Learning Patient Similarity via Heterogeneous Medical Knowledge Graph Embedding

Zhihuang Lin, Dan Yang, Hua Jiang, Hang Yin

Abstract—**With the effective adoption of EHRs in clinical care, an increasing number of researchers contribute to meaningful use of EHRs for advancing in performance of patient similarity. Existing approaches leverage EHRs by the knowledge graph representation learning techniques, which shows higher effectiveness than the embedding vectors computed from medical texts. The unique challenge in embedding heterogeneous medical knowledge graph from real world EHR data is that such accurately structured medical data is very sparse. To address the above challenge, we propose a similarity framework named PSI. Specifically, PSI constructs a high-quality medical knowledge graph from ICD-9 ontology, MIMIC-III and DrugBank. Then, we utilize graph representation learning models to obtain the embedding vectors of the entities (prescriptions, diseases and patients). Moreover, PSI applies the embeddings to study the similarities of patients, which leverages Siamese CNN with SPP. The excellent performance demonstrates the effectiveness of PSI in expressively representing patients and measuring patient similarity.**

Index Terms—**Siamese CNN with SPP; Heterogeneous Medical Knowledge Graph Embedding; Patient Similarity;**

I. INTRODUCTION

The accumulation of EHR data provides a valuable resource for identifying similar patients, including demographics, prescriptions, procedures, vital signs, etc. Patient similarity learning [1] aims at finding patients who have similar clinical characteristics to the patients. Patient similarity aims to learn a clinical distance measurement to compute the similarities of patients. The patient similarity measures are the enabling technique for various healthcare domains [2-4].

A. Motivation

The precise patient similarity measures play an important role in grouping patients into disease cohorts. Consequently, how to derive an effective and accurate patient similarity measure is a significant challenge.

B. Challenges

Knowledge graph as an auxiliary data source have recently attracted a considerable amount of interest to extract the valuable clinical knowledge reflecting medical facts of prescriptions and diseases from EHRs. Therefore, a high-quality heterogeneous knowledge graph enables to learn the better representations of patients for patient similarity learning. However, there are still significant challenges on the heterogeneous medical knowledge graph: (1) **Computation inefficiency**: Knowledge representation learning models based on conventional graph factorization have many parameters, which makes the models too complex to explain and requires a lot of computational cost. The shortcomings such as complicated training and difficult expansion will occur. (2) **Data sparsity**: Most of the existing works merely focus on medical entities in EHRs, ignoring the relationships among medical entities. Given the clinical data from EHRs, discovering the relationships among medical entities in a wider scope can provide strong supports for clinical decision-making. Clearly, merely concentrating on EHR data is not reasonable.

C. Proposal

To exploit knowledge graphs for patient similarity learning as well as to address the above challenges, a framework named PSI is proposed, in which it consists of two parts: patient similarity learning and heterogeneous medical knowledge graph embedding. In heterogeneous medical knowledge graph embedding, a high-quality heterogeneous knowledge graph (See Fig. 1.) is constructed by extracting medical entities of patients and bridge the entities with ICD-9 ontology [5] and DrugBank [6]. Then, PSI enables a graph representation model to obtain the embedding vectors of entities. In this way, the entity embeddings preserve the information from the structures of heterogeneous medical knowledge graphs. In patient similarity learning, we stack the medical concept embeddings up patient matrices. Given the embedding matrix representations of patients, we adopt Siamese CNN [7] to find the patients with similar features.

C. Contributions

The works of PSI are as follows:

- We adopt the graph representation model to obtain entity vectors. The learned representations of patients, prescriptions, and diseases can help to have a positive effect on evaluating the similarities of patients.

Manuscript received June 25, 2020; revised October 21, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 6167214, in part by General Scientific Research Projects of Liaoning Province under Grant 2019LNJC07.

Zhihuang Lin is a Master of Computer Science and Software Engineering Department, University of Science and Technology LiaoNing, Anshan, China (e-mail: lzh_ustl@ 163.com).

Dan Yang is a Professor of Computer Science and Software Engineering Department, University of Science and Technology LiaoNing, Anshan, China (e-mail: asyangdan@ 163.com).

Hua Jiang is a Master of Computer Science and Software Engineering Department, University of Science and Technology LiaoNing, Anshan, China (e-mail: jh_ustl@ 163.com).

Hang Yin, the corresponding author, is a Lecturer of Computer Science and Software Engineering Department, University of Science and Technology LiaoNing, Anshan, China (corresponding author to provide e-mail: myworkspace08@ 126.com).

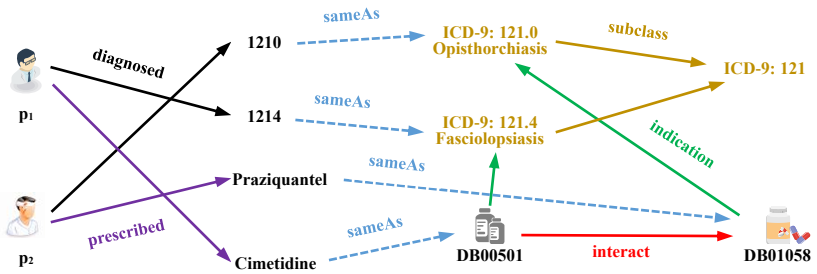


Fig. 1. An Annotated Toy Example of Heterogeneous Medical Knowledge Graph

- We learn the dense patient representations and utilize Siamese CNN with SPP to compute the patient similarity.
- The experimental results summary that PSI achieves the excellent performance in patient similarity learning than the existing baseline methods, including the SVD-based matrix factorization method, Word2vec-based method, CNN-based method and distance metric method.

II. Related Work

We first describe the medical knowledge graph. Then, we review the patient similarity learning algorithms.

A. Medical Knowledge Graph

Utilizing medical knowledge graphs has been demonstrated to effectively reflect relationships among clinical data. This enables to have great significance to improve the efficiency of doctors and reduce the burden on doctors. In recent years, the biomedical knowledge graphs have appeared, such as Bio2RDF, Chem2Bio2RDF and PDD [8-10].

B. Patient Similarity Learning

How to measure the patient similarity has caused the enormous concerns. For example, Chan *et al*. [11] proposed a method named SimSVM to measure patient similarity. Sun *et al*. [12] proposed the Locally Supervised Metric Learning (LSML) which combines multiple similarity measures from multiple physicians. Reference [13] proposed a CNN-based deep learning framework to obtain patient representations which preserve the temporal information of EHRs.

III. The Proposed Framework

Firstly, we briefly introduce the important notations employed in this paper. Then, we introduce how to effectively represent medical entity vectors. At last, we describe how to utilize Siamese CNN with SPP for measuring the similarities of patients. The Fig. 2 illustrates the overall framework of PSI.

A. Notations

A patient medical record is conventionally represented as a medical concept sequence ranked in a temporal order, and medical concepts include diseases and prescriptions that the patients suffered from or took. The medical concepts are presented in the form of ICD-9 and NDC. The set of distinct medical concepts in EHRs is denoted as $\mathcal{E} = \{c_1, c_2, \ldots, c_{|\mathcal{E}|}\}$, where c_i is the medical concept in EHRs and $|\mathcal{E}|$ is the total number of distinct medical concepts. We have a heterogeneous medical knowledge graph $G = (E, R)$ available, which is comprised of triples in the form (e_h, r, e_t). Here $e_h \in E$ (set of entities), $r \in R$ (set of relations), and $e_t \in E$ denote the head, relation and tail of a medical knowledge triple. For example, the triple (p_1, *prescribed*, *Cimetidine*) states the fact that the patient p_1 *prescribed* the prescription *Cimetidine*.

B. Heterogeneous Medical Knowledge Graph Embedding

Recently, translation-based methods have been proposed to train knowledge graph embeddings. Here, we use TransR [14], which is one of the most representative translation-based methods and has the excellent performance on many tasks, as our heterogeneous medical knowledge embedding model.

Formally, head entity e_h, tail entity e_t and relation r have a embedding respectively, i.e., $\mathbf{e}_h \in \mathbb{R}^k$, $\mathbf{e}_t \in \mathbb{R}^k$ and $\mathbf{r} \in \mathbb{R}^d$. For each relation r, we use a projection matrix $\mathbf{H}_r \in \mathbb{R}^{k \times d}$ to map the head entity e_h and the tail entity e_t from their entity space to the specific relation space. The scoring function $z(e_h, r, e_t)$ is to correct the triplet (e_h, r, e_t), which is defined as follows:

$$z(\mathbf{e}_h, \mathbf{r}, \mathbf{e}_t) = b - \|\mathbf{e}_h\mathbf{H}_r + \mathbf{r} - \mathbf{e}_t\mathbf{H}_r\|_{L1/L2} \qquad (1)$$

where b is a constant.

Then, we define the conditional probability of observing a triple (e_h, r, e_t) as follows:

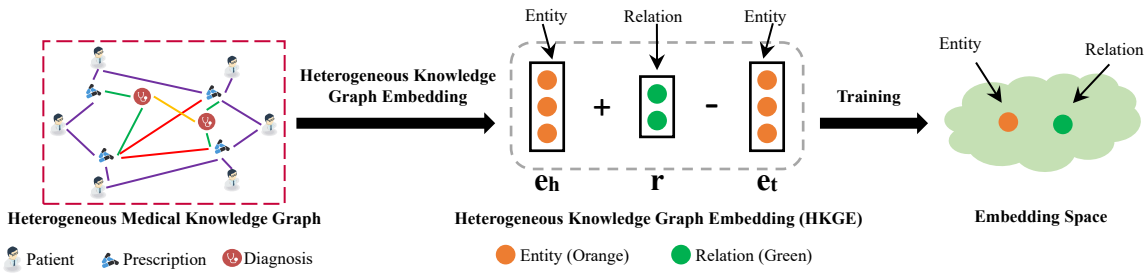


Fig. 2. The Overall Architecture of Heterogeneous Medical Knowledge Graph Embedding

$$P(e_h|r,e_t) = \frac{exp\{z(\mathbf{e}_h,\mathbf{r},\mathbf{e}_t)\}}{\sum_{\hat{e}_h \in E} exp\{z(\hat{\mathbf{e}}_h,\mathbf{r},\mathbf{e}_t)\}} \quad (2)$$

In addition, $P(e_t|e_h,r)$, $P(r|e_h,e_t)$ are defined in the same way as $P(e_h|r,e_t)$. The likelihood of observing a triple (e_h, r, e_t) is defined as follows:

$$\pounds(e_h,r,e_t) = log\, P(e_h|r,e_t) + log\, P(e_t|e_h,r) + log\, P(r|e_h,e_t) \quad (3)$$

Subsequently, the heterogeneous medical knowledge graph embedding model maximizes the likelihood of observing all triplets from the graph $G = (E, R)$:

$$\pounds_G = \sum_{(e_h,r,e_t)\in G} \pounds(e_h,r,e_t) \quad (4)$$

Optimization. To learn the effective medical entity/relation embeddings, we maximize the logarithm likelihood of the objective function:

$$\pounds(T) = \pounds_G + \gamma\Theta(T) \quad (5)$$

where T is the medical entity/relation embeddings, γ is a hyper-parameter, which is defined as follows:

$$\Theta(T) = \sum_{e\in E}[\|e\| - 1]_+ + \sum_{r\in R}[\|r\| - 1]_+ \quad (6)$$

where $[x]_+ = max(0, x)$ denotes the positive part of x.

Optimizing the objective function Equation (4) in Equation (5) are computationally expensive. To address this problem, we use NEG to transform the original objective.

For Equation (4), we should transform $log\, P(e_t|e_h,r)$, $log\, P(r|e_h,e_t)$, $log\, P(e_h|r,e_t)$ in Equation (3). Taking $P(e_h|r,e_t)$ as an example, we maximize the following objective function instead of it:

$$log\, \sigma\big(z(\mathbf{e}_h,\mathbf{r},\mathbf{e}_t)\big) + \sum_{i=1}^{\mu} \mathbb{E}_{\tilde{e}_h^i \sim z_{neg}(\{(\tilde{e}_h,r,e_t)\})} \left[\sigma\left(z\big(\tilde{\mathbf{e}}_h^i,\mathbf{r},\mathbf{e}_t\big)\right)\right] \quad (7)$$

where μ is the number negative examples, $\sigma(x) = 1/(1 + exp\,(-x))$ is the sigmoid function. $\{(\tilde{e}_h, r, e_t)\}$ is the invalid triple set, and z_{neg} is a function randomly sampling instances from $\{(\tilde{e}_h, r, e_t)\}$.

C. Patient Similarity Learning

Siamese CNN structure, inspired by the success of Siamese LSTM model in tackling the text similarity problem, can measure the patient similarity. Unfortunately, the fixed-size input of CNN limits the aspect ratio of the patient embedding matrices. When applying the patient embedding matrices of arbitrary sizes to the CNN, most of the existing methods via cropping or warping to fit the fixed-size input required by the CNN. But the cropping operation cannot fully consider the relationship, while the warping operation may result in stretching and deformation. To address the constraint, we add an SPP layer [15] behind the convolutional layer and before the fully-connected layer. Specifically, we obtain the patient matrix representations by stacking the medical concept embeddings. The patient matrix representations contain the disease information and prescription information. Through Siamese CNN with SPP, patient matrix representations are mapped to the fixed-size embeddings. Afterwards, we calculate the Euclidean distance between the fixed-size vectors.

1) *Patient Matrix Representation*

We describe how to obtain the patient matrix representations based on the learned embeddings and given patient medical records. In the existing related works, usually the patient representation is as simple as converting all medical concepts in his/her medical records to medical concept vectors, then summing all those vectors to obtain a single representation vector. Unfortunately, this kind of patient representation will loss the temporal information in EHRs. Instead, we utilize a novel patient representation method which stacks all medical concept embeddings appearing in a patient medical record into a matrix.

2) *Siamese CNN with SPP*

The Siamese CNN consists of two identical CNN-based sub-networks that join together at their output levels. The parameters and weights in the two Siamese sub-networks are shared. Because the dimensions of the patient matrix representations are different, we apply the spatial pyramid pooling strategy to extract the fixed-size spatial feature from patient matrix representations of arbitrary sizes. The main advantage of Siamese CNN with SPP is to overcome the defect brought by the patient matrix representations of arbitrary sizes and improve the performance of patient similarity learning.

a) *The Architecture of Siamese CNN with SPP*

Fig. 3 shows the schematic diagram of Siamese CNN with SPP. We assume that $\mathbf{X}_1 = [\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_M]^T$ and $\mathbf{X}_2 = [\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_N]^T$ are embedding matrix representations of two patients p_1 and p_2 respectively, where M and N are the total number of medical concepts appearing in their medical records respectively, and $\mathbf{c}_i$ is the embedding vector of medical concept c_i. A pair of patient matrix representations $\mathbf{X}_1$ and $\mathbf{X}_2$ are taken as the input of Siamese CNN with SPP and each of the two identical CNN-based sub-networks respectively handles one patient matrix representation from the patient pair. Through the operations of two Siamese

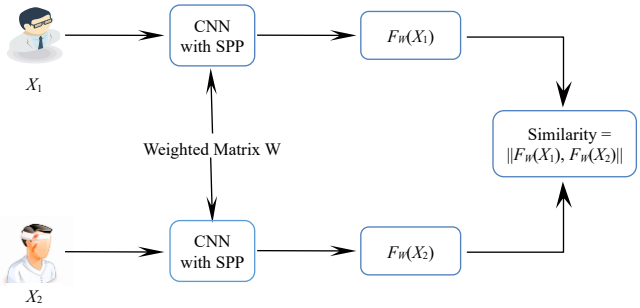


Fig. 3. Patient Similarity Learning Using Siamese CNN with SPP

sub-networks, we can obtain the feature vectors $\mathbf{F_W}(\mathbf{X}_1)$ and $\mathbf{F_W}(\mathbf{X}_2)$ of the pair of input patients. The output of Siamese CNN with SPP is the similarity between the feature vectors $\mathbf{F_W}(\mathbf{X}_1)$ and $\mathbf{F_W}(\mathbf{X}_2)$ of input patients $\mathbf{X}_1$ and $\mathbf{X}_2$, denoted $Dis_{\mathbf{W}} = \|\mathbf{F_W}(\mathbf{X_1}) - \mathbf{F_W}(\mathbf{X_2})\|$. The goal of Siamese CNN with SPP is to make the input patients of the same disease cohort as close as possible in the vector space, while the input patients of different disease cohorts are as far away as possible in the vector space.

b) Loss Function Used for Siamese CNN with SPP

Let $\mathbf{X}_1$ and $\mathbf{X}_2$ be a pair of inputs, which represent the embedding matrix representations of two patients, respectively. Let $\mathbf{W}$ be a shared weighted matrix, and $\mathbf{F_W}(\mathbf{X}_1)$ and $\mathbf{F_W}(\mathbf{X}_2)$ be two low-dimension vectors of $\mathbf{X}_1$ and $\mathbf{X}_2$. The contrastive loss [16] function are defined as follows:

$$Dis_{\mathbf{W}}(\mathbf{X}_1, \mathbf{X}_2) = \|\mathbf{F_W}(\mathbf{X}_1) - \mathbf{F_W}(\mathbf{X}_2)\|_2 \tag{8}$$

If $\mathbf{X}_1$ and $\mathbf{X}_2$ belong to the same disease cohort, the contrastive loss function should be small. Otherwise, the loss should be large. In detail, the loss function can be formed as:

$$\begin{aligned} L(\mathbf{W}, Y, \mathbf{X}_1, \mathbf{X}_2) &= \frac{Y}{2} Dis_{\mathbf{W}}(\mathbf{X}_1, \mathbf{X}_2)^2 \\ &+ \frac{1-Y}{2} (max\{0, m - Dis_{\mathbf{W}}(\mathbf{X}_1, \mathbf{X}_2)\})^2 \end{aligned} \tag{9}$$

where $m > 0$ is a margin and Y is a binary label assigned to the pair of input patient matrix representations. $Y = 1$ shows that the two patient matrix representations belong to the same disease cohort, which is indicated as positive patient pairs. Otherwise, $Y = 0$ shows that the two patient matrix representations belong to the different disease cohorts, which is indicated as negative patient pairs.

When $Y = 1$, the second term of Equation (9) is zero, and the first term of Equation (9) is called L_D which is the partial loss function for a pair of input patients belonging to the different disease cohorts. Otherwise, the first term of Equation (9) is zero, and the second term of Equation (9) is called L_S which is the partial loss function for a pair of input patients belonging to the same disease cohort.

D. Patient Similarity Identification—PSI

Algorithm 1 displays the steps of our proposed patient similarity learning framework PSI written in pseudo code. Specifically, A batch of medical fact triplets (e_h, r, e_t) are selected from the heterogeneous medical knowledge graph G, and we random generate μ negative triplets based on the positive samples (Steps 2-5). To update the embeddings of medical entities and relations, we further conduct SGD algorithm in the learning process. In addition, given a patient p, we can obtain the embedding matrix $\mathbf{X}_p$ of patient p by stacking the embedding of medical concepts orderly that appear in his/her medical record (Steps 7-8). Then, we train Siamese CNN with SPP using the patient embedding representations as the input to measure the clinical similarities of all patient pairs (Step 9). Lastly, for each patient, we select the patient corresponding to the highest similarity score (Steps 10-16).

Algorithm 1 Patient Similarity Identification—PSI

Input: Patient set P, patient medical record set D, heterogeneous medical knowledge graph G, medical entity set E, relation set R and number of negative samples μ

Output: The most similar patient set $\hat{P}$

1: **repeat**
2: Sample a batch of triples from G_{batch} from G
3: **for** $(e_h, r, e_t) \in G_{batch}$ **do**
4: Update $\mathbf{e}_h, \mathbf{r}, \mathbf{e}_t$ by using Equation (1-4) with negative sampling
5: **end for**
6: **until** Convergence
7: **foreach** $p \in P$ **do**
8: Obtain the embedding matrix $\mathbf{X}_p$ according to the medical record D_p of patient p
9: Train Siamese CNN with SPP to measure the similarities of patients
10: $\hat{P} \leftarrow \{\}$
11: **foreach** $p_i \in P$ **do**
12: **foreach** $p_j \in P \backslash p_i$ **do**
13: Compute the similarity score between p_i and p_j
14: Rank the similarity score
15: Select the patient p_j corresponding to the highest similarity score
16: $\hat{P} \leftarrow p_j$
17: Return $\hat{P}$

IV. Experiments

We conduct the experiments to demonstrate the effectiveness of PSI. Firstly, we introduce several real-world datasets and baseline methods to be compared. Then, we evaluate the performance of PSI on six data mining tasks: top-k most similar patients, patient similarity analysis, visualization of patient, disease cohort classification, patient clustering and parameter sensitivity analysis.

A. Experimental Settings

1) Datasets

- **MIMIC-III** [17] is a free and open intensive care medical information database, which collects clinical data of the Beth Israel Deaconess Medical Center between 2001 and 2012.
- **ICD-9 ontology** contains 13,000 international standard codes of diagnoses and their hierarchical relationships.
- **DrugBank** is a database that integrates bioinformatics and cheminformatics, and provides detailed drug data, drug target information and comprehensive molecular information about its mechanism. The database contains 10,971 drugs information and 4,900 protein targets information.

2) Construction of Heterogeneous Medical Knowledge Graph

We introduce the process of constructing the heterogeneous medical knowledge graph from EHR data MIMIC-III and two medical knowledge bases (ICD-9 ontology and DrugBank). Furthermore, we present how to extract the patients and their medical concepts from MIMIC-III.

a) Patient Extraction

According to the studies in [13], we carry out the selection

TABLE I
STATISTICS OF MEDICAL ENTITIES

Medical Entities	# Cardinality
Patient	26,009
Diagnosis	4,759
Medication	3,062
Diagnosis-related	4,759
Medication-related	1,500
Total	40,089

TABLE II
STATISTICS OF RELATIONS

Relations	# Cardinality
Patient-Diagnosis	283,976
Patient-Medication	695,089
Diagnosis-Diagnosis	6,037
Medication-Medication	36,768
Medication-Diagnosis	763,265
sameAs	8,117
Total	1,793,252

of disease cohort, including Atherosclerosis, Heart Failure, Kidney Failure, Intestinal Diseases, Liver Diseases, Pneumonia, Septicemia, Respiratory Failure and Gastritis. Each disease cohort consists of a set of case patients who are confirmed with one of the nine diseases according to their diagnosis information in MIMIC-III and discharged from the hospital before 2200/1/1, and each patient medical record comes with more than three medical concepts including diagnosis and prescription information, etc. As a result, the remaining dataset contains 26,009 patients.

b) Medical Concept Extraction

In the process of generate a high-quality heterogeneous medical knowledge graph, we focus on discovering the medical fact triples indicating that a patient is diagnosed with a disease, a patient takes a prescription, or a prescription cures a disease. Consequently, we need to extract two types of medical concepts from MIMIC-III (prescriptions and diagnoses), and generate the medical fact triples used in heterogeneous medical knowledge graph embedding. To make use of MIMIC-III and avoid biases and noise, we remove medical concepts occurring less than 30 times. Totally 7,821 distinct medical concepts are finally selected for construction of the heterogeneous medical knowledge graph.

c) Linking EHRs to Medical Knowledge Bases

To generate a high-quality medical knowledge graph, we need to link EHRs with two knowledge bases. In the linking process, the *sameAs* links becomes an obstacle. For example, the prescription names are various and contain some noisy words (10%, 200ml, glass bottle, etc.). To address this problem, we adopt an entity linking method [10] to link prescription names to DrugBank. Table I shows the result of medical entities from MIMIC-III, ICD-9 ontology and DrugBank, and Table II shows the result of the relationships between medical entities in the heterogeneous medical knowledge graph.

3) Baselines

We compare PSI with the baselines for patient similarity measurment, in which the first baseline is Word2Vec-free while the rest are all Word2Vec-aware methods.

- **PCA** [18], or Principal Component Analysis, is used to retain some important features of high-dimensional data, and remove noise features.
- **CSM** [19] represents a patient by summing up all its medical concept vectors, absolutely eliminating the sequential structure of medical concepts.
- **Deep Embedding** is a deep learning framework based on CNN, which computes the clinical similarities between patients.
- **RV** is an unsupervised method [13] that utilizes RV coefficient to estimate the clinical similarity between patient matrix representations.
- **dCor** is an unsupervised method [13] that uses dCor coefficient to measure linear and non-linear relations between the pairwise embedding matrices.

4) Evaluation Metrics

Precisely, we adopt two popular metrics [20] hospital readmission rate (HRR) and incident rate difference for mortality ($IRDM$) for evaluating the similarities of patients. We evaluate the performance of patient clustering by using two popular criteria: Rand Index (RI) [21], $Purity$ [22] and Normalized Mutual Information (NMI) [23]. All the metrics except $IRDM$ are preferred to be higher to present better performance. Their definitions are listed as follows.

a) Hospital Readmission Rate (HRR)

Assume $P = \{p_1, p_2, \dots, p_N\}$ is the collection of readmission statuses of N patients and $SP = \{p'_1, p'_2, \dots, p'_N\}$ is the collection of readmission statuses of the most similar patients of N patients. HRR is computed as follows:

$$HRR = \sum_{i=1}^{N} \omega(P[i], SP[i]) / N \tag{10}$$

where $\omega(P[i], SP[i]) = \begin{cases} 0, P[i] \neq SP[i] \\ 1, P[i] = SP[i] \end{cases}$.

b) Incidence Rate Difference for Mortality (IRDM)

Assume $P = \{(t_1, d_1), (t_2, d_2), \dots, (t_N, d_N)\}$ is the collection of tuples (discharge date, death date) of N patients, where t_i is the discharge date, and d_i is the death date. The incidence rate of the collection of N patients is computed as follows:

$$IR(Case) = \frac{count(death)}{\sum_{i=1, d_i \neq null}^{N} (d_i - t_i) + \sum_{i=1, d_i = null}^{N} (d_{null} - t_i)} \tag{11}$$

where $count(death)$ is the number of patients which have the death dates and d_{null} is 2200/1/1.

Similarly, we can compute the incidence rate of the most similar patients of N patients, called $IR(SP)$. $IRDM$ is computed as follows:

$$IRDM = |IR(Case) - IR(Control)| \tag{12}$$

TABLE III
TOP-3 SIMILARITY PATIENTS (PCA)

Patient (ID)	1st Patient (ID)	2nd Patient (ID)	3rd Patient (ID)
876	14216	211	3434
454	19916	5398	9232
1064	2156	9542	16407
7747	11560	17399	8554
23147	20703	5433	5919

TABLE IV
TOP-3 SIMILARITY PATIENTS (CSM)

Patient (ID)	1st Patient (ID)	2nd Patient (ID)	3rd Patient (ID)
876	15383	2996	13202
454	18604	5516	11644
1064	422	3781	2878
7747	9016	25206	21253
23147	24941	22098	25213

TABLE V
TOP-3 SIMILARITY PATIENTS (DEEP EMBEDDING)

Patient (ID)	1st Patient (ID)	2nd Patient (ID)	3rd Patient (ID)
876	22146	6904	15383
454	115	21746	9950
1064	24855	9675	422
7747	10332	495	21946
23147	24768	1338	18959

TABLE VI
TOP-3 SIMILARITY PATIENTS (RV)

Patient (ID)	1st Patient (ID)	2nd Patient (ID)	3rd Patient (ID)
876	18878	11150	18401
454	10416	19211	5516
1064	14441	422	11003
7747	4112	19241	1802
23147	14791	22098	18037

TABLE VII
TOP-3 SIMILARITY PATIENTS (DCOR)

Patient (ID)	1st Patient (ID)	2nd Patient (ID)	3rd Patient (ID)
876	20234	18401	12912
454	10416	903	12139
1064	10449	2896	2842
7747	14348	25206	2087
23147	25086	14665	10569

TABLE VIII
TOP-3 SIMILARITY PATIENTS (PSI)

Patient (ID)	1st Patient (ID)	2nd Patient (ID)	3rd Patient (ID)
876	25042	20723	20438
454	2728	16980	12116
1064	16258	15007	8546
7747	164	23726	24447
23147	20305	23901	21641

c) Rand Index (RI)

RI is an evaluation metric that measures the pros and cons of data clustering. RI is computed as follows:

$$RI = \frac{TP + TN}{\binom{n}{2}} \tag{13}$$

where TP is the number of times a pair of patients belonging to the same cohort who are grouped into one single cluster. TN is the number of times a pair of patients from different cohorts who are grouped into different clusters. n is the total number of patients.

d) Purity

$Purity$ is a simple and transparent evaluation measure for data clustering, which can be computed as follows:

$$Purity(Cluster, Cohort) = \frac{1}{n}\sum_{i} \max_{j} \left|p_i \cap q_j\right| \tag{14}$$

where n is the total number of patients, $Cluster = \{p_1, p_2, \dots, p_I\}$ is the set of patient clusters, and $Cohort = \{q_1, q_2, \dots, q_J\}$ is the set of disease cohorts.

e) Normalized Mutual Information (NMI)

NMI is an evaluation metric used to measure the degree of agreement between two data distributions. NMI is computed as follows:

$$NMI(X, Y) = \frac{2 \cdot I(X, Y)}{[H(X) + H(Y)]} \tag{15}$$

where $I(X, Y)$ is Mutual Information, which is the relative entropy of the joint distribution $p(x, y)$ and the product distribution $p(x)(y)$, whose formula is:

$$I(X, Y) = \sum_{x}\sum_{y} p(x, y)\, log\, \frac{p(x, y)}{p(x)p(y)} \tag{16}$$

$H(X)$ is the information entropy, and the formula is:

$$H(X) = -\sum_{i} p(x_i)\, log\, p(x_i) \tag{17}$$

5) Parameter Settings

The latent dimension sizes of medical entity embedding k, relation embedding d are both set to 100. The fixed learning rate α for TransR is set to 0.025. The number of negative samples μ is set to 10. The hyper-parameter γ is optimized from {0, 1E-7, 1E-5, 1E-3, 1E-1}. The margin b is set to 8.0 and the distance function is set to L_1-norm.

We set learning rate to be 0.001 and use $f = 100$ for the number of convolutional filters in the CNN. We set the dropout rate at the fully connected layer as 0.6 and use 0.0009 for the regularization parameter. The margin m for the contrastive loss function is tuned from $\{0.5, 1.0, 1.5, 2.0, 2.5\}$. Our proposed framework PSI is implemented on TensorFlow.

B. Experimental Results

We present experimental evaluation results of the proposed framework PSI. We argue that PSI can produce meaningful patient representations for patient similarity learning by fully taking advantage of knowledge representation learning.

1) Top-k most similar patients

TABLE IX
HOSPITAL READMISSION RATE (*HRR*)

Methods	Technique	*HRR*
PCA	Non-embedding	0.593
CSM	Word2Vec	0.684
Deep Embedding	Word2Vec-CNN	0.737
RV	Distance metrics	0.691
dCor	Distance metrics	0.693
PSI	HKGE	0.792

TABLE X
INCIDENCE DIFFERENCE FOR MORTALITY (1E-5)

Methods	Technique	*IRDM*
PCA	Non-embedding	0.420
CSM	Word2Vec	0.336
Deep Embedding	Word2Vec-CNN	0.298
RV	Distance metrics	0.359
dCor	Distance metrics	0.363
PSI	HKGE	0.267

In this experiment, we first choose 5 patients at random, and then run our proposed framework PSI and all baseline methods to obtain the top-k (k = 3) most similar patients. Table III, IV, V, VI, VII and VIII summary the top-3 most similar patients obtained by PCA, CSM, Deep Embedding, RV, dCor and PSI. We can see that the results of PSI and all baseline methods have nothing in common. This is probably due to the fact that PSI utilizes graph representation models to learn the low-dimensional and continuous feature representations of medical entities, which considers the structure of medical knowledge graph. However, the result of CSM has a lot in common with that of RV and dCor. For example, with respect to the patient ID 1064, the 1st patient' ID obtained by CSM is the same with the 2nd patient' ID obtained by RV. For patient ID 454, the 1st patient' ID obtained by RV is the same with the 1st patient' ID obtained by dCor. This is mainly because these three patient similarity learning methods leverage the Skip-gram model to learn the medical concept vectors. In addition, the result of Deep Embedding has a little in common with that of CSM and RV. For example, with respect to the patient ID 876, the 3rd patient' ID obtained by Deep Embedding is the same with the 1st patient' ID obtained by CSM. For patient ID 1064, the 3rd patient' ID obtained by Deep Embedding is the same with the 2nd patient' ID obtained by RV.

2) Performance Comparison of Patient Similarity Learning

We use *HRR* and *IRDM* as our evaluation metric in patient similarity analysis task. The experimental results of *HRR* and *IRDM* are shown in Table IX and X respectively. From the Table IX and X, we can observe that proposed PSI significantly outperforms even all the baselines, which indicates that the performance of patient similarity learning can be improved better through the medical knowledge graph embedding techniques. Compared to the best performing baseline Deep Embedding, PSI achieves an improvement from 0.737 to 0.792 in *HRR* and 0.298 to 0.267 in *IRDM*.

Among the five baseline methods, the PCA based approach achieves the lowest *HRR* and *IRDM*. The reason we consider is that PCA learns low-dimensional representations directly from the correlation matrix while not considering the contextual correlations between medical concepts. CSM, Deep Embedding, RV and dCor instead employ Word2Vec to embed medical concepts in EHRs, which preserves the semantic information of medical contexts. Deep Embedding combines CNN with distributional medical events embeddings from Word2Vec to derive the similarity scores for pairs of patients, which performs better than CSM, RV and dCor. The result is not as good as our proposed framework PSI. This is probably due to ignoring the structural information of knowledge graphs. The superior performance of PSI indicates that capturing the structural information of knowledge graphs can help the similarity learning of patients. Overall, the experiment results on patient similarity indicate that PSI is effective in improving the performance of patient similarity learning.

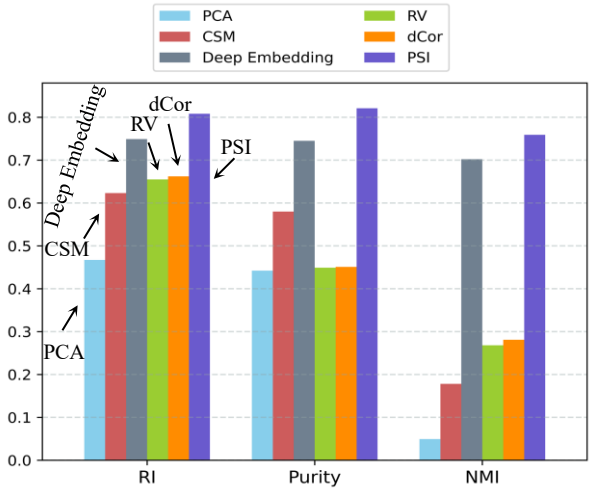
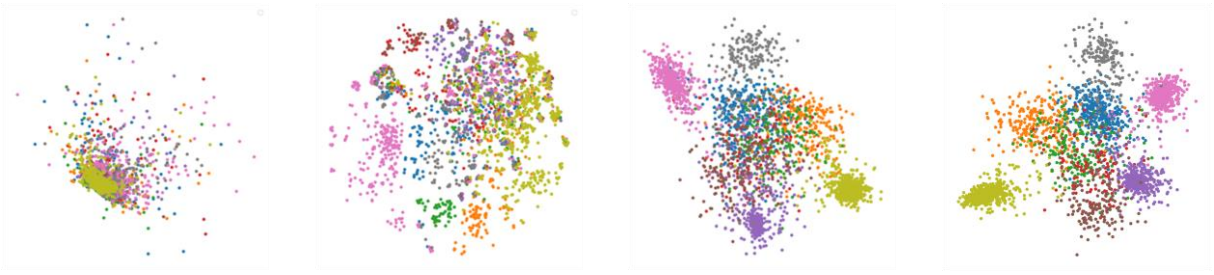


Fig. 4. Performance of Patient Clustering

3) Clustering Results

Fig. 4 summaries the results of patient clustering. As we can see, the prop osed PSI is clearly superior to others. Measured by *RI*, it can achieve the performance of 0.808, which outperforms the second best one with 0.749 in a light margin. The superiority of PSI is illustrated in *Purity* and *NMI* as well. Measured by *Purity*, PSI can achieve 0.821, which outperforms others with a wide margin. Measured by *NMI*, PCA, CSM, Deep Embedding, RV, dCor and PSI achieve 0.0493, 0.178, 0.702, 0.268, 0.281 and 0.759 separately. On one hand, compared with PCA, CSM, Deep Embedding, RV and dCor can only improve 261%, 1324%, 444% and 470% on *NMI* respectively. On the other hand, the proposed PSI achieves about more than 1440% improvement over them.

As a reasonable explanation, we view that the proposed PSI can be divided into two stages. During the first stage, the clinical features of patients are summarized in the heterogeneous medical knowledge graph embedding, achieving the better performance than Word2Vec. Next, global features are learned by leveraging Siamese CNN with SPP. In such a way, the learned patient representations make continuous improvement, which leads to an ultimate expression of patients.



(a) PCA (b) CSM (c) Deep Embedding (d) PSI

Fig. 5. Visualization of patients. Each point indicates one patient. Color of a point indicates the cohort of the patient

4) Visualization of Patients

One way of assessing the quality of the patient embedding representations is through visualization. We conduct visualization experiments to compare the performance of PSI with PCA, CSM and Deep Embedding on patients. The patient embedding representations are fed as features into t-SNE [24], which mapped all points into a 2D space, where the patients of the same disease cohort are highlighted with the same color.

From Fig. 5, we can see that the visualization of outputs from PCA and CSM shows unclear boundaries and diffuse clusters. Deep Embedding is much better. Although Deep Embedding is superior to PCA and CSM, in the results we can see the points of different colors are still intermixed in the center of the visualization. PSI is the clear winner.

5) Disease Cohort Classification

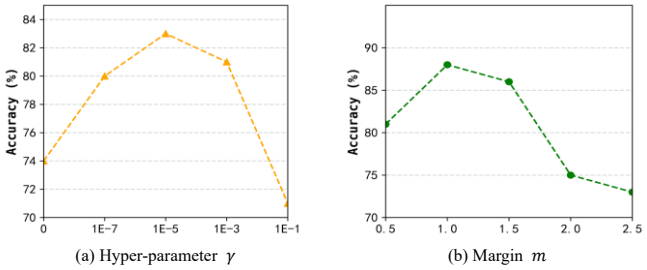
We further investigate the effectiveness of the proposed PSI on disease cohort classification task. In the experiment, we successfully transform the patients of EHRs into the low-dimensional representations using different patient similarity learning methods including PCA, CSM, Deep Embedding and PSI, and apply AdaBoost classification on the learned patient representations in order to correctly diagnose the diseases suffered by the patients. In addition, we use Macro Area Under The Curve (Macro-AUC), accuracy and Macro-F1 to evaluate the performance of disease cohort classification task, and use 10-fold cross-validation in which we randomly select 80% of the patients for learning and the remaining 20% of patients for testing the AdaBoost classification.

Comparative results of different patient similarity learning methods for disease cohort classification task are shown in Table XI. We observe that our proposed PSI achieves Macro-AUC of 0.854, accuracy of 0.911, and Macro-F1 of 0.619, which outperforms all the other methods, and Deep Embedding achieves the second highest performance. It is reasonable that the structural information of the heterogeneous medical know ledge graph plays a more important role than the semantic information of medical concepts in deriving meaningful representations of patients from EHRs.

TABLE XI
DISEASE COHORT CLASSIFICATION RESULTS

Method	Technique	Macro-AUC	Accuracy	Macro-F1
PCA	Principal Component Analysis	0.604	0.738	0.417
CSM	Word2Vec	0.726	0.792	0.446
Deep Embedding	Word2Vec-CNN	0.807	0.875	0.521
PSI	KGE	0.854	0.911	0.619

C. Parameter Sensitivity



(a) Hyper-parameter γ (b) Margin m

Fig. 6. Parameter Sensitivity Analysis

Fig. 6 (a) shows the parameter sensitivity analysis on the hyper-parameter γ used for maximizing the logarithm likelihood of the objective function. We evaluate five different values in hyper-parameter γ. From the results, we find the performance is continuously improved when the value of the hyper-parameter γ is increased until a turning point. If the value of hyper-parameter γ is too large, the performance starts to degrade since it is easily overfitted to the training set. For instance, the performance of patient similarity learning starts to degrade after the value of hyper-parameter γ is larger than $1\mathrm{E}-5$.

Fig. 6 (b) shows the parameter sensitivity analysis on the margin m used to optimize the contrastive loss in Siamese CNN with SPP. We evaluate five different values in the margin m, including 0.5, 1.0, 1.5, 2.0 and 2.5. From the results, we observe that neither a too small nor a too big value of margin m results in optimal performance. On one hand, a too small value of margin m could learn a bad network easily in the learning process, thus leading to degraded performance. On the other hand, a too large value of margin m easily leads to separating the data from different cohorts but makes it difficult to train the network in the learning process, resulting

in suboptimal performance. For instance, the performance of $m = 2.0$ is worse than that of $m = 1.0$ in patient similarity learning.

V. Conclusions and Future Work

In this paper, we present a general framework named Patient Similarity Identification (PSI) for learning similarity between patient pairs. PSI enables a translation-based embedding model to learn the medical entity representations, which automatically captures the structural information of the heterogeneous medical knowledge graph. Meanwhile, a deep learning model named Siamese CNN with SPP is designed to automatically learn the correlation degree between patients. Extensive experiments conducted on three datasets demonstrate that PSI significantly outperforms existing patient similarity learning approaches, including both traditional and Word2Vec-based ones.

References

[1] J. Sun, F. Wang, J. Hu, and S. Edabollahi, "Supervised patient similarity measure of heterogeneous patient records," *ACM SIGKDD Explorations Newsletter*, vol. 14, no. 1, pp. 16–24, 2012.

[2] Dongyang Li, Dan Yang, Jing Zhang, and Xuedong Zhang, "AR-ANN: Incorporating Association Rule Mining in Artificial Neural Network for Thyroid Disease Knowledge Discovery and Diagnosis," *IAENG International Journal of Computer Science*, vol. 47, no.1, pp. 25-36, 2020.

[3] Riyanarto Sarno, Shoffi Izza Sabilla, Dedy Rahman Wijaya, and Hariyanto, "Electronic Nose for Detecting Multilevel Diabetes using Optimized Deep Neural Network," *Engineering Letters*, vol. 28, no.1, pp. 31-42, 2020.

[4] M. Lamy, R. Pereira, J. C. Ferreira, F. Melo, and I. Velez, "Extracting clinical knowledge from electronic medical records," *IAENG International Journal of Computer Science*, vol. 45, no. 3, pp. 488–493, 2018.

[5] L. M. Schriml, C. Arze, S. Nadendla, Y. W. Chang, M. Mazaitis, V. Felix, G. Feng, and W. A. Kibbe, "Disease ontology: a backbone for disease semantic integration," *Nucleic Acids Research*, vol. 40, pp. 940–946, 2012.

[6] V. Law, C. Knox, Y. Djoumbou, T. Jewison, A. C. Guo, Y. Liu, A. Maciejewski, D. Arndt, M. Wilson, V. Neveu et al., "Drugbank 4.0: shedding new light on drug metabolism," *Nucleic Acids Research*, vol. 42, pp. 1091–1097, 2014.

[7] J. Bromley, J. W. Bentz, L. Bottou, I. Guyon, Y. Lecun, C. Moore, E. Sackinger, and R. Shah, "Signature verification using a siamese' time delay neural network," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 7, no. 4, pp. 669–688, 1993.

[8] A. Callahan, J. Cruz-Toledo, P. Ansell, and M. Dumontier, "Bio2rdf release 2: Improved coverage, interoperability and provenance of life science linked data," in *The Semantic Web: Semantics and Big Data*. Springer Berlin Heidelberg, 2013, pp. 200–212. doi: 10.1007/978-3-642-38288-8_14.

[9] B. Chen, X. Dong, D. Jiao, H. Wang, Q. Zhu, Y. Ding, and D. J. Wild, "Chem2bio2rdf: a semantic framework for linking and data mining chemogenomic and systems chemical biology data," *BMC Bioinformatics*, vol. 11, no. 1, p. 255, 2010.

[10] M. Wang, J. Zhang, J. Liu, W. Hu, S. Wang, X. Li, and W. Liu, "Pdd graph: bridging electronic medical records and biomedical knowledge graphs via entity linking," *arXiv: Databases*, vol. 10588, pp. 219–227, 2017.

[11] L. Chan, T. Chan, L. Cheng, and W. Mak, "Machine learning of patient similarity: A case study on predicting survival in cancer patient after locoregional chemotherapy," in *Bioinformatics and Biomedicine Workshops (BIBMW), 2010 IEEE International Conference on*. IEEE, 2010, pp. 467–470.

[12] J. Sun, D. Sow, J. Hu, and S. Ebadollahi, "Localized supervised metric learning on temporal physiological data," in *2010 20th International Conference on Pattern Recognition*, 2010, pp. 4149–4152. doi: 10.1109/ICPR.2010.1009.

[13] Z. Zhu, C. Yin, B. Qian, Y. Cheng, J. Wei, and F. Wang, "Measuring patient similarities via a deep architecture with medical concept embedding," in *2016 IEEE 16th International Conference on Data Mining (ICDM)*, 2016, pp. 749–758. doi: 10.1109/ICDM.2016.0086.

[14] Y. Lin, Z. Liu, X. Zhu, X. Zhu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Proc. AAAI*, Feb. 2015, pp. 2181–2187. doi: 10.5555/2886521.2886624.

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[16] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, 2006, pp. 1735–1742.

[17] A. E. W. Johnson, T. J. Pollard, L. Shen, L. H. Lehman, M. Feng, M. M. Ghassemi, B. Moody, P. Szolovits, L. A. Celi, and R. G. Mark, "Mimic-iii, a freely accessible critical care database," *Scientific Data*, vol. 3, no. 1, p. 160035, 2016.

[18] I. T. Jolliffe, "Principal component analysis," *Journal of Marketing Research*, vol. 87, no. 4, p. 513, 2002.

[19] E. Choi, A. Schuetz, W. F. Stewart, and J. Sun, "Medical concept representation learning from electronic health records and its application on heart failure prediction," Feb. 2016, *arXiv:1602.03686*. [Online]. Available: https://arxiv.org/abs/1602.03686

[20] D. Nguyen, W. Luo, S. Venkatesh, and D. Phung, "Effective identification of similar patients through sequential matching over icd code embedding," Journal of Medical Systems, vol. 42, no. 5, p. 94, 2018.

[21] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical Association*, vol. 66, no. 336, pp. 846–850, 1971.

[22] C. D. Manning, P. Raghavan, H. Scütze et al., "Introduction to information retrieval," *Cambridge University Press Cambridge*, 2008, vol. 1, no. 1.

[23] M. Meil, "Comparing clusteringsan information based distance," *Journal of Multivariate Analysis*, vol. 98, no. 5, pp. 873–895, 2007.

[24] S. Arora, W. Hu, and P. Kothari, "An analysis of the t-sne algorithm for data visualization," in *Conference on Learning Theory*, 2018, pp. 1455–1462.

Zhihuang Lin is currently pursuing the master's degree in the School of Computer Science and Software Engineering, University of Science and Technology Liaoning, China. He received his B.S. in the School of Computer Science and Software Engineering, University of Science and Technology Liaoning, China, from 2014 to 2018. His current research interests include deep learning, network embedding, and data mining.

Dan Yang is currently a professor at School of Computer Science and Software Engineering, University of Science and Technology Liaoning, China. She received her M.S. and Ph.D. degrees in Computer Software and Theory from Northeastern University, China, in 2004 and 2013 respectively. Dr. Yang was a visiting scholar in New Jersey Institute of Technology, U.S.A from June 2015 to May 2016 supported by Chinese Scholarship Council of the Ministry of Education. Dr. Yang is a member of the CCF (China Computer Federation). Her research interests include data integration, big data management and applications in health care.

Hua Jiang is currently pursuing the master's degree in the School of Computer Science and Software Engineering, University of Science and

Technology Liaoning, China. She received her B.S. from the School of Computer Science and Software Engineering, University of Science and Technology Liaoning, China, in 2017. Her current research interests include data mining, big data management and applications in health care.

Hang Yin is currently a lecturer at School of Computer Science and Software Engineering, University of Science and Technology Liaoning, China. He received the M.S. degree in computer application from Liaoning Technical University, China, in 2007. His current research interests include data mining, big data management and applications in health care.