Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

MedionRep: Medical Ontology Representation using Graph Embedding with Medical Text

Young Hak Kim (mdyhkim@amc.seoul.kr)
University of Ulsan College of Medicine
Jee Eun Song
Korea Advanced Institute of Science and Technology
Tae Joon Jun
Asan Medical Center

Research Article

Keywords: artificial intelligence, electronic health records, electronic medical records, graph embedding, international classification of diseases code, medical concept embedding

Posted Date: March 7th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1419796/v1

License: 🅭 🅯 This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

RESEARCH

MedionRep: Medical Ontology Representation using Graph Embedding with Medical Text

Jee Eun Song[1], Tae Joon Jun[2*] and Young Hak Kim[3*]

[*]Correspondence:
taejoon@amc.seoul.kr;
mdyhkim@amc.seoul.kr
[2]Big Data Research Center of the Asan Institute for Life Sciences, Asan Medical Center, Seoul, Republic of Korea
[3]Division of Cardiology of the department of Internal Medicine, Asan Medical Center and the University of Ulsan College of Medicine, Seoul, Republic of Korea
Full list of author information is available at the end of the article

Abstract

Background: Electronic Medical Record (EMR) is an electronic record of a patient's health information. Nowadays, the importance of EMR analysis is increasing in medical informatics. An EMR is multi-modal data containing medical text and medical concepts. Recent studies attempt to embed either medical text or medical concepts to analyze an EMR partially. However, no type of embedding understands an entire EMR, including both the medical text and the medical concepts. Moreover, most medical concept embeddings train concept sequences from the EMR. These embeddings do not reflect the ontology of the medical concepts, which contain medical semantics. Thus, this study proposes a novel medical ontology representation with medical text for understanding an entire EMR.

Methods: First, we generated the International Classification of Disease (ICD)-10 graph using the ICD-10 graph dataset we created and initialized each node based on the pre-trained medical text embedding. Next, the ICD-10 nodes were trained by the code sequences sampled from the ontology using the encoder-decoder-based graph embedding. Lastly, we trained the ICD-10 nodes by the relation between the ICD code and the text.

Results: For quantitative evaluation, we created similarity pairs of the ICD-10 codes dataset and compared the similarities of the ICD-10 code pairs. The average cosine similarity of ours is 0.87, which is the highest average among the comparison models. We also conducted a comparison of the similarity pairs using open datasets. The pearson correlation coefficients of ours are about 0.461 to 0.463, which is similar to the best model, 0.464. Both comparisons demonstrated that our medical ontology embedding performed well while maintaining the medical text embedding characteristics.

Conclusions: In this study, we proposed a MedionRep which is a medical ontology representation method using graph embedding with medical text. MedionRep is a new medical concept embedding method with medical text and reflect the ontology of the medical concept including medical semantics. Our method could broaden the analysis of the EMR data, which includes several types of medical data.

Keywords: artificial intelligence; electronic health records; electronic medical records; graph embedding; international classification of diseases code; medical concept embedding

Background

As the healthcare industry grows these days, the demand for medical-related artificial intelligence (AI) increases. The importance of Electronic Medical Record (EMR) analysis is also increasing in medical informatics. The EMR is an electronic record of a patient's health information and contains medical history, medication, radiology images, laboratory results, etc. The standardization and sharing of massive EMR data enable a data-driven analysis of EMR using machine learning [1].

Machine learning is one of the data-driven analysis methods. In recent years, massive healthcare AI applications have been proposed such as the prediction of heart failur [2], mortality [3][4][5], sepsis [6][7][8]. There is a process that must be carried out first before creating a healthcare AI application. That is embedding. The embedding is converting the input data to be analyzed into a computer-readable vector format. After the embedding process, learning for tasks such as disease prediction is carried out. That is, embedding has a significant impact on the performance of the model to be made for the task in the future, and it is essential for an excellent model to make a good quality embedding that reflects the meaning of words well. Therefore, to analyze an EMR using data-driven-based deep learning, the first step is to create an embedding that reflects the meaning of EMR data well.

The EMR is multi-modal data containing free-text and many medical concepts (code) related to diseases, drugs, surgical methods, etc. For example, physicians write free-text clinical notes when they diagnose and treat patients. The EMR contains different kinds of the medical concept, including the International Classification of Diseases (ICD), Anatomical Therapeutic Chemical (ATC), and in-house medical concepts used by hospitals. These various medical concepts are used to represent a patient's health and prescription status with free-text notes. Recently, researchers have proposed new embeddings of the medical concepts and medical text embedding separately.

Several studies report medical text embedding. Some studies proposed training the medical corpus (e.g., the PubMed abstract) using an existing embedding method (e.g., Bidirectional Encoder Representations from Transformers (BERT) [9])[10],[11]. Another study employed the modified conventional Word2Vec [12], [13] and added the EMR free-text corpus to improve the medical understanding [14].

Various studies tried to embed the medical concept for their medical AI applications. Some studies try to embed medical concepts by the EMR concept sequence rather than medical concept ontology [15], [16]. Other studies attempted to reflect the ontology of the medical concept for drug-diseases relation extraction [17] and cross-referencing between different medical concepts[18].

Previous studies indicate that conventional medical embeddings can only understand medical textual words or the medical conceptual codes. Thus, analyzing an entire EMR is limited because it contains both free-text and numerous medical concepts. Various medical concepts (i.e., ICD-10 codes, ATC, SNOMED-CT) can be transformed into a knowledge graph called an ontology. Medical concept ontologies are well-refined medical concepts. These ontologies already contain meaningful semantics linking medical terms. Clinically meaningful embedding can be obtained by reflecting the ontology information of the code system rather than embedding a code sequence based on distributed hypothesis [19].

This paper proposes a new medical ontology embedding that understands both the medical text and medical concept. Our proposed embedding reflects the ontology information over the existing medical text-based embedding. In our approach, we initialize the ICD-10 graph using pre-trained medical text embedding and adopt the node2vec embedding method, which reflects the structure of a graph. After the training embedding, we also train the relation between medical concepts and medical text to tighten them.

The significant contributions of this paper are:

- Combining medical concept embeddings without breaking the relationships in the medical text embeddings. Therefore, a new embedding is created where the text and code ontology information can coexist in the same embedding space.
- Reflecting the ontology information to the whole embedding. As training from the knowledge graph of the medical concept, our embedding reflects the medical concept ontology information.
- Providing scalability for the EMR analysis. Unlike the existing methods, medical text and medical concepts are combined as input. With additional information, we can generate more accurate analysis results.

Related Work

In recent years, there have been numerous developed embedding methods based on distributional hypothesis[19], including Word2Vec[12] [13], Glove[20], fasttext[21], BERT [9]. In particular, the Word2Vec [12][13] and BERT[9] methods are widely used in medical embedding. Historically, medical text embedding and medical concept embedding have been developed independently.

Medical Text Embedding

According to [22], embedding trained using clinic notes of the EMR better captures medical semantic than the embeddings trained using medical corpus (e.g., PubMed abstract). Many existing medical text embedding methods attempt to train text in an EMR over a large corpus (e.g., PubMed text). Projection embedding [14] was used to modify Word2Vec [12] [13] to train a small EMR of free-text notes over Wikipedia[1] and PubMed abstract[2] corpora. This method employed Wikipedia for newly emerged diseases (e.g., COVID-19) combined with the EMR for better medical semantic understanding. Moreover, they proposed projection embedding, which is a modified form of Word2Vec. This method trains the EMR using a pre-trained Wikipedia or PubMed embedding projection to compensate for the small amount of data in EMR.

Studies have trained medical data using the BERT embedding method [9]. BioBert [10] trained the Pubmed abstracts and PubMed Central full text[3] using a Wikipedia pre-trained BERT. Clinical BERT [11] is an available pre-trained BERT model that adds the clinic notes from MIMIC-III[23] as the training corpus. These studies are used with medical text embeddings and can only understand medical text words.

Medical Concept Embedding

Medical concept embedding has recently been proposed. Medical concept embedding expresses the medical concept more intensively than plain text. Some studies[15][16]

[1] https://dumps.wikimedia.org/enwiki/

[2] ftp://ftp.ncbi.nlm.nih.gov/PubMed/baseline/

[3] ftp://ftp.ncbi.nlm.nih.gov/pub/pmc/oa_bulk/

embedded the medical codes using Word2Vec except reflecting the ontology of medical code. The medical concept embedding was trained with the ICD code sequence from the EMR using a skip-gram of the Word2vec embedding method. In [18], modified pointwise mutual information (PMI)[24] and modified negative sampling[13] were used to embed medical codes from different ontologies of medical codes.

These studies attempted to embed medical concepts using the modified Word2Vec. Additionally, these studies trained the embedding model using code sequences from the EMR rather than using medical code ontology.

Most medical concepts result from the refinement and classification of diagnosis, medications, laboratory tests, and procedures. These codes can be expressed in a tree or graph structure. The medical semantic is contained in this structure ontology. In this paper, we will embed the code to reflect the inherent semantics of the code ontology.

The previously mentioned studies embed either the medical text or the medical concept (e.g., ICD code). This means that we can input only the free text of the EMR or the medical concept of the EMR to the deep learning model and partially analyze the EMR. However, the EMR is multi-modal data containing both medical text and medical concepts. Therefore, a new embedding method is required to interpret an entire EMR.

Preliminary of Encoder and Decoder based Graph Embedding

In this section, we define the notations and briefly explain the graph embedding, which is based on encoder and decoder. According to [25], encoder and decoder based graph embedding consists of three parts : Encoder function (ENC), Decoder function (DEC), and Similarity function.

Let us assume that $v \in V$, where V is a vertex set, and each v represents a node (i.e., ICD-10 code in this paper). The encoding function maps v_i to d-dimensions in space $\mathbb{R}$. This function reproduces the embedding vector of nodes. Equation (1) represents this concept. Equation (2) defines that z_v as an embedding vector of the node v. Usually, a simple lookup of the embedding matrix is adopted as the encoding function.

$$ENC : V \to \mathbb{R}^d \tag{1}$$

$$z_v = ENC(v) \tag{2}$$

A pairwise decoding function predicts the relationship between the two nodes. For instance, the decoding function can predict the existence of an edge between two nodes. The similarity function represents the ground truth values. In this case, the similarity function is the adjacency matrix. Reducing the gap between decoding function value ($DEC(z_u, z_v)$) and the similarity function value ($Similarity(u, v)$) is the main goal when training the graph embedding.

$$DEC(z_u, z_v) \approx Similarity(u, v) \tag{3}$$

As previously discussed, the lookup of the embedding is primarily used as an encoding function. Therefore, we need to define the decoding function and the similarity function to construct the graph embedding portion of our method.

Methods

In this section, we discuss our proposed scheme in detail. Fig. 1 depicts the overall structure of our method. Our MedionRep consists of three main parts: graph generation, ontology information engraving, and concept-text relation training. First, we transfer the medical concept into a graph data structure for ontology information engraving. Next, MedionRep engraves the ontology information of medical concept into the medical text embedding space. Lastly, we proceed concept-text relation training for tightening the relation between medical concept and its related text words. The rest of this section provides a detailed description of our methodology. We use the ICD-10 code as a medical concept. We used up to four letters of the ICD-10 code except U class and V class. Additionally, we utilize the texts from the note events table from the MIMIC-III clinical database as the EMR data.

Graph Generation of medical Concept

We utilize the ICD-10 codes as a medical concept in this paper. There is no existing graph dataset of ICD-10 and we create a graph dataset of the ICD-10 codes. Fig.2 shows a visual form of the ICD-10 graph dataset. From the root downward, the ICD-10 ontology is divided into large-class nodes (e.g., A00-B99), medium-class nodes(e.g., A00-A09), diagnosis nodes(e.g., A00,A02) and detailed-diagnosis nodes(e.g., A00.0, A01.0). The medium-class can be several layers. For example, the M40-M54 class is divided into M40-M43, M45-M49, and M50-M54. The C00-C97 class is divided into the C00-C75, C76-C80, C81-C96, and the C00-C75 class also is split into the C00-C14, C15-C26, and so on. All nodes except diagnosis nodes and detailed-diagnosis nodes are classifiers, and these nodes represent which category the diagnosis node and the detailed-diagnosis node belong to. Both Diagnosis nodes and detailed-diagnosis nodes are mainly used in a medical situation. We construct an ICD-10 graph using the networX package [26] with the attribute "name" to represent each node's literal name. For instance, node C01 has the attribute "name = Malignant neoplasm of base of tongue". There are 10604 nodes in our ICD10 graph.

Ontology Information Engraving

The left dotted box in Fig.1 represents the ontology information engraving training steps. First, the node vectors are initialized to the average vector of each word in the attribute name using a pre-trained Pubmed+EMR embedding. The initial value is based on the value of the pre-trained embedding to prevent the code from being embedded in the ontology method only. This is to ensure that ontology embedding works well with the pre-trained medical text embedding.

Node sampling for preparing the training sets is a key factor in reflecting the ontology information in graph embedding. This paper adopts the sampling technique from the node2vec embedding method [27], which is a random walk-based methods for reflecting a graph structure into a graph embedding. The node2vec embedding method employs the return parameter p and in-out parameter q. The return parameter p is defined as the degree of returning to the initial node, and the in-out q is defined as the degree of moving to other nodes from the initial node. The proper parameters p and q depend on the size of graph, characteristics of the graph. The optimal parameters p and q are obtained by grid search over some set of p, q.

Our proposed method samples nodes in two phases. One of the most characteristic features of the ICD10 graph form is that about 80% of the nodes are located in leaf nodes. In this case, if the entire graph is sampled with single pair of parameter p and q without dividing the phase, we found that ontology learning for detailed diagnostic nodes is hardly performed. Therefore, two-phase sampling is performed by dividing the graph between the diagnostic node and the detailed diagnostic node layer. Let the entire ICD-10 graph be visualized (shown in Fig.2) be G and let the sub-graph from the root to 3 letter nodes (e.g., A01, and A02) be G'. In the first phase, we set the q be greater than the p for broad exploration like a breadth-first search of the G'. In our model, we set the p to 0.5 and the q to 2.0 for the G'. These p and q values demonstrated the best performance in our experiments. In the second phase, we set the p to be extremely larger than the q to learn the entire graph ontology information by inducing a graph a depth-first search. We set the p to 10^5 and the q to 10^{-5} for the G. Next, we first train the samples from G' and then train the samples from G.

As discussed in previous section, we need to define the encoding function, decoding function, and similarity function to embed the nodes of the ICD-10 graph. The lookup of the embedding matrix is used as the encoding function. We adopt decoding and similarity functions from the node2vec embedding method [27] to reflect the structural information to the embedding and well-training of the sampled node sequences. The main concept of the node2vec metric is the probability that a target node u appears from the starting node v in the random walk of length w. The decoding function and similarity function for this method are expressed in (4) and (5), respectively.

$$DEC(z_u, z_v) = \frac{e^{z_u^T z_v}}{\sum_{k \in V} e^{z_u^T z_k}} \tag{4}$$

$$Similarity(u, v) = P_w(u|v) \tag{5}$$

Fig. 3 shows that the visualization of the ICD-10 nodes in two cases: after initializing weights and before training the graph embedding (left figure), and after training the graph embedding (right figure). In Fig. 3, the target node is the C22 node and the rest of the nodes represent the 20 most similar ICD-10 nodes of the target node, C22. In the left figure of Fig.3 there are unrelated codes (e.g., D135, D376, K824, and Q44). After training ontology embedding, there are the associated node with C22, including C23, C261, C19, and C154 shown in the right figure of Fig. 3,

Concept-Text Relation Training

To combine medical concept embedding with medical text embedding, we desinged two modules: the average initializing, and the concept-text relation training. The average initializing was performed based on the existing pre-trained medical text embedding. The concept-text relation training will be preceded in this step.

In concept-text relation training part, we train the relation between the medical concepts and the medical text for better performance. Fig.4 illustrates this procedure. A concatenation of a medical concept and its literal name is considered to be a sentence for the medical concept-text relation training. Next, we extract only positive samples with the window size set to the maximum length of the sentences. Finally, we train these samples using an Adam optimizer. In this learning phase, the specific vectors of the medical concepts and the medical text are simultaneously updated. This step tightens the relationship between the concepts and the texts in our embedding.

Lastly, we set the α parameter, which adjusts the degree of the reflection of the text-based embedding (W_i in Fig.1) and the ontology embedding (W_f in Fig.1). We can obtain the final embedding W by adjusting α based on the EMR to be analyzed. Equation 6 shows the calculation of the final embedding W.

$$W = (1 - \alpha) * W_i + \alpha * W_f \tag{6}$$

The relationship between the concept and the text gets looser as α approaches 0. The relationship between the concept and the text gets stronger as α approaches 1. Finally, we obtain the final embedding W that can understand both the medical concepts and the medical text.

Results ans Experiments

In this section, we qualitatively and quantitatively evaluate our method. We create the similarity pairs of the ICD-10 codes, which are examined by cardiologists from the Seoul Asan Medical Center. We compute the cosine similarity of these pairs and compare the similarity pairs with open datasets (e.g., Mayo and UMNSRS) for the quantitative evaluation. In the qualitative evaluation, we visualize the results of the 50 most similar words for the specific ICD-10 codes.

Basic Word2Vec-based embeddings and projection-based embedding are used as the comparison groups. We use Wikipedia and PubMed for the base embedding training. According to [14], training a word embedding with an EMR to understand the medical semantic is better than only-training with a large corpora, such as Wikipedia and PubMed. We also use the free-text notes from the note event table of the MIMIC-III clinical database v1.4[23] as the EMR data.

For basic Word2Vec embedding[12] [13], we train using Wikipedia, PubMed abstracts, notes from MIMIC-III[23], and a combination of these three corpora. In medical NLP, embedding is rarely used except for EMR. Therefore, we select the ”Wiki+EMR”, ”Pubmed+EMR” and ”Wiki+Pubmed+EMR” combinations for comparison. For the projection-based embedding, we use the Wikipedia corpus as the base embedding because it shows better performance than the PubMed-based and the Wikipedia+PubMed based. Then, the projection-based embedding is trained using notes from the MIMIC-III database [23].

The training setting of the basic Word2vec[12] [13], projection-based embedding, [14], and base embedding of our proposed method are applied using an identical technique for precise comparison. The settings are as follow: skip-gram, 12 window sizes, 50 dimensions, 20 minimum word frequency, 5 negative samplings, and 0.0001 learning rate. We use a Gensim [28] for the basic Word2Vec training and implement a projection-based method using Keras [29]. We utilize the networkX [26] and StellarGraph [30] Python packages to implement our method. [14]

ICD-10 Pair Comparison

A new dataset that contains clinically relevant concept pairs (i.e., the ICD-10 code pair in this paper) is required to evaluate the quantitative performance of medical concept embedding. We create our own set of similar ICD-10 code pairs verified by cardiologists from the Seoul Asan Medical Center. This dataset includes 43 pairs of similar or related ICD-10 codes. We calculate the cosine similarities for all pairs in this dataset and calculate the average. The ICD-10 codes cannot be directly understood by the Word2Vec and projection text embeddings. Therefore, the average vector of each ICD-10 code literal name is used as the input. In contrast, our proposed embedding directly uses the ICD-10 code as input.

Table 1 details the cosine similarity scores of the selected ICD-10 pairs. Table 2 shows the average of all the ICD-10 pairs cosine similarity scores for all of the embeddings. Overall, our proposed method (hereafter referred to as MedionRep) outperformed the Word2Vec based embeddings and the projection embedding. As α increases, MedionRep shows better performance. This is because the α represents the degree of the reflection of the ontology embedding or text embedding. The higher the α, the higher the reflection of the ontology embedding. We observe the best performance in this evaluation when α is close to 0.9.

For ICD-10 pairs located close in the ontology, MedionRep shows higher similarity than the Word2Vec based embedding and the projection embedding. I10-I20, I20-I21, I21-I25, I61-I63, and J30-J32 are closely located in the ICD-10 ontology and show the highest score among the counter embeddings. Most of the scores are greater than 0.95. However, the Word2Vec based embeddings show poor performance (e.g., I10-I20 and I20-I21 from "PubMed+EMR") even though the two medical concepts are similar. Projection embedding shows inconsistent performances depending on the ICD-10 pair. Projection embedding show poor performance at I10-I20 and good performance at I20-I21, I21-I25, I61-I63, and J30-J32.

In the ICD-10 pairs located a little further in the ontology, MedionRep also shows good performance for the I10-I63, I10-I70, I20-I63, and I20-I70 pairs, which have different parent nodes. Most of the scores are close to 0.83 because the two terms are not closely located. Even though these are lower scores than the previous case (e.g., 0.9545 for I20-I21), our method shows improvement compared to the Word2Vec based and the projection-based embeddings.

In the I25-I70, I61-I70, J32-J45, and N18-N30 ICD-10 pairs, projection-based embedding outperforms MedionRep. The MedionRep method scores less when the two terms are located further apart. When the words constituting the term are similar, or the co-occurrence ratio is high, the projection embedding method shows the best performance in the medical text embedding and can generate a better similarity than MedionRep. If there is a little distance between the two terms on the ontology, our proposed method comes out lower than the projection. The difference between the projection embedding and our MedionRep method is only about 0.02. Our MedionRep method also treats the two terms which are similar as an absolute number.

For the I21.2-I25, I21.3-I25.5, I25.3-I61, and I25.3-I70 pairs, our MedionRep method scores are highest among the counter embeddings. The score is penalized as the number of digits increases and the distance increases. Nevertheless, all of the pairs are treated in similar words. This occurs because the whole graph is divided by the range of the sampling to generate the sub-graph.

In the F20-F32 and F41-F51 pairs, our MedionRep method with an α equal to 0.8 shows the best score. An F class is a disease code group with many sequential occurrences or co-occurrences. In particular, the F20-F32 and F41-F51 pairs are disease codes that frequently co-occur. When frequent simultaneous occurrence, text embedding can be advantageous with frequent simultaneous occurrences. It is advantageous because the case of $\alpha = 0.8$ with a little bit of text embedding shows better performance than the case of $\alpha = 0.9$, which reflects the ontology embedding more than the text embedding.

Open Medical Term Pair Comparision

The relationship between the text word vectors changes during the combining of the medical concept embedding with the existing text embedding. This change often leads to poor performance of the text embedding. Our proposed MedionRep method minimizes this change and this evaluation verifies it.

In a second quantitative evaluation, we compare the similarity between the two medical terms with several open dataset (MayoSRS[31], MiniMayoSRS[32][33],UMNSRS-Relatedness[34], UMNSRS-Similarity[34], UMNSRS-Relatedness-MOD[35], and UMNSRS-Similarity-MOD [35]). These datasets are a set of medical concept pairs with the similarity score determined by medical coders, physicians, and medical residents from the Mayo Clinic and the University of Minnesota Medical School. The higher the similarity score, the higher the similarity between the two medical term pairs. MayoSRS contains medical concept pairs for semantic relatedness. MiniMayoSRS is a subset of the MayoSRS, which contains 29 pairs that are highly scored by raters. UMNSRS-Relatedness contains related Unified Medical Language System (UMLS) concept pairs and UMNSRS-Similarity contains similar UMLS concept pairs. The UMNSRS -Relatedness-MOD and UMNSRS-Similarity-MOD are subsets of the UMNSRS-Relatedness and UMNSRS-Similarity, respectively. UMNSRS-Relatedness-MOD and UMNSRS-Similarity-MOD exclude unrelated clinical and biomedical concept pairs.

We use each pair's similarity score to represent ground truth and calculate the cosine similarity of two medical terms in the datasets. After that, we estimate the Pearson correlation coefficient between the ground truth and the list of calculated cosine similarities.

Table 3 shows the result of this evaluation. In this evaluation, we examine the performance of the text embedding portion. Overall, the average of "Pubmed+EMR" resulted in the best performance with a score of 0.4640. Our proposed MedionRep method with α set to 0.2 also shows similar performance with a score of 0.4636 indicating a minimal difference. In the Mayo series, the "Pubmed+EMR" performs the best. Our MedionRep method performed second best, with a slightly lower score. The lower the α value, the higher the performance in our method. The effect of the ontology embedding decreases when α is lowered, and consequently, the effect of the text embedding increases. In other words, the higher the α, the more affected the ontology embedding. The higher the α, the lower the performance in the Mayo series pairs. This occurs because the embedding of the ontology adversely affects the existing text embedding. However, the proposed MedionRep method minimizes the changes which cause a breaking relationship between the text words with little difference compared to the best performance by "Pubmed+EMR." For the MiniMayoSRS database, we observed that α increases by only using the selected data in all cases.

The UMNSRS databases are sets of text words on a wide range of biomedical and clinic topics divided into similar and related pairs. For the UMNSRS relatedness and similarity databases, our proposed scheme ($\alpha = 0.9$) shows the best performance, 0.4375, and 0.4990 repectively. However "Pubmed+EMR" method also shows similar values of 0.4366, and 0.4970 respectively. From these results, we infer that the proposed MedionRep method shows better performance without breaking the relationship between the text embedding. UMNSRS-Relatedness-Mod and UMNSRS-Similarity-Mod are subsets of the UMNSRS-Relatedness and UMNSRS-Similarity, respectively. Since these databases contain more relevant words than the previous UMNSRS databases, all of the embeddings perform better compared to the UMNSRS-Relatedness and UMNSRS-Similarity databases. In this case, the "Wiki+Pubmed+EMR" shows the best performance, with scores of 0.5036, and 0.5674. The UMNSRS-Relatedness-Mod database shows a difference of about 0.03 compared to our MedionRep method, but the UMNSRS-Similarity-Mod database shows a slightly larger difference of 0.05. Overall, as our proposed MedionRep method shows stable results in all data sets, thus proving that our proposed MedionRep embeds medical concepts without significantly breaking the relationship with medical text embedding.

Finally, we visualize the embedding results from our proposed MedionRep method using Principal Component Analysis (PCA). Fig.5 displays the 50 related nodes of C90, F01, G30, and I50 nodes. The target nodes are shown as the largest node in each embedding space, and the 50 related nodes are shown as smaller nodes than the target node. We observe that embedded nodes are generally well-gathered, and the related nodes are near the target node location.

Discussion

The structured characteristics of the ICD-10 code are different for each classification. For example, the I00- I99 class (i.e., Diseases of the circulatory system), which is one of the largest classes, and there is a tendency for the disease to sequentially progress in the code. It is likely to proceed in the order of I10 (Essential hypertension), → I20 (Angina pectoris), → I25 (Chronic ischaemic heart disease), → I21

(Acute myocardial infarction), → I61 (Intracerebral haemorrhage), → I63 (Cerebral infarction). In the case of F00-F99 (Mental and behavioural disorders), the F22 (Persistent delusional disorders), F51 (Nonorganic sleep disorders), and F32 (Depressive episode) codes are likely to occur simultaneously.

However, even if the code is located in a similar location, it is possible that it is not a similar disease or may not develop at the same time. For the C01-C99 class, C16 (Malignant neoplasm of stomach), C22 (Malignant neoplasm of liver and intrahepatic bile ducts), and C25 (Malignant neoplasm of liver and intrahepatic bile ducts) are close in the codes. These codes are classified as codes in similar locations due to the commonality of the Malignant neoplasm. Moreover, a close location does not mean that these diseases are likely to co-occur or at the same time. Therefore, codes located close to the ontology may be less relevant. Usually, our embedding predicts that it is highly relevant if the code is in a similar position in the code ontology. In the case of the C01-C99 class, our embedding might show degraded performance. Therefore we need to adjust the proper α value for a specific C class for the appropriate application, which is a downstream task if desired by the user.

Our proposed MedionRep method is capable of understanding both the medical text and the medical concepts, as opposed to only understanding either the medical text embedding or the medical concept embedding when analyzing the entire EMR. This work can be extended by applying another medical concept such as NDC and embedding multiple medical concept with medical text into one embedding space. However, there is a limitation of our proposed work. Our MedionRep reflects the ontology of medical concept and considers the codes which are located close in the ontology as similar terms. When the classification criteria are different for each class(e.g., C class and F class mentioned in the previous paragraph), the nuance of the most similar concept judged by our proposed MedionRep may vary from class to class. Therefore, we designed the α parameter to alleviate this limitation. The α parameter adjusts the ratio between medical concept embedding and the medical text embedding. An α value between 0.5 and 0.8 is appropriate for the entire embedding.

Conclusions

This paper proposes a new medical ontology embedding method called MedionRep, which employs graph embedding. Unlike existing methods, the MedionRep embedding method contains two types of data (e.g., medical words and medical codes) for the EMR analysis. Our research enables rich analysis by understanding EMR with both medical text and medical concepts. We both quantitatively and qualitatively evaluate our proposed method. These evaluations verify that our embedding reflects the ICD-10 ontology information to the ICD-10 code embedding and merges with the medical text embedding without breaking the relationship between the medical words. As training from medical concept ontology which is regarded as a knowledge graph, there is an advantage. Our proposed MedionRep can learn objective knowledge of medical concept without the bias of training data (i.e., ICD-10 sequences from EMR, consisting of particular ICD-10 codes) and concern of the number of training data. However, there is a limitation of our work. Since codes in close positions in the ontology are viewed as similar words, the nuances of similar words may differ for each class if the classification criteria are changed for each class when defining the ontology. Several future studies remain in this research. First, the embedding of several medical concepts (e.g., ICD-10 code and NDC code) into a single embedding space may be a future task. Finally, application verification for our proposed MedionRep embedding method remains for a follow-up study.

Acknowledgements
Not applicable. . .

Funding
This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (2021-0-00982, Development of the multi-center distributed intelligence reinforcement federation technology for cooperating optimal treatments, 50%) and by a grant of the Korea Health Technology R&D Project through the Korea Health Industry Development Institute(KHIDI), funded by the Ministry of Health & Welfare, Republic of Korea (grant number : HR21C0198, 50%). . .

Abbreviations
AI: artificial intelligence ATC : anatomical therapeutic chemical DEC : decoder function EHR: electronic health record EMR: electronic medical record ENC: encoder function ICD: international classification of disease ML: machine learning PCA: principal component analysis UMLS: unified medical language system . . .

Availability of data and materials
The Wikipedia text dataset analysed during the current study are available in the https://dumps.wikimedia.org/enwiki/. The Pubmed abstract text dataset analysed during the current study are available in the ftp://ftp.ncbi.nlm.nih.gov/PubMed/baseline/. The MIMIC-3 dataset analysed during the current study are available in the ftp://ftp.ncbi.nlm.nih.gov/pub/pmc/oa bulk/. The datasets (ICD-10 graph dataset and ICD-10 similar pairs dataset) generated during the current study are available from the corresponding author on reasonable request.. . .

Ethics approval and consent to participate
Not applicable. . .

Competing interests
The authors declare that they have no competing interests.

Consent for publication
Not applicable. . .

Authors' contributions
JES conceive the study and performed the experiment and wrote the draft manuscript. TJJ and YHK analyzed and interpreted the experiment result and reviewed the manuscript. All authors read and approved the final manuscript. . . .

Author details
[1]School of Computing, Korea Advanced Institute of Science and Technology, Daejeon, Republic of Korea. [2]Big Data Research Center of the Asan Institute for Life Sciences, Asan Medical Center, Seoul, Republic of Korea. [3]Division of Cardiology of the department of Internal Medicine, Asan Medical Center and the University of Ulsan College of Medicine, Seoul, Republic of Korea.

References

1. Hecht, J.: The future of electronic health records. Nature **573**(7775), 114–114 (2019)
2. Choi, E., Bahadori, M.T., Kulas, J.A., Schuetz, A., Stewart, W.F., Sun, J.: Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. arXiv preprint arXiv:1608.05745 (2016)
3. Theis, J., Galanter, W., Boyd, A., Darabi, H.: Improving the in-hospital mortality prediction of diabetes icu patients using a process mining/deep learning architecture. IEEE Journal of Biomedical and Health Informatics (2021)
4. Soffer, S., Klang, E., Barash, Y., Grossman, E., Zimlichman, E.: Predicting in-hospital mortality at admission to the medical ward: A big-data machine learning model. The American journal of medicine **134**(2), 227–234 (2021)
5. Estiri, H., Strasser, Z.H., Klann, J.G., Naseri, P., Wagholikar, K.B., Murphy, S.N.: Predicting covid-19 mortality with electronic medical records. NPJ digital medicine **4**(1), 1–10 (2021)
6. Goh, K.H., Wang, L., Yeow, A.Y.K., Poh, H., Li, K., Yeow, J.J.L., Tan, G.Y.H.: Artificial intelligence in sepsis early prediction and diagnosis using unstructured data in healthcare. Nature communications **12**(1), 1–10 (2021)
7. Shashikumar, S.P., Josef, C.S., Sharma, A., Nemati, S.: Deepaise–an interpretable and recurrent neural survival model for early prediction of sepsis. Artificial Intelligence in Medicine **113**, 102036 (2021)
8. Qin, F., Madan, V., Ratan, U., Karnin, Z., Kapoor, V., Bhatia, P., Kass-Hout, T.: Improving early sepsis prediction with multi modal learning. arXiv preprint arXiv:2107.11094 (2021)

9. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (2019). doi:10.18653/v1/N19-1423. https://www.aclweb.org/anthology/N19-1423
10. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C.H., Kang, J.: Biobert: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics **36**(4), 1234–1240 (2020)
11. 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, pp. 72–78 (2019)
12. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
13. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2. NIPS'13, pp. 3111–3119. Curran Associates Inc., Red Hook, NY, USA (2013)
14. Lin, C., Lou, Y.-S., Tsai, D.-J., Lee, C.-C., Hsu, C.-J., Wu, D.-C., Wang, M.-C., Fang, W.-H.: Projection word embedding model with hybrid sampling training for classifying icd-10-cm codes: longitudinal observational study. JMIR medical informatics **7**(3), 14499 (2019)
15. Choi, E., Schuetz, A., Stewart, W.F., Sun, J.: Medical concept representation learning from electronic health records and its application on heart failure prediction. arXiv preprint arXiv:1602.03686 (2016)
16. Nguyen, D., Luo, W., Venkatesh, S., Phung, D.: Effective identification of similar patients through sequential matching over icd code embedding. Journal of medical systems **42**(5), 1–13 (2018)
17. Xuan, P., Gao, L., Sheng, N., Zhang, T., Nakaguchi, T.: Graph convolutional autoencoder and fully-connected autoencoder with attention mechanism based method for predicting drug-disease associations. IEEE Journal of Biomedical and Health Informatics (2020)
18. Bai, T., Egleston, B.L., Bleicher, R., Vucetic, S.: Medical concept representation learning from multi-source data. In: IJCAI: Proceedings of the Conference, vol. 2019, p. 4897 (2019). NIH Public Access
19. Harris, Z.S.: Distributional structure. Word **10**(2-3), 146–162 (1954)
20. Pennington, J., Socher, R., Manning, C.: GloVe: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543. Association for Computational Linguistics, Doha, Qatar (2014). doi:10.3115/v1/D14-1162. https://www.aclweb.org/anthology/D14-1162
21. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics **5**, 135–146 (2017)
22. Wang, Y., Liu, S., Afzal, N., Rastegar-Mojarad, M., Wang, L., Shen, F., Kingsbury, P., Liu, H.: A comparison of word embeddings for the biomedical natural language processing. Journal of biomedical informatics **87**, 12–20 (2018)
23. Johnson, A.E., Pollard, T.J., Shen, L., Li-Wei, H.L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L.A., Mark, R.G.: Mimic-iii, a freely accessible critical care database. Scientific data **3**(1), 1–9 (2016)
24. Turney, P.D., Pantel, P.: From frequency to meaning: Vector space models of semantics. Journal of artificial intelligence research **37**, 141–188 (2010)
25. Hamilton, W.L., Ying, R., Leskovec, J.: Representation learning on graphs: Methods and applications. arXiv preprint arXiv:1709.05584 (2017)
26. Hagberg, A., Swart, P., S Chult, D.: Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States) (2008)
27. Grover, A., Leskovec, J.: node2vec: Scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 855–864 (2016)
28. Řehůřek, R., Sojka, P.: Software Framework for Topic Modelling with Large Corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, pp. 45–50. ELRA, Valletta, Malta (2010). http://is.muni.cz/publication/884893/en
29. Chollet, F., et al.: Keras. https://keras.io (2015)
30. Data61, C.: StellarGraph Machine Learning Library. GitHub (2018)
31. Pakhomov, S.V., Pedersen, T., McInnes, B., Melton, G.B., Ruggieri, A., Chute, C.G.: Towards a framework for developing semantic relatedness reference standards. Journal of biomedical informatics **44**(2), 251–265 (2011)
32. Pedersen, T., Pakhomov, S.V., Patwardhan, S., Chute, C.G.: Measures of semantic similarity and relatedness in the biomedical domain. Journal of biomedical informatics **40**(3), 288–299 (2007)
33. McInnes, B.T., Pedersen, T., Pakhomov, S.V.: Umls-interface and umls-similarity: open source software for measuring paths and semantic similarity. In: AMIA Annual Symposium Proceedings, vol. 2009, p. 431 (2009). American Medical Informatics Association
34. Pakhomov, S., McInnes, B., Adam, T., Liu, Y., Pedersen, T., Melton, G.B.: Semantic similarity and relatedness between clinical terms: an experimental study. In: AMIA Annual Symposium Proceedings, vol. 2010, p. 572 (2010). American Medical Informatics Association
35. Pakhomov, S.V., Finley, G., McEwan, R., Wang, Y., Melton, G.B.: Corpus domain effects on distributional semantic modeling of medical terms. Bioinformatics **32**(23), 3635–3644 (2016)

Figures

Figure 1 Overall structure of the proposed method

Figure 2 ICD-10 ontology visualization using our ICD-10 graph dataset

Figure 3 Visualization of the ICD-10 nodes after node initializing in embedding space R_i (left) and ICD-10 nodes after training ontology information engraving in embedding space R_t (right) using Principal Component Analysis (PCA). The C22 node represents a target node. The rest of the nodes indicate the 20 most similar nodes of the target node. There are unrelated nodes (e.g., D376, K824) shown in the left figure. After training the ontology embedding, the only related nodes are selected as the 20 most similar nodes shown in the right figure.

Figure 4 Sampling and training Procedure of the concept and text relation training

Figure 5 Visualization of target code's 50 most similar ICD-10 codes using Principal Component Analysis (PCA). The target nodes (i.e., C90, F01, G30, and I50) were randomly

Tables

Table 1 Extraction of the 5 most similar words to each input code using word embedding models

ICD Term pair		Word2Vec			Projection	Ours			
		Wiki+EMR	Pubmed+EMR	Wiki+Pubmed+EMR		Ours a = 0.9	Ours a = 0.8	Ours a = 0.5	Ours a = 0.2
I10 Essential(primary) hypertension	I20 Angina pectoris	0.617237	0.35647	0.527157	0.664072	**0.857427**	0.84803	0.794366	0.632776
I10 Essential(primary) hypertension	I63 Cerebral infarction	0.534124	0.381881	0.542013	0.727649	**0.834249**	0.825144	0.774667	0.626551
I10 Essential(primary) hypertension	I70 Atherosclerosis	0.61906	0.191216	0.554215	0.760461	**0.817353**	0.801271	0.716453	0.500507
I20 Angina pectoris	I21 Acute myocardial infarction	0.695868	0.482994	0.640412	0.884295	**0.95455**	0.946274	0.896874	0.744606
I20 Angina pectoris	I63 Cerebral infarction	0.498423	0.250978	0.432017	0.80553	**0.83082**	0.81688	0.741139	0.538369
I20 Angina pectoris	I70 Atherosclerosis	0.631687	0.409095	0.555519	0.783266	**0.837473**	0.824972	0.759305	0.602404
I21 Acute myocardial infarction	I25 Chronic ischaemic heart disease	0.756248	0.677981	0.719612	0.94566	**0.977777**	0.97492	0.955095	0.876022
I21.2 Acute transmural myocardial infarction of other sites	I25 Chronic ischaemic heart disease	0.828314	0.761725	0.815618	0.911540	**0.928121**	0.924283	0.904688	0.854007
I21.3 Acute transmural myocardial infarction of unspecified site	I25.5 Ischaemic cardiomyopathy	0.799282	0.705931	0.793451	0.881932	**0.912833**	0.903654	0.85909	0.760811
I25.3 Aneurysm of heart	I61 Intracerebral haemorrhage	0.686284	0.548904	0.668795	0.796803	**0.834401**	0.830469	0.806502	0.733664
I25 Chronic ischaemic heart disease	I70 Atherosclerosis	0.60235	0.28073	0.571466	**0.879232**	0.835606	0.823055	0.754345	0.570502
I25.3 Aneurysm of heart	I70 Atherosclerosis	0.634291	0.265740	0.571397	0.656353	**0.799416**	0.786065	0.712326	0.52376
I61 Intracerebral haemorrhage	I63 Cerebral infarction	0.828573	0.751647	0.812114	0.878434	**0.966855**	0.963233	0.941255	0.872185
I61 Intracerebral haemorrhage	I70 Atherosclerosis	0.591873	0.217485	0.562741	**0.787886**	0.75507	0.736968	0.647324	0.449824
F20 Schizophrenia	F32 Depressive episode	0.734625	0.669657	0.718826	0.802276	0.861834	**0.862417**	0.857068	0.808702
F20 Schizophrenia	F51 Nonorganic sleep disorders	0.701571	0.568012	0.754401	0.824845	**0.860941**	0.858718	0.840104	0.761067
F32 Depressive episode	F41 Other anxiety disorders	0.784391	0.642887	0.778016	0.824857	**0.870721**	0.868572	0.853463	0.792601
F32 Depressive episode	F51 Nonorganic sleep disorders	0.763378	0.621002	0.835819	0.746794	**0.851236**	0.848647	0.831526	0.767373
F41 Other anxiety disorders	F51 Nonorganic sleep disorders	0.876281	0.755774	0.867316	0.896847	0.89678	**0.896923**	0.893622	0.863354
J30 Vasomotor and allergic rhinitis	J32 Chronic sinusitis	0.75712	0.562256	0.695509	0.927679	**0.96898**	0.964006	0.932079	0.812552
J31 Chronic rhinitis, nasopharyngitis and pharyngitis	J32 Chronic sinusitis	0.812331	0.675377	0.788267	0.968676	**0.978166**	0.980795	0.9807	0.98056
J32 Chronic sinusitis	J45 Asthma	0.71763	0.601529	0.673234	**0.865178**	0.853692	0.855888	0.856019	0.856088
N17 Acute renal failure	N30 Cystitis	0.616321	0.424285	0.55692	0.805435	**0.855102**	0.84871	0.807889	0.672204
N18 Chronic kidney disease	N30 Cystitis	0.713509	0.490069	0.649639	**0.861052**	0.849878	0.844305	0.809991	0.700476

Table 2 Average of cosine similarities for similar ICD10 code pairs set we created

	Word2Vec			Projection	Ours			
	Wiki+EMR	PubMed+EMR	Wiki+PubMed+EMR		Ours a = 0.9	Ours a = 0.8	Ours a = 0.5	Ours a = 0.2
Similar ICD-10 word pairs								
Average of cosine similarities	0.711778	0.534551	0.682194	0.855268	**0.870578**	0.864785	0.830749	0.725989

Table 3 Pearson correlation coefficient between the similarity value and the list of cosine similarities of two medical pair calculated with word embedding models

	Word2Vec			Projection	Ours			
	Wiki+EMR	Pubmed+EMR	Wiki+Pubmed+EMR		Ours $\alpha = 0.9$	Ours $\alpha = 0.8$	Ours $\alpha = 0.5$	Ours $\alpha = 0.2$
Mayo								
MayoSRS	0.219939	**0.322944**	0.280534	0.015373	0.308711	0.310332	0.315164	0.319893
MiniMayoSRS	0.427627	**0.541682**	0.454335	0.223404	0.533470	0.534462	0.537329	0.540009
UMNSRS								
UMNSRS_Relatedness	0.320192	0.436663	0.420611	0.211006	**0.437590**	0.437586	0.437442	0.437066
UMNSRS_Relatedness_Mod	0.408632	0.472277	**0.503658**	0.212519	0.473744	0.473711	0.473435	0.472858
UMNSRS_Similarity	0.387678	0.497058	0.485515	0.157698	**0.499044**	0.498889	0.498333	0.497626
UMNSRS_Similarity_Mod	0.466346	0.513683	**0.567419**	0.141235	0.516561	0.516323	0.515497	0.514481
Average	0.371736	**0.464051**	0.452012	0.160206	0.461520	0.461884	0.462867	**0.463656**

Figures

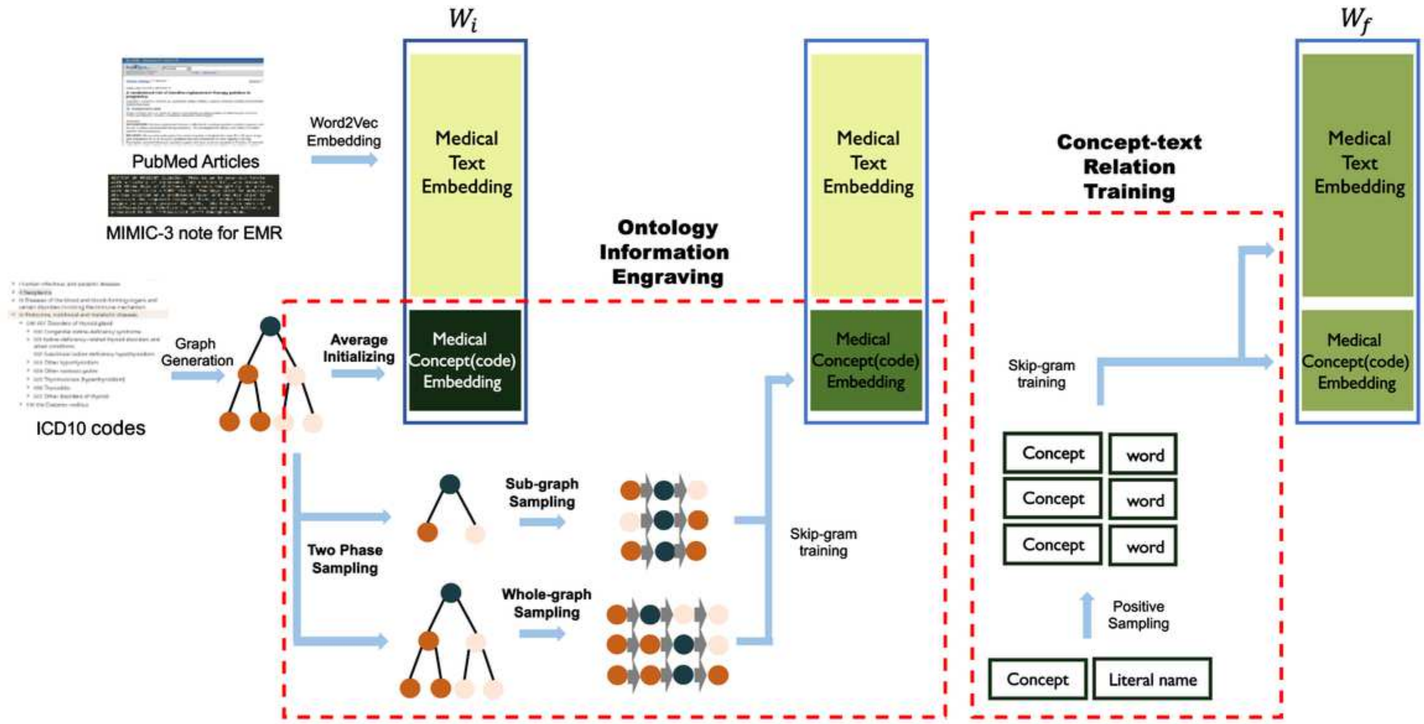


Figure 1

Overall structure of the proposed method

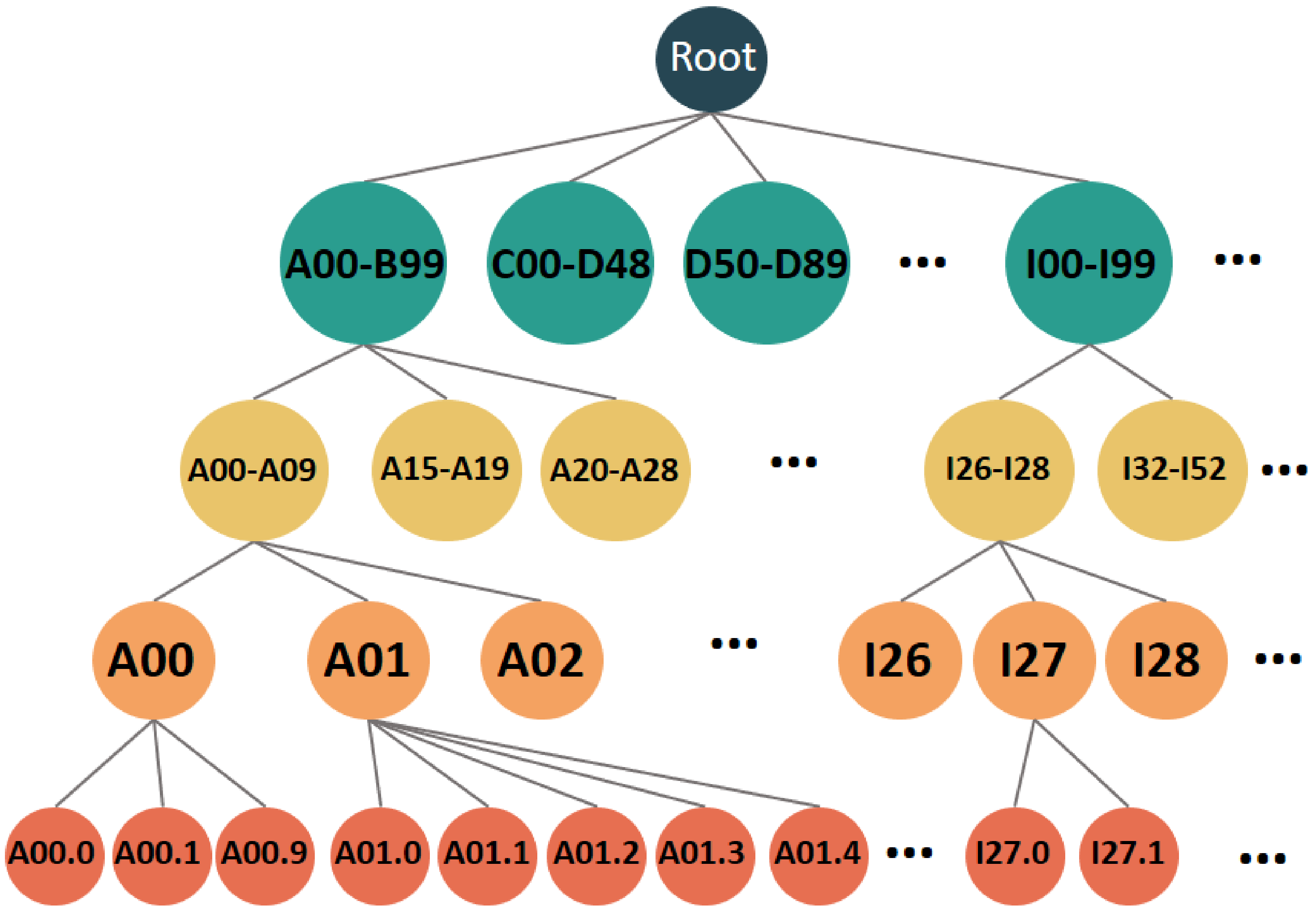


Figure 2

ICD-10 ontology visualization using our ICD-10 graph dataset

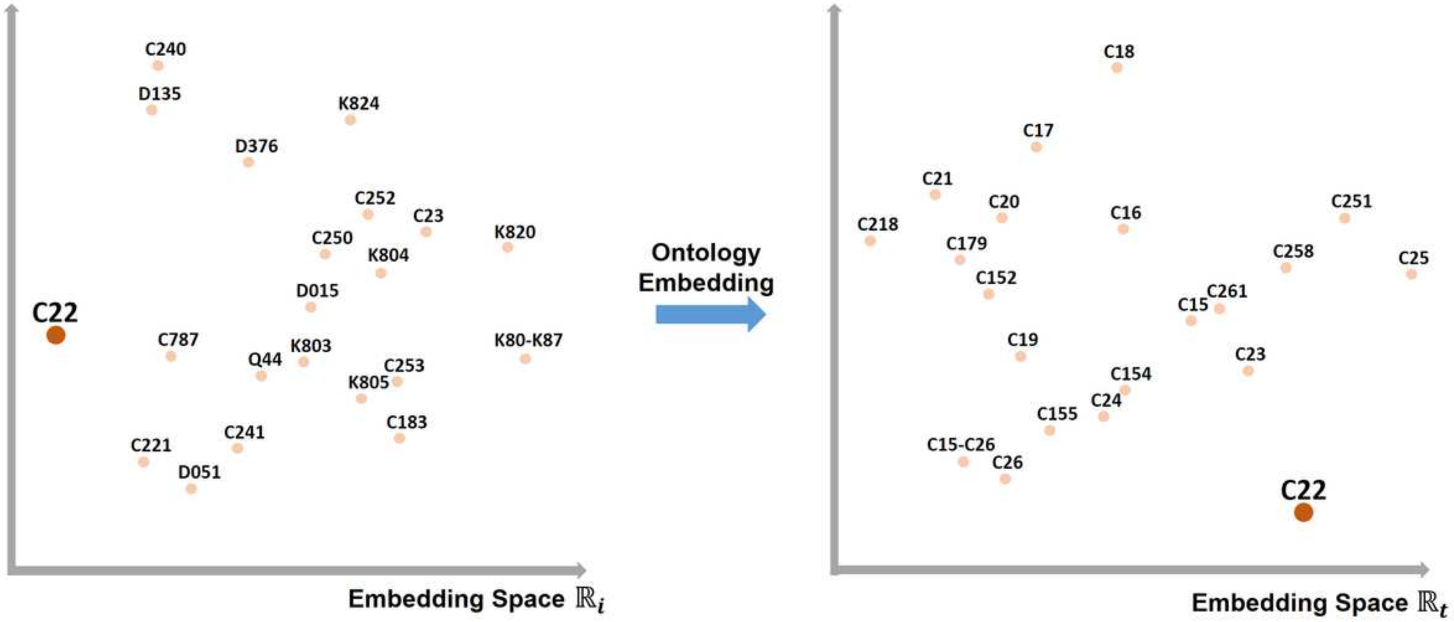


Figure 3

Visualization of the ICD-10 nodes after node initializing in embedding space Ri (left) and ICD-10 nodes after training ontology information engraving in embedding space Rt (right) using Principal Component Analysis (PCA). The C22 node represents a target node. The rest of the nodes indicate the 20 most similar nodes of the target node. There are unrelated nodes (e.g., D376, K824) shown in the left figure. After training the ontology embedding, the only related nodes are selected as the 20 most similar nodes shown in the right figure.

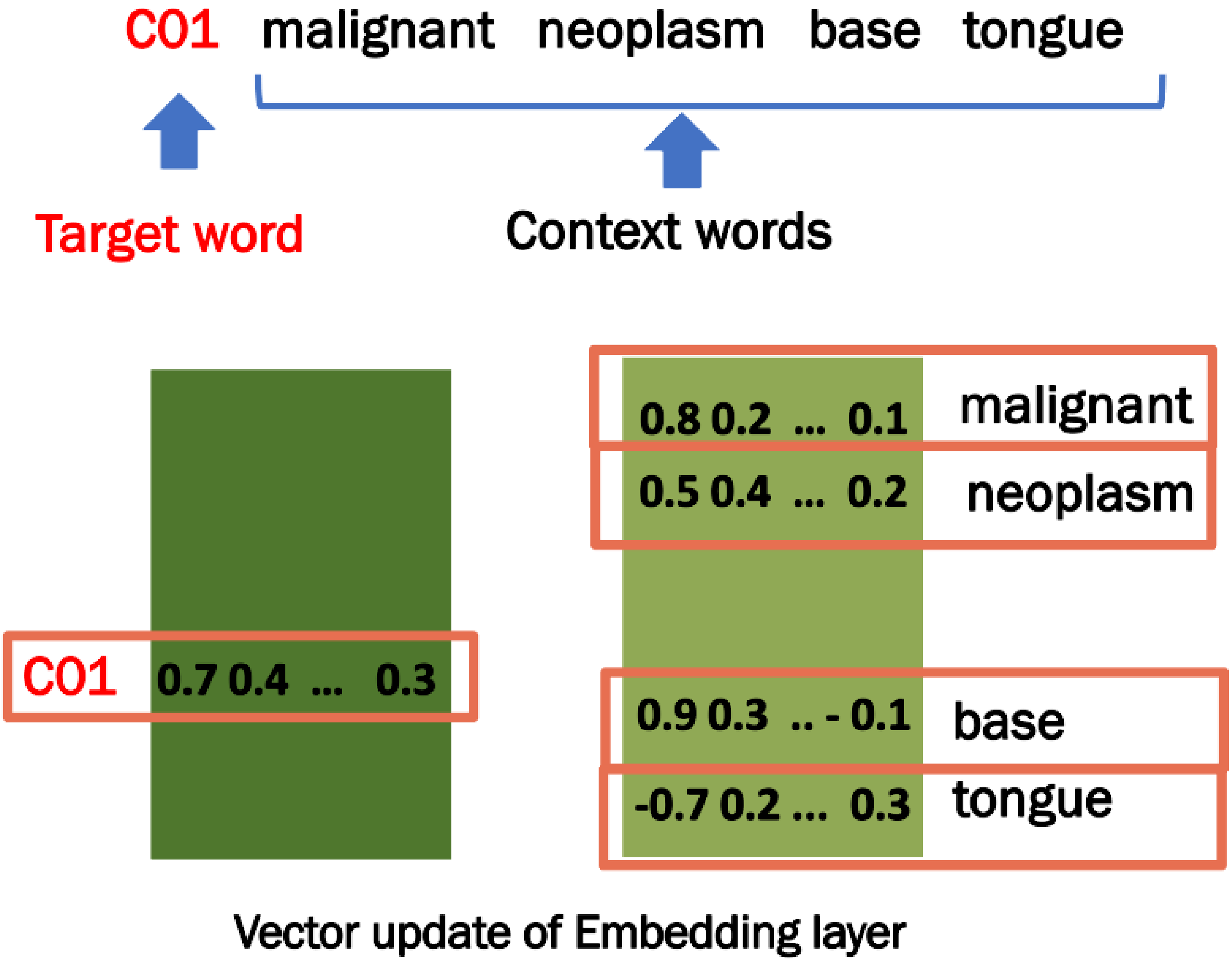
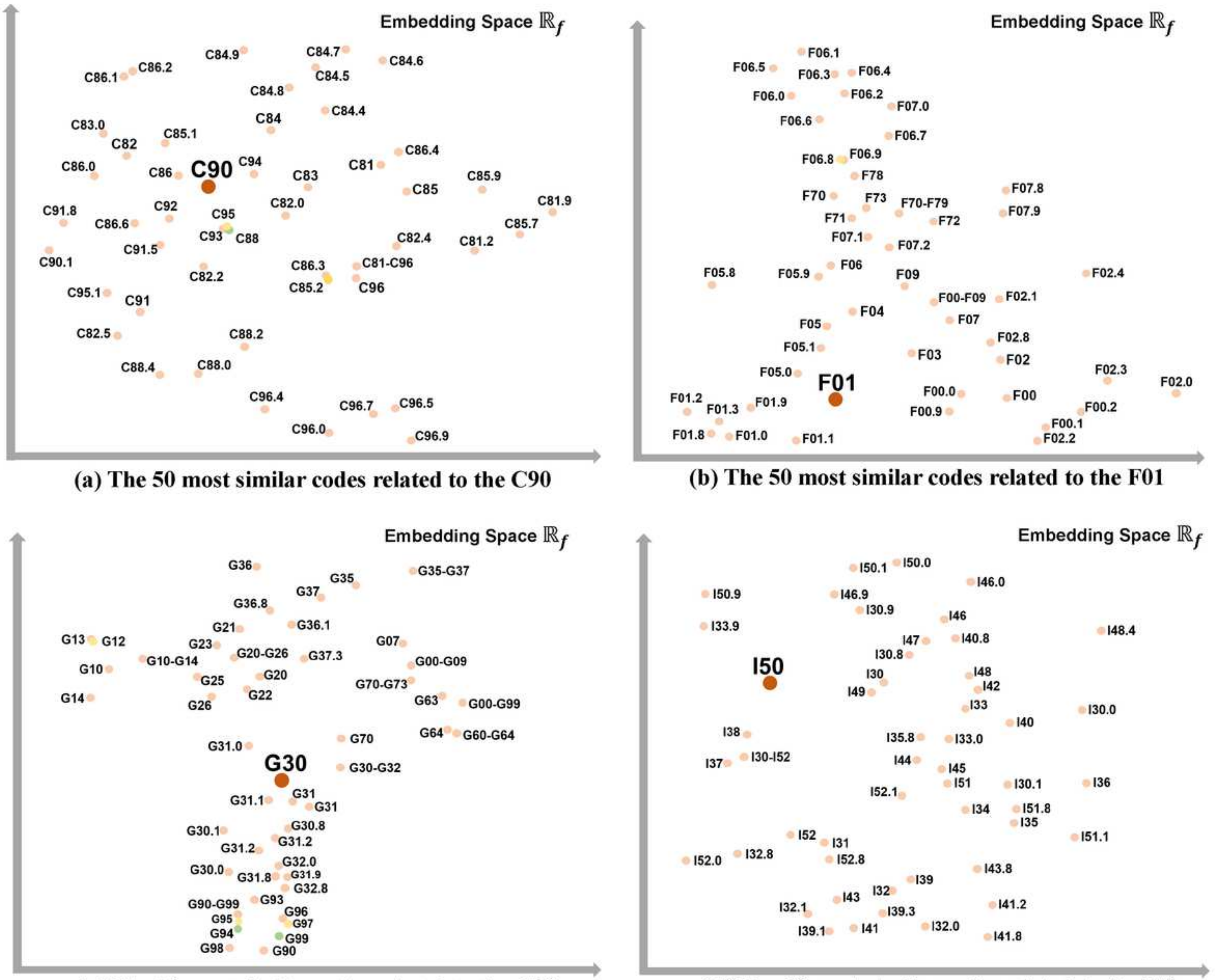


Figure 4

Sampling and training Procedure of the concept and text relation training



(a) The 50 most similar codes related to the C90

(b) The 50 most similar codes related to the F01

(c) The 50 most similar codes related to the G30

(d) The 50 most similar codes related to the I50

Figure 5

Visualization of target code’s 50 most similar ICD-10 codes using Principal Component Analysis (PCA). The target nodes (i.e., C90, F01, G30, and I50) were randomly