samples into training, validation, and test sets. To ensure fairness, we use a similar partitioning strategy as in the previous work. Table 1 shows the statistical results of the data for both datasets.

5.2. Evaluation matrics

In this study, the model evaluation metrics include the area under the curve (AUC) and the F1 score. Furthermore, the evaluation of the model encompasses the computation of the proportion of the top k labels having the highest scores in the ground truth, denoted as P@k. The evaluation employs two sets of metrics, namely micro and macro, for evaluating the model's performance. Specifically, the micro metrics are employed to assess the model's performance at the instance level, while the macro metrics are used to evaluate its overall performance at a higher level.

Table 1
Statistics for MIMIC-III datasets.

Dataset/split		Samples	Tokens	Average tokens	Labels	Unique labels	Average labels
Full-label dataset	train	47,723	70,846,774	1484	758,216	8686	15.89
	val	1631	2,910,870	1784	28,897	3009	17.72
	test	3372	6,043,743	1792	61,579	4075	18.26
Top-50 label dataset	train	8066	12,338,529	1529	45,919	50	5.69
	val	1573	2,830,895	1799	9283	50	5.90
	test	1729	3,156,602	1825	10,477	50	6.06

Table 2
Experimental hyper-parameter settings. The '/' symbol indicates different settings for the label-50 and label-full datasets.

Hyper-parameter	Value	Description
N	4096	Maximum word length of the input text.
$head\_num$	1	Number of EHR transformer encoder attention heads.
d_e	768	Dimension of the EHR word embeddings.
d_a	768	Dimension of the label attribute representation.
d_m	256	Dimension of the middle layer representation.
θ	0.2/0.3	A Threshold parameter for filtering neighbor information.
T	2/3	Steps for message passing in GPM.
d_r	1536	Dimension of the label-specific feature and the interactive feature.
lr	0.001	Learning rate for gradient descent optimization.
$batch\_size$	16	Number of training examples per batch.
$epoch$	15/35	Number of times to iterate over the training set.

5.3. Implementation details

We implement KGENet using PyTorch and train the model on RTX 3090 GPU (memory 24 GB). We use Adam optimizer and early stopping in training. The hyper-parameter settings of our experiment are shown in Table 2.

6. Result and analysis

Through the analysis of the experimental results, we investigated the following research questions:

- (RQ1) Does KGENet perform best on the EHR coding task compared to the baseline model?
- (RQ2) How does the knowledge enhancement of the disease knowledge graph affect the model? What is the effect of different modules?
- (RQ3) What is the basis for the explainability of the model? What are the advantages of model explainability compared to previous approaches?

6.1. Overall performance (RQ1)

In response to RQ1, the present study provides an account of the evaluation metrics for the top-50 and full-label datasets, as presented in Table 3. All KGENet results are obtained by setting random number seeds and averaging 5 experiments.

First, the results in Table 3 show that KGENet outperforms the other models on most evaluation metrics. On the MIMIC-III-50 dataset, KGENet achieves $93.7(^{+0.2}_{-0.3})$, $95.1(^{+0.1}_{-0.2})$, $68.6(^{+0.1}_{-0.2})$, $74.2(^{+0.2}_{-0.2})$, $68.8(^{+0.3}_{-0.2})$, in terms of macro-AUC, micro-AUC, macro-F1, micro-F1, and P@5, respectively. On the MIMIC-III-Full dataset, KGENet achieves $94.1(^{+0.3}_{-0.2})$, $98.9(^{+0.2}_{-0.1})$, $12.5(^{+0.2}_{-0.3})$, $56.8(^{+0.3}_{-0.3})$, $76.4(^{+0.2}_{-0.3})$, $60.3(^{+0.3}_{-0.4})$ in terms of macro-AUC, micro-AUC, macro-F1, micro-F1, P@8, and P@15, respectively. Positive and negative values represent the upper and lower bounds, respectively. This result suggests that the disease knowledge graph has significantly improved the EHR encoding task. Notably, when compared to the ISD baseline, KGENet demonstrated an improvement of approximately 1.7% and 2.5% in terms of Micro F1 and P@8, respectively, on the MIMIC-III-Full dataset. This enhancement is attributed to KGENet's ability to represent labels using multi-view attributes, as well as the transfer learning mechanism employed for identifying rare labels based on EHR interactive features.

The second finding of this study is that Transformer-based models such as TransICD [9], Longformer+DLAC [10], and HiLAT [11] do not demonstrate superior overall performance compared to CNN-based and RNN-based models. This result may be attributed to the sample size of the dataset and the emphasis placed on local features of the text rather than long-distance associations in the EHR coding task. Hence, it can be inferred that the efficacy of an EHR coding model is not solely reliant on the type of encoder employed. Rather, the overall framework and the size and distribution of the benchmark dataset are the key determinants of a model's effectiveness.

The third observation of this study is that as the ICD code space transitions from top-50 to full, accurately predicting the corresponding ICD encoding becomes progressively challenging. We discovered that for all models, macro-average metrics, such as macro-F1, exhibit the most substantial decline. For instance, CAML's macro F1 decreased from 53.2% to 8.8%, while KGENet's macro F1 decreased from 68.6% to 12.5%. The reason for this phenomenon is due to the severe imbalance in ICD code distribution in the full-label dataset, where some ICD codes appear only once or twice, rendering the amount of data available for learning insufficient. However, it is noteworthy that KGENet, which leverages prior knowledge enhancement, experienced a significantly lower decrease compared to other baseline models.

6.2. Ablation study (RQ2)

In order to address RQ2, we conducted ablation experiments to examine the effect of different modules within KGENet. The experimental outcomes for the various KGENet variants are presented in Table 4. Firstly, "w/o knowledge" refers to KGENet models that do not utilize knowledge enhancement. We accomplished this by eliminating the representation process for the attributes and relations of codes in the disease knowledge graph and instead employing the original label-wise attention mechanism. Additionally, we evaluated the effects of removing attribute and relation representations through "w/o knowledge attribute" and "w/o knowledge relation", respectively. Secondly, "w/o HAM" refers to KGENet models without the Hybrid Attention Module. Lastly, "w/o GPM" denotes the removal of the Graph Interactive Module from KGENet. Based on these results show on Table 4, we have obtained the following insights:

In the context of the MIMIC-III-Full dataset, we observed that the removal of label knowledge in KGENet led to a significant drop in its performance, with its macro-F1, micro-F1, and P@8 metrics decreasing by 22.4%, 13.2%, and 7.7%, respectively. This result underscores the crucial role of knowledge, particularly attributes knowledge, in the KGENet framework. The ICD code knowledge helps the model to identify medical keywords and phrase features that are relevant to the ICD codes in the EHR, which are essential for accurate code recognition, particularly when dealing with large label space.

The experimental results demonstrate that KGENet's performance is adversely affected when the HAM and GPM modules are removed. Specifically, when the HAM module is eliminated, the micro-F1, macro-AUC, and P@8 metrics exhibit a respective decrease of 17.6%, 8.7%,

Table 3
Results (in %) of comparison with baselines on MIMIC-III test sets. The score values are taken from the paper on the baseline model in related work. The bold scores indicate the best results for each metric. The results of KGENet are tested 5 times with random number seeds and the average of each metric is taken as the result.

Models	MIMIC-III-50					MIMIC-III-Full					
	AUC		F1		P@k	AUC		F1		P@k	
	Macro	Micro	Macro	Micro	P@5	Macro	Micro	Macro	Micro	P@8	P@15
CAML [4]	87.5	90.9	53.2	61.4	60.9	89.5	98.6	8.8	53.9	70.9	56.1
DR-CAML [4]	88.4	91.6	57.6	63.3	61.8	89.7	98.5	8.6	52.9	69.0	54.8
MultiResCNN [5]	89.9	92.8	60.6	67.0	64.1	91.0	98.6	8.5	55.2	73.4	58.4
G-coder [17]	–	93.3	–	69.2	65.3	–	–	–	–	–	–
HyperCore [14]	89.5	92.9	60.9	66.3	63.2	93.0	98.9	9.0	55.1	72.2	57.9
MSATT-KG [16]	91.4	93.6	63.8	68.4	64.4	91.0	**99.2**	9.0	55.3	72.8	58.1
HLAN [8]	88.4	91.9	57.1	64.1	62.5	88.5	98.1	3.6	40.7	61.4	–
LAAT [7]	92.5	94.6	66.6	71.5	67.5	91.9	98.8	9.9	57.5	73.8	59.1
JointLAAT [7]	92.5	94.6	66.1	71.6	67.1	92.1	98.8	10.7	57.5	73.5	59.0
Longformer+DLAC [10]	87.8	91.6	52.2	62.2	61.1	–	–	–	–	–	–
TransICD [9]	89.4	92.3	56.2	64.4	61.7	–	–	–	–	–	–
Fusion [18]	93.1	95.0	68.3	72.5	67.9	91.5	98.7	8.3	55.4	73.6	–
HiLAT [11]	92.7	95.0	**69.0**	73.5	68.1	–	–	–	–	–	–
ISD [6]	93.5	94.9	67.9	71.7	68.2	93.8	99.0	11.9	55.9	74.5	–
KGENet(ours)	**93.7**	**95.1**	68.6	**74.2**	**68.8**	**94.1**	98.9	**12.5**	**56.8**	**76.4**	**60.3**

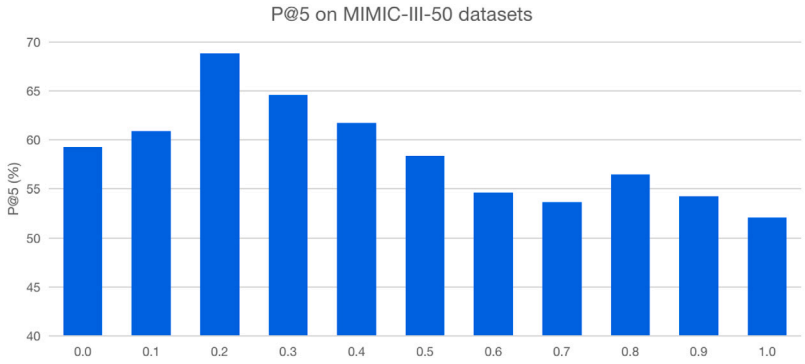

Fig. 5. Changes in P@5 relative to θ on the MIMIC-III-50 dataset.

Table 4
Average results of our model's ablation experiments on the MIMIC-III-Full test dataset. w/o stands for without.

Model	AUC		F1		P@k
	Macro	Micro	Macro	Micro	P@8
KGENet	94.1	98.9	12.5	56.8	76.4
w/o knowledge	87.8	86.1	9.7	49.3	70.5
w/o attribute	89.2	90.1	10.4	53.7	73.9
w/o relation	91.5	91.7	11.3	56.1	69.8
w/o HAM	90.8	90.3	10.3	54.9	70.7
w/o GPM	90.2	96.5	10.8	52.6	65.5

and 7.5%. These findings suggest that the HAM module is instrumental in improving model accuracy by leveraging attribute knowledge. Moreover, the removal of the GPM module causes a significant decrease in the micro-F1, macro-F1, and P@8 metrics by 13.6%, 7.4%, and 14.3%, respectively. These results validate the significance of the interaction between EHR label-specific features and indirectly demonstrate the critical role of ICD relations.

6.3. Effects of hyper-parameter

6.3.1. Effects of hyper-parameter θ

Here, we test the effect of the important hyper-parameter θ through experiments. The hyper-parameter in question serves as a threshold parameter responsible for culling extraneous information from the edges to avoid the issue of over-smoothing, as demonstrated in the formula (14). The value range of the hyper-parameter θ is [0.0, 1.0], and the interval is 0.1. The effectiveness of θ is evaluated using the performance metric P@5 for the MIMIC-III-50 dataset and P@8 for the MIMIC-III-Full dataset.

Based on the analysis of the outcomes illustrated in Figs. 5 and 6, the most favorable values of the hyper-parameter θ on the MIMIC-III-50 and MIMIC-III-Full datasets are 0.2 and 0.3, respectively. This may be due to the fact that the MIMIC-III-Full dataset contains a larger number of labels, making it necessary to escalate the threshold to filter out relatively irrelevant information from neighboring nodes.

Table 5
Micro-F1 score for different T values on two dataset settings.

Dataset	T		
	1	2	3
MIMIC-III-50	73.1	**74.2**	73.8
MIMIC-III-Full	56.3	56.6	**56.8**

6.3.2. Effects of hyper-parameter T

The hyperparameter T represents the steps for message passing in GPM. We perform parameter sensitivity analyses using the Micro-F1 score, as it is more appropriate for evaluating multi-label classification. Since the number of levels of the ICD-9 code in the MIMIC-III dataset we used is 4, we compare and analyze the results for $T = 1, 2, 3$.

The results are shown in Table 5. For the top-50 and full-label settings, the optimal T values are 2 and 3, respectively. The experimental results reflect to some extent that as the label space increases, T needs to take a larger value to propagate more information from neighboring nodes.

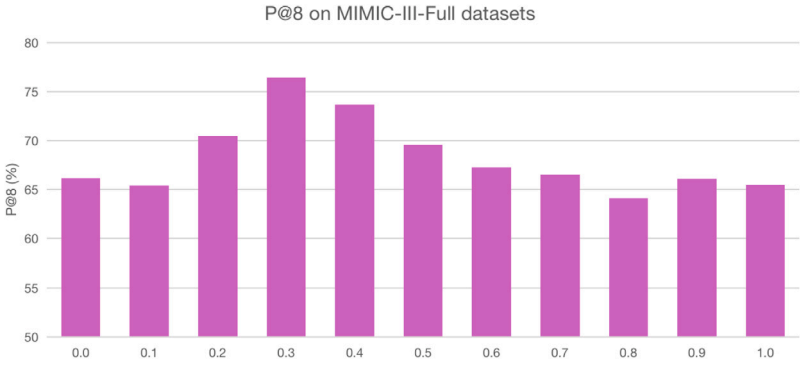

Fig. 6. Changes in P@8 relative to θ on the MIMIC-III-Full dataset.

admitted for dry gangrene on both feet past medical history end stage renal disease on hemodialysis type ii diabetes mellitus atrial fibrillation nephrolithiasis grade bladder transitional cell ca depressoin anemia gi bleed hypertension hyperlipidemia gastric cancer s p gastrectomy billroth b12 deficiency squamous cell cancer pertinant studies labs on this admission echo the left atrium is elongated the right atrium is dilated there is severe regional left ventricular systolic dysfunction overall left ventricular systolic function is severely depressed resting regional wall motion abnormalities there is focal hypokinesis of the apical free wall of the right ventricle the ascending aorta is mildly dilated the aortic valve leaflets are moderately thickened mild aortic regurgitation is seen the mitral valve leaflets are mildly thickened moderate to severe mitral regurgitation is seen the tricuspid valve leaflets are mildly thickened there is moderate pulmonary artery systolic hypertension there is a trivial physiologic pericardial effusion ef of ct head impression no intra or extra axial hemorrhage infarct or mass effect stable appearance of an ill defined area of low attenuation within the right temporal lobe corresponding to a cystic lesion with surrounding edema seen on the previous mri examinations recommend follow up mri with contrast to monitor lesion size characteristics chronic microvascular infarction and lacunar infarction ischemia continue asa nitro gtt heparin gtt lipitor to will consider addition of integrillin and plavix however if remains stable will hold off will need to check with pharmacy if there is a hd dosing of integrillin holding beta blocker given 1st degree av delay and bradycardia ekg findings concerning for antero septal infero ischemia infarct high lad npo after midnight for possible cath in am continue cycling enzymes if chest pain recurs and persists or enzymes rise again will need to consider earlier cath pump last echo suggestive of diastolic dysfunction if becomes hemodynamically unstable will need to consider pacer wire vs pharmacotherapeutics dopamine isoproterenol etc continue amiodarone presumably for afib esrd hd will need dialysis in am recheck lytes now given asterixis concerning for uremia hd completed before cath chest pain improves on nitro heparin drip pt transfered to micu echo reveal ef percent pt c o left hand pain pt bp increased pressor initiated bp drops sbp despite pressors has increase

ICD-9: 427.31 414.01 414.01

Fig. 7. Use cases for attribute-aware approach for explainability. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

6.4. Explainability (RQ3)

We propose two interpretable methods by mapping the output of KGENet to attributes and relations in the disease knowledge graph.

Specifically, in the HAM output, for each ICD code, we compare the words or phrases with the highest attention weights to the ICD attributes to provide an explainable basis. We refer to this method as the **attribute-aware** approach. Fig. 7 shows the explanatory result of a case using attribute-aware as the method. '427.31', '414.01', and '518.81' represent 'atrial fibrillation', 'coronary atherosclerosis', and 'acute respiratory failure', respectively, in the ICD-9 code. We used color to display several words with the highest attention weight in the EHR text. The experimental results show that in most of the test samples, the attribute-aware method can effectively extract keywords associated with ICD codes in EHR.

However, some rare diseases have no or insufficient attributes. The attributes of other diseases are not obvious in the EHR and need to be discovered through associated diseases such as their causes and comorbidities. KGENet realizes the correct identification of the above diseases through the GPM. To this end, we propose a **relationship-augmented** approach that complements the lack of explainability of the attribute-aware method through the relationship graph between diseases. As shown in Fig. 8, KGENet identified three ICD codes '427.31', '428.0', '585.9' through HAM, and '518.83' missing. However, the code can be accurately identified by GPM. The right side of Fig. 8 shows the association between these codes in the disease relation graph. It can be seen that the relationship-augmented method can be a good complement to the attribute-aware method in terms of explainability.

7. Conclusion

In this work, we present a method for constructing a disease knowledge graph that incorporates multi-view attributes and disease relationships. To effectively extract independent and interactive features associated with ICD codes in EHR, we developed a model called KGENet. KGENet comprises two innovative core modules, namely HAM and GPM, which enable attribute-aware and relation-augmented explainable evidence for the prediction results. Our experiments on a benchmark dataset demonstrate that KGENet surpasses the state-of-the-art methods in both accuracy and explainability.

The limitation of our method is that, compared to other baselines, the construction of the disease knowledge graph and the preprocessing of the EHR text require more time. However, we believe this cost

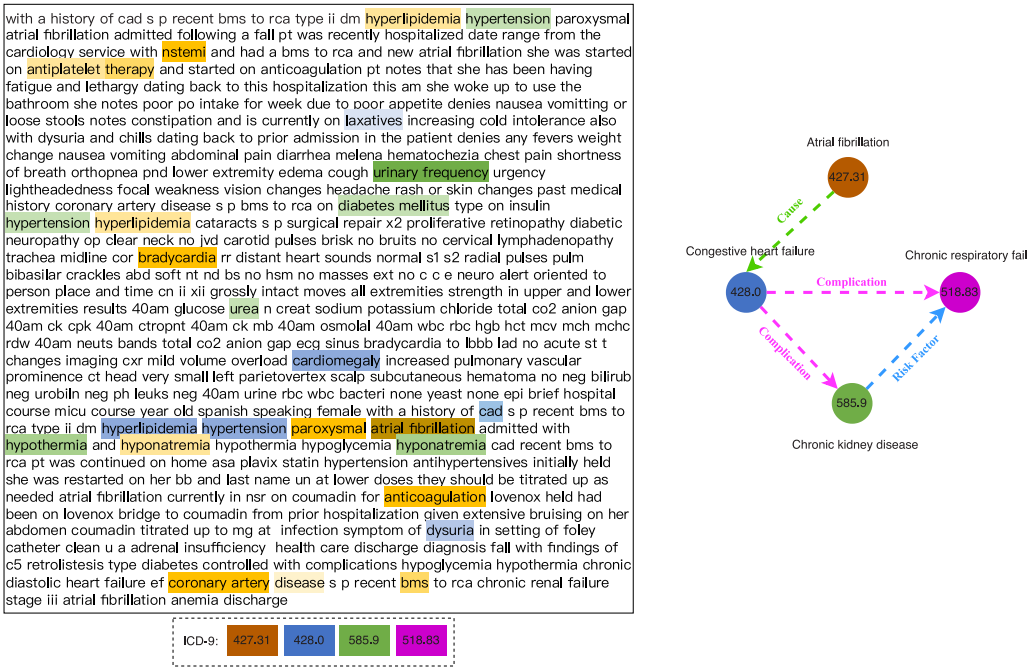

Fig. 8. Use cases for relation-causal approach for explainability.

is worthwhile to improve the performance and explainability of the model.

To ensure the fairness of the comparison experiment with the baseline model, this study uses only unstructured textual data. In future work, we will add structured numerical data, such as laboratory test data and clinical monitoring information, to further improve the performance of the model.

CRediT authorship contribution statement

Tianhan Xu: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Writing – original draft, Writing – review & editing. **Bin Li:** Investigation, Resources, Supervision, Writing – review & editing, Project administration. **Ling Chen:** Investigation, Resources, Supervision, Writing – review & editing. **Chao Yang:** Validation, Writing – review & editing. **Yixun Gu:** Investigation, Resources, Visualization. **Xiang Gu:** Project administration, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This paper is supported by the National Natural Science Foundation of China under Grant No. 61972335 and cross-collaboration between Northern Jiangsu Province People Hospital and Yangzhou University under Grant No. SBJC21002.

References

[1] Yan C, Fu X, Liu X, Zhang Y, Gao Y, Wu J, Li Q. A survey of automated ICD coding: Development, challenges, and applications. Intell Med 2022.
[2] Miotto R, Wang F, Wang S, Jiang X, Dudley JT. Deep learning for healthcare: review, opportunities and challenges. Brief Bioinform 2018;19(6):1236–46.
[3] World Health Organization. International classification of diseases. 2022, URL https://www.who.int/standards/classifications/classification-of-diseases. [Accessed 1 October 2022].
[4] 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 (NAACL-HLT 2018). Association for Computational Linguistics; 2018, p. 1101–11.
[5] Li F, Yu H. ICD coding from clinical text using multi-filter residual convolutional neural network. In: Proceedings of the AAAI conference on artificial intelligence. Vol. 34, 2020, p. 11894–5.
[6] Zhou T, Cao P, Chen Y, Zhu X, Liu Z, Huang M. Automatic ICD coding via interactive shared representation networks with self-distillation mechanism. In: Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing. Online: Association for Computational Linguistics; 2021, p. 5948–57.
[7] Vu T, Nguyen DQ, Nguyen A, Nguyen P, Nguyen TH, Zhang X. A label attention model for ICD coding from clinical text. In: Proceedings of the twenty-ninth international joint conference on artificial intelligence (IJCAI-20). 2020, p. 3285–92.
[8] Dong H, Suárez-Paniagua V, Whiteley W, Ng K, Lu Z, Xie B, Moseley E, Nie Q. Explainable automated coding of clinical notes using hierarchical label-wise attention networks and label embedding initialisation. J Biomed Inform 2021;116:103728.
[9] Biswas B, Pham T-H, Zhang P. TransICD: Transformer based code-wise attention model for explainable ICD coding. In: Tucker A, Henriques Abreu P, Cardoso J, Pereira Rodrigues P, Riaño D, editors. Artificial intelligence in medicine. Cham: Springer International Publishing; 2021, p. 469–78.
[10] Feucht M, Wu Z, Althammer S, Schüller P. Description-based label attention classifier for explainable ICD-9 classification. In: Proceedings of the 2021 EMNLP workshop w-NUT: the seventh workshop on noisy user-generated text. 2021, p. 62–6. http://dx.doi.org/10.18653/v1/2021.wnut-1.8, URL https://aclanthology.org/2021.wnut-1.8.
[11] Liu L, Perez-Concha O, Nguyen A, Xu H, Zhang Y, Li X, Lawley M. Hierarchical label-wise attention transformer model for explainable ICD coding. J Biomed Inform 2022;133:104161.
[12] Wang S, Ren P, Chen Z, Liu X. Few-shot electronic health record coding through graph contrastive learning. IEEE Trans Knowl Data Eng 2021;33(3):1223–36. http://dx.doi.org/10.1109/TKDE.2020.3030742.
[13] Song C, Zhang S, Sadoughi N, Balachandran V. Generalized zero-shot text classification for ICD coding. In: Proceedings of the twenty-ninth international joint conference on artificial intelligence (IJCAI-20). International Joint Conferences on Artificial Intelligence Organization; 2020, p. 2465–71. http://dx.doi.org/10.24963/ijcai.2020/341, URL https://www.ijcai.org/Proceedings/2020/341.
[14] Cao P, Chen Y, Liu K, Li J, Li J, Huang M. HyperCore: Hyperbolic and co-graph representation for automatic ICD coding. In: Proceedings of the 58th annual meeting of the association for computational linguistics. Online: Association for Computational Linguistics; 2020, p. 3105–14.
[15] Wang S, Ren P, Chen Z, Liu Y, Sun M. Coding electronic health records with adversarial reinforcement path generation. In: Proceedings of the 43rd

international ACM SIGIR conference on research and development in information retrieval. ACM; 2020, p. 2483–6.
[16] Xie X, Xiong Y, Yu PS, et al. EHR coding with multi-scale feature attention and structured knowledge graph propagation. In: Proceedings of the 28th ACM international conference on information and knowledge management. ACM; 2019, p. 2331–4.
[17] Teng F, Yang W, Chen L, Zhang H, Zhang H. Explainable prediction of medical codes with knowledge graphs. Front Genet 2020;11:857.
[18] Luo J, Xiao C, Glass L, Sun J, Ma F. Fusion: towards automated ICD coding via feature compression. In: Findings of the association for computational linguistics: ACL-IJCNLP 2021. 2021, p. 2096–101.
[19] Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. In: 5th international conference on learning representations, ICLR 2017, toulon, France, April 24-26, 2017, conference track proceedings. OpenReview.net; 2017, URL https://openreview.net/forum?id=SJU4ayYgl.
[20] Rios A, Kavuluru R. Few-shot and zero-shot multi-label learning for structured label spaces. In: Proceedings of the conference on empirical methods in natural language processing. conference on empirical methods in natural language processing. Vol. 2018, NIH Public Access; 2018, p. 3132.
[21] Wang W, Xu H, Gan Z, Li B, Wang G, Chen L, Yang Q, Wang W, Carin L. Graph-driven generative models for heterogeneous multi-task learning. In: Proceedings of the AAAI conference on artificial intelligence. Vol. 34, 2020, p. 979–88.
[22] Teng F, Liu Y, Li T-J, Zhang Y, Li S, Zhao Y. A review on deep neural networks for ICD coding. IEEE Trans Knowl Data Eng 2023;35:4357–75.
[23] Zou Y, Pesaranghader A, Song Z, Lin S, Sun J. Modeling electronic health record data using an end-to-end knowledge-graph-informed topic model. Sci Rep 2022;12(1):17868.
[24] Yuan Z, Tan C, Huang S. Code synonyms do matter: Multiple synonyms matching network for automatic ICD coding. In: Annual meeting of the association for computational linguistics. 2022, URL https://api.semanticscholar.org/CorpusID:247222710.
[25] National Center for Health Statistics. ICD-9-CM. 2011, URL https://www.cdc.gov/nchs/icd/icd9cm.htm. [Accessed 12 October 2022].
[26] Albano GD, Gagliardo RP, Montalbano AM, Profita M. Overview of the mechanisms of oxidative stress: Impact in inflammation of the airway diseases. Antioxidants 2022;11(11):2237.
[27] Márquez-Nogueras KM, Vuchkovska V, Kuo IY. Calcium signaling in polycystic kidney disease-cell death and survival. Cell Calcium 2023;102733.
[28] National Library of Medicine (US). Unified medical language system. 2023, Available online: https://www.nlm.nih.gov/research/umls/index.html. [Accessed: 28 April 2023].
[29] Clinic M. Bioelectronics neurophysiology and engineering: Gregory A. Worrell. 2022, URL https://www.mayo.edu/research/labs/. [Online Accessed 03 December 2022].
[30] Wikidata. Wikidata query service. 2022, URL https://query.wikidata.org/. [Online Accessed 01 November 2022].
[31] OpenAI. ChatGPT API documentation. 2023, Available online: https://openai.com/api/chat/. [Accessed 28 April 2023].
[32] Wu X, Duan J, Yi P, Li M. Medical knowledge graph: Data sources, construction, reasoning, and applications. Big Data Min Anal 2022.
[33] Shi L, Li S, Yang X, Qi J, Pan G, Zhou B. Semantic health knowledge graph: semantic integration of heterogeneous medical knowledge and services. BioMed Res Int 2017;2017.
[34] Wang M, Zhang J, Liu J, Hu W, Wang S, Li X, Liu W. Pdd graph: Bridging electronic medical records and biomedical knowledge graphs via entity linking. In: The semantic web–ISWC 2017: 16th international semantic web conference, vienna, Austria, October 21-25, 2017, proceedings, part II 16. Springer; 2017, p. 219–27.
[35] Explosion. SpaCy: Industrial-strength natural language processing in Python. 2020, https://spacy.io.
[36] Li Y. Clinical-longformer. 2023, https://huggingface.co/yikuan8/Clinical-Longformer.
[37] Beltagy I, Peters ME, Cohan A. Longformer: The long-document transformer. 2020, ArXiv abs/2004.05150.
[38] Alsentzer E, Murphy J, Boag W, Weng W-H, Jindi D, Naumann T, McDermott M. Publicly available clinical BERT embeddings. In: Proceedings of the 2nd clinical natural language processing workshop. Minneapolis, Minnesota, USA: Association for Computational Linguistics; 2019, p. 72–8. http://dx.doi.org/10.18653/v1/W19-1909, URL https://aclanthology.org/W19-1909.
[39] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. In: Advances in neural information processing systems. 2017, p. 5998–6008.
[40] Li Y, Tarlow D, Brockschmidt M, Zemel RS. Gated graph sequence neural networks. 2015, CoRR abs/1511.05493.
[41] Cho K, van Merrienboer B, Gülçehre Ç, Bahdanau D, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using RNN Encoder–Decoder for statistical machine translation. In: Conference on empirical methods in natural language processing. 2014.
[42] Chen Z-M, Wei X-S, Wang P, Guo Y. Multi-label image recognition with graph convolutional networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 5177–86.
[43] 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, p. 1532–43.
[44] Johnson AE, Pollard TJ, Shen L, Lehman L-wH, Feng M, Ghassemi M, Moody B, Szolovits P, Anthony Celi L, Mark RG. MIMIC-III, a freely accessible critical care database. Sci Data 2016;3(1):1–9.