

KGNN: Knowledge Graph Neural Network for Drug-Drug Interaction Prediction

Xuan Lin¹, Zhe Quan^{1,*}, Zhi-Jie Wang^{2,*}, Tengfei Ma¹ and Xiangxiang Zeng¹

¹College of Information Science and Engineering, Hunan University

²College of Computer Science, Chongqing University

{jack_lin, quanzhe, tfma, xzeng}@hnu.edu.cn, cszjwang@yahoo.com

Abstract

Drug-drug interaction (DDI) prediction is a challenging problem in pharmacology and clinical application, and effectively identifying potential DDIs during clinical trials is critical for patients and society. Most of existing computational models with AI techniques often concentrate on integrating multiple data sources and combining popular embedding methods together. Yet, researchers pay less attention to the potential correlations between drug and other entities such as targets and genes. Moreover, recent studies also adopted knowledge graph (KG) for DDI prediction. Yet, this line of methods learn node latent embedding directly, but they are limited in obtaining the rich neighborhood information of each entity in KG. To address the above limitations, we propose an end-to-end framework, called **Knowledge Graph Neural Network (KGNN)**, to resolve the DDI prediction. Our framework can effectively capture drug and its potential neighborhoods by mining their associated relations in KG. To extract both high-order structures and semantic relations of the KG, we learn from the neighborhoods for each entity in KG as their local receptive, and then integrate neighborhood information with bias from representation of the current entity. This way, the receptive field can be naturally extended to multiple hops away to model high-order topological information and to obtain drugs potential long-distance correlations. We have implemented our method and conducted experiments based on several widely-used datasets. Empirical results show that KGNN outperforms the classic and state-of-the-art models.

1 Introduction

Drug-drug interactions (DDIs) can often occur when a drug is co-administered with another and multiple drugs, which will result in many adverse drug reactions (ADRs) that may cause injuries or deaths [Vilar *et al.*, 2014]. Therefore, to alleviate

the impact of unexpected pharmacological effects, it is critical to effectively identify potential DDIs, which can minimize unexpected ADRs and maximize synergistic benefits [Lin *et al.*, 2019] when treating a disease to some extent.

Most of existing DDI prediction methods often concentrate on integrating multiple data sources to obtain drug features, including similarity features [Vilar *et al.*, 2012; Ma *et al.*, 2018; Ryu *et al.*, 2018], adverse or side effects [Jin *et al.*, 2017; Zitnik *et al.*, 2018] and multi-task learning [Chu *et al.*, 2019]. These methods are relied on the assumption — drugs with similar representations will perform similar DDIs. In the meanwhile, some computational approaches prefer to combine with popular embedding methods [Quan *et al.*, 2018; Le *et al.*, 2018; Quan *et al.*, 2019b], which seek to learn drug representation automatically, and then model DDI by specific operations such as matrix factorization, random walk, and graph neural networks [Hamilton *et al.*, 2017]. Although the methods mentioned above have achieved strong performance, a neglected deficiency is that they model DDI as an independent data sample and do not take their related correlations (e.g., drug-target pairs) into consideration.

On the other hand, owing to the ubiquity of knowledge graph (KG), it has widely led to an influx of research on relation inference and recommendation [Wang *et al.*, 2019], and particularly, recent studies [Celebi *et al.*, 2018; Karim *et al.*, 2019] have used KG for DDI prediction. They both apply KG to machine learning models to extract drug features using various embedding methods. These methods learn node latent embedding directly, but they are limited in obtaining the rich neighborhood information of each entity in KG.

To address the aforementioned limitations of related correlations and neighborhood information existed in KG, our design objective is to automatically capture both high-order structures and semantic relations in KG. Inspired by graph neural networks [Hamilton *et al.*, 2017; Quan *et al.*, 2019a] that try to learn from neighboring information, in this paper we propose a novel end-to-end framework named **Knowledge Graph Neural Network (KGNN)** for DDI prediction. In a nutshell, our framework consists of three major building blocks. The first block is to extract DDIs as well as to construct knowledge graph from collections of datasets. The second block learns drug and its topological neighborhood representations from KG, by using graph neural networks to extract both high-order structures and semantic relations. The fi-

*Corresponding author.

nal block is to predict the potential DDIs after trained samples are sent to a classifier for binary classification. Different from existing methods, our framework needs neither chemical structure nor specialized drug representation, and so it could be more easy-to-use. Additionally, the proposed framework takes advantage of abundant information with structural relations of KG and uses a newly designed graph neural network, which differentiates KGNN from the existing deep learning models. Our contributions are summarized as follows:

- To the best of our knowledge, we are the first to propose knowledge graph neural network (KGNN), an end-to-end framework that explores drugs' topological structures in knowledge graph for potential drug-drug interaction prediction. By extending the receptive field of each entity in KG, KGNN is able to capture high-order relations between drug pairs.
- The KGNN framework has three distinct technical highlights. (i) KGNN exploits topological information of each entity in KG that is beneficial for DDI prediction. (ii) KGNN aggregates all topological neighborhood information from their local receptive to extract both high-order structures and semantic relations. (iii) KGNN adopts graph neural networks compatible with (i) and (ii) to predict the potential DDIs.
- The experimental results on two widely used datasets illustrate the superiority of KGNN over classic and state-of-the-art DDI prediction models.

2 Related Work

Over the years, most of previous works are proposed to predict the potential DDIs by either integrating multiple data sources or combining the popular embedding methods. Different from drug similarity obtained from multiple sources [Vilar *et al.*, 2014], a deep learning framework named DeepDDI [Ryu *et al.*, 2018] is proposed to use molecular structures of drug as inputs, and to predict additional DDI types. In the same line of work, [Yue *et al.*, 2019] integrates several graph embedding methods for DDI task, and with the assistance of knowledge graph, [Karim *et al.*, 2019] models DDI as link prediction. With comparison to the classic and graph embedding methods, our proposed framework is able to automatically extract drug features from the data, and requires neither chemical structure nor specialized drug representation.

Recently, there has been an increasing interest in applying graph neural networks for DDI prediction. To effectively aggregate the feature vectors of its neighbors, different aggregation strategies lead to different variants of GNNs. *Decagon* [Zitnik *et al.*, 2018] applies a relational GNN for modeling polypharmacy side effects. To extract DDIs from text, [Asada *et al.*, 2018] utilizes a graph convolutional network (GCN) to encode the molecular structures. Moreover, attentive multi-view graph auto-encoders [Ma *et al.*, 2018] is integrated into an coherent representation. And a co-attentional mechanism [Deac *et al.*, 2019] is proposed by using side-effect information and the molecular structure of the drugs alone. More recently, *CASTER* [Huang *et al.*, 2019] develops a dictionary learning framework for predicting DDIs given chemical

structures of drugs. Although these methods have achieved strong performance, a neglected deficiency is that they model DDI as an independent data sample and do not consider their multiple related correlations in knowledge graph. The major difference between our work and the literature is that we offer a new perspective for DDI prediction with the assistance of knowledge graph neural network.

Some works that employ knowledge graph (KG) [Wang *et al.*, 2019] are also related to our work. In general, KG is inherently described as a heterogeneous network, and can provide abundant information with structural relations among multiple entities, as well as unstructured semantic relations associated with each node. Different from traditional graphs or networks with only a single relation, KGs normally consist of multiple entities (e.g., drug, target and pathway) and different types of relations that encode heterogeneous information. Owing to its merits, KG has been applied to DDI prediction [Celebi *et al.*, 2018; Karim *et al.*, 2019]. Existing KG-based models often extract drug features using various embedding methods, and learn node latent embedding directly. Thus, they are limited in obtaining the rich neighborhood information of each entity in KG. Compared to this line of methods, our model takes advantage of abundant information with structural relations of KG, and uses a newly designed graph neural network, which differentiates it from the existing KG-based models.

3 The Proposed KGNN

In this section, we first formulate the DDI prediction problem (Section 3.1). Then we provide an overview of the proposed KGNN framework (Section 3.2). After that, we introduce the input of the proposed framework and the KGNN layer, respectively (Section 3.3 and 3.4). Finally, we discuss the DDI prediction with KGNN (Section 3.5).

3.1 Problem Formulation

We formulate the knowledge graph-based DDI prediction problem from two aspects as follows.

DDI matrix. In a typical DDI scenario, we consider a set N_d of drugs, and define the drug-drug interaction matrix $Y \in (0, 1)^{|N_d| \times |N_d|}$, where $|N_d|$ denotes the number of drugs. In the matrix, for each entry $y_{i,j} = 1$ ($i, j \in N_d, j \neq i$), if its value is 1, then it means that drug j interacts with drug i . Note that when $y_{i,j} = 0$, it does not necessarily mean no-interaction between drug pair (i, j) in KG, as it may be the potential interaction while it has not been found before.

Knowledge graph. In addition to the interactions between drug pairs, we consider neighborhood topologies for drug related entities (e.g., targets), in the form of knowledge graph. Formally, we denote a KG by $G = (N_e, N_r)$, which is comprised of entity-relation-entity triples, where N_e (resp., N_r) is the set of entities (resp., relations). For any knowledge graph triple $T_i = (h_i, r_i, t_i)$, where $h_i, t_i \in N_e$ and $r_i \in N_r$, it describes a relationship of type r_i between entity h_i and t_i (the head and tail of a triple, respectively).

Given the DDI matrix Y and the knowledge graph G , we aim to predict whether drug i ($i \in N_d$) has potential interaction with drug j ($j \in N_d, j \neq i$), while such an interaction

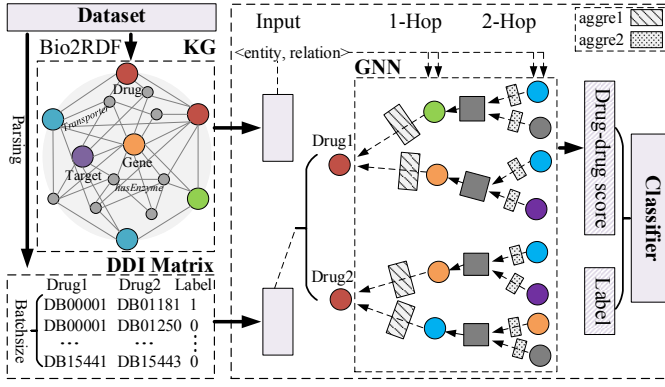


Figure 1: The overview of KGNN.

has not been discovered before. To achieve it, our main task is to learn a prediction function $\hat{y}_{i,j} = \Gamma(i, j | \beta, Y, G)$, where $\hat{y}_{i,j}$ denotes the probability that drug j will interact with drug i , and β denotes the model parameters of function Γ .

3.2 Overview

Figure 1 shows the overview of KGNN. It takes the parsed DDI matrix and knowledge graph obtained from preprocessing (*i.e.*, DDI extraction and KG construction) of dataset as the input. It outputs the interaction value for the drug-drug pair. Remind that the central idea of KGNN is to consider both high-order structure and semantic relation, by using graph neural network to encode the drug and its topological neighborhood information to a distributed representation. Therefore, we design KGNN as a three-step framework for DDI prediction:

1. Extracting DDI and constructing the KG from the collection of datasets;
2. Encoding the drug features and its neighborhood structures (*i.e.*, entities and relations) between drug-drug pairs in KG;
3. Predicting the interaction value based on the encodings learned from the previous step.

Generally, in the **first step** we collect, from the dataset, to extract the DDI data sources that contains the drug-drug pairs, and simultaneously to construct the corresponding KG. In the **second step** we extract features of drug and its neighborhood structures of related entities using KGNN, from the DDI matrix and the constructed knowledge graph. To further predict the interaction value between drug-drug pairs, in the **third step** KGNN output the latent representations of drugs and their neighborhood topologies between drug pairs. Then we compute the score between them, and output a real value interaction. Next, we present the detail of our model.

3.3 DDI Extraction and KG Construction

The first step of KGNN is divided into two parts, including DDI extraction and KG construction.

DDI extraction. The extraction phase is trivial. In brief, one can first download the data from the public drug database (*e.g.*, DrugBank), and then parse the DDI information from

the provided *xml* file, and finally compile an edge list of drug identifiers combinations.

KG construction. We recollect the latest raw data (*e.g.*, DrugBank dataset) from the corresponding portals and convert them into RDF graph by using Bio2RDF tool, which interlinks data that contains multiple types of biological entities (*e.g.*, drug, protein, and disease). Then, the RDF graph is uploaded to a RDF triplestore. After that, the federated SPARQL queries are executed based on the billion triples benchmark to extract selected triples. Therefore, the extracted triples in the form of (subject, predicate, object) are constructed in KG, which shows that the specified relation exists between the subject and the object. Consequently, we can obtain the triples (entity, relation, entity) from the preprocessed RDF file.

Note that, we here create two data sources from DrugBank dataset: (i) the parsed DDI matrix that contains the drug-drug pairs; and (ii) the constructed knowledge graph.

3.4 KGNN Layer

The KGNN layer has guided the development of our method for tackling the topological neighborhood representation problem. Conceptually inspired by spatial-based GNN method [Hamilton *et al.*, 2017], our work here can be regarded as a spatial-based method for a special type of graphs (*i.e.*, knowledge graph). The central idea of the proposed KGNN is to effectively capture high-order neighborhood topologies of drugs between drug pairs in KG. To generate the embedding for a drug’s neighborhood topologies between drug-drug pairs, we apply similar convolutions that aggregate and integrate topological neighborhood information (*i.e.*, entities and relations) from drugs’ local receptive field. Such an operation can learn how to capture the local topological structures, and meanwhile characterize both the semantic information of KG and relations between drugs and related entities.

To understand, consider a candidate pair (i, j) of drug i and j , where $i, j \in N_d$ and $j \neq i$, we denote the set of entities connecting directly to a drug by $N_{neigh}(e)$, where $N_{neigh}(e) \subset N_e$. We compute the score between a drug i and a relation r by a function g as follows: $g(i, r) = C_r^i$, where i ($i \in R^d$) and r ($r \in R^d$) are the drug and relation representations, respectively (the superscript d denotes the dimension of representation), and C_r^i performs the correlation of a relation r to a drug i . To better represent the topological neighborhood structure of a drug i ’s entities e ($e \in N_e$, $\acute{e} \in N_{neigh}(e)$), we also compute the linear combination of each entity e ’s neighborhood: $e_{N_{neigh}(e)}^i = \sum_{\acute{e} \in N_{neigh}(e)} \tilde{C}_{r_{e,\acute{e}}}^i e$, where $\tilde{C}_{r_{e,\acute{e}}}^i$ denotes the normalized drug-relation score, and e ($e \in R^d$) is the representation of entity e .

Local receptive. Inspired by GNN approach [Hamilton *et al.*, 2017] that simply examines H -hop/order graph neighborhoods, our solution also defines H -hop neighborhoods (or equally the depth of receptive field). Note that, besides the immediate neighbors, we also extend KGNN to 2-hop/layer to extract both high-order structures and semantic relations.

Neighborhood sampling. In a real-world knowledge graph, the size of $N_{neigh}(e)$ may vary significantly over all

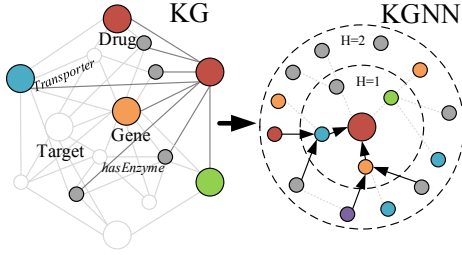


Figure 2: A two-layer receptive field (different entities with different color) of the given drug (red) in a KG.

entities. To consider efficiency and the fixed computation pattern of each batch data, we uniformly sample a fixed size set $S(e)$ instead of using its full neighbors $N_{neighbor}(e)$. Then, we can obtain the neighborhood representation $e_{S(e)}^i$ of entity e . Technically, $S(e)$ may contain duplicates if $|N_{neighbor}(e)| < k$, where k denotes the number of neighbor sampling size of each depth. Note that, in our paper $S(e)$ is also called the receptive field of entity in a single layer, as the final representation \hat{e}^i is sensitive to these locations. Figure 2 gives an example of a 2-layer receptive field, we here set the neighbor sampling size $k=2$ in each layer.

Aggregation. The final step in a KGNN layer is to aggregate the entity representation e and its neighborhood representation $e_{S(e)}^i$ into a single vector. We implement similar multiple types of aggregations [Wang *et al.*, 2019] as $aggre$ in KGNN as follows: $aggre_{sum} = \sigma(W \cdot (e + e_{S(e)}^i) + b)$, $aggre_{concat} = \sigma(W \cdot concat(e, e_{S(e)}^i) + b)$ and $aggre_{neighbor} = \sigma(W \cdot e_{S(e)}^i + b)$. Aggregation is a key step in KGNN, as the representation of a drug is bound up with its topological neighborhoods by aggregation. We will evaluate these aggregations in the ablation study (Section 4.3).

Algorithm 1 shows the pseudo-codes of applying KGNN for topological neighborhood representation between given drug pairs. H represents the depth of receptive field or the number of aggregation iterations. Specifically, for a given drug-drug pair (i, j) (Line 2), we first calculate the receptive field RF of entity e between drug-entity pair (i, e) (also for (j, e)) in an iterative manner (Line 3), and we consider $RF(e) = RF(e) \cup S(e)$ in each depth. Meanwhile, we calculate the entity representation e^i and e^j of drug i and j at the current depth, respectively (Line 4). Then the aggregation is repeated h ($h \in [1, H]$) times (Line 5): in iteration h for drug i , we calculate the neighborhood representation $e_{S(e)}^i$ of each entity $e \in RF[h]$ (Line 7), then aggregate it with its own representation $e^i[h-1]$ to obtain the one to be used at the next iteration (Line 8). The final h -order entity representation is denoted as e^i (Line 11), which is fed into a function τ together with drug representation i and j for calculating the score of drug and entities (Line 12), and finally obtain the predicted probability of $\hat{y}_{i,j}$.

3.5 Drug-drug Interaction Prediction

In this study, we view DDI prediction as a binary classification task by predicting the interaction values. With the representation learned from the previous sections, we can inte-

Algorithm 1 KGNN algorithm

Input: DDI matrix Y ; knowledge graph $G(N_e, N_r)$; neighborhood field $S(e)$ ($e \in N_e$); hyper-parameter: $H, k, g(\cdot), aggre(\cdot), \tau(\cdot), f(\cdot)$

Output: $\Gamma(i, j | \beta, Y, G)$

```

1: while KGNN not converge do
2:   for  $(i, j) \in Y$  do
3:      $\{RF[h]\}_{h=0}^H \leftarrow \text{Receptive-Field}(e)$ ;
4:      $e^i[0] \leftarrow e, \forall e \in RF[0]$ ;
5:     for  $h=1, \dots, H$  do
6:       for  $e \in RF[h]$  do
7:          $e_{S(e)}^i[h-1] \leftarrow \sum_{e \in S(e)} \tilde{C}_{r_{e,e}}^i \hat{e}^i[h-1]$ ;
8:          $e^i[h] \leftarrow aggre(e_{S(e)}^i[h-1], e^i[h-1])$ ;
9:       end for
10:      end for
11:       $\hat{e}^i \leftarrow e^i[h]$ ;
12:      Calculate the score  $\hat{y}_i = \tau(i, \hat{e}^i), \hat{y}_j = \tau(j, \hat{e}^j)$ ;
13:      Calculate predicted probability  $\hat{y}_{i,j} = f(\hat{y}_i, \hat{y}_j)$ ;
14:      Update parameters  $\beta$ ;
15:    end for
16:  end while
17: return  $\Gamma$ 

```

grate all the information from drugs and topological neighborhood to predict the interaction value between drug-drug pairs. In brief, we concatenate all the representations and feed them to output the interaction value. More precisely, for the KGNN layer, we use multiple aggregation methods to update the neighborhood structures of drugs in KG considering their topology. Besides, we use Rectified Linear Unit (ReLU) as the activation function, which is widely adopted in deep learning research. Given a set of drug-drug pairs and the ground-truth interaction values in the training dataset, we can use the binary cross-entropy as the loss function: $Loss = \sum_{(i,j) \in Y} (y_{i,j} \log \hat{y}_{i,j} + (1 - y_{i,j}) \log(1 - \hat{y}_{i,j}))$, where $\hat{y}_{i,j}$ is the predicted value, $y_{i,j}$ is the ground-truth value, and Y represents the set of drug-drug pairs.

4 Experiment

4.1 Datasets and Settings

We evaluate our proposed KGNN¹ by using two datasets. (1) **DrugBank**: we parse the verified DDIs of the provided profile from DrugBank (V5.1.4) and compile an edge list of drug identifier combinations, which obtains 2,578 approved small molecule drugs and 612,388 unique approved DDIs spanning 13,339 drugs; (2) **KEGG-drug**: we parse the sources from KEGG and map it to DrugBank identifiers (IDs), which results in 1,925 approved drugs and 56,983 approved interactions spanning 11,147 drugs and 324,183 interactions respectively.

Different from the preprocessing of [Karim *et al.*, 2019] that it integrates multiple datasets into a unified KG, which will result in the loss of mutually exclusive information. Specifically, we collect and construct the KG for each dataset

¹<https://github.com/xzenglab/KGNN>

| | DrugBank | KEGG-drug |
|----------------|-----------|-----------|
| Drugs | 2,578 | 1,925 |
| Interactions | 612,388 | 56,983 |
| Entities | 2,129,712 | 129,910 |
| Relation Types | 72 | 167 |
| KG Triples | 7,852,852 | 362,870 |

Table 1: The statistic of two widely used datasets.

to keep its original integrity. Meanwhile, the KG should not contain any explicit information about drug-drug interactions, we exclude the information in the form of *url: ddi-interactor-in* and *url: Drug-Drug-Interaction* from DrugBank and KEGG-drug datasets, respectively. The basic statistics of the used datasets are summarized in Table 1.

Baselines. We compare KGNN against a variety of baselines which can be categorized as follow.

- (1) **Matrix Factorization:** We choose two most representative MF-based methods, Laplacian [Belkin and Niyogi, 2003] and GraRep [Cao *et al.*, 2015]. They aim to factorize the matrix of input data into lower dimensional matrices.
- (2) **Random Walk:** RW-based methods learn node representations by generating node sequences and by using different random walks strategies in graphs. We choose two representative RW-based methods, namely DeepWalk [Perozzi *et al.*, 2014] and Struc2Vec [Ribeiro *et al.*, 2017].
- (3) **Neural Network:** We choose three most representative NN-based methods including LINE [Tang *et al.*, 2015], SDNE [Wang *et al.*, 2016] and GAE [Kipf and Welling, 2016]. They adopt different neural architectures and different kinds of graph information as input to learn node embedding.
- (4) **DeepDDI:** [Ryu *et al.*, 2018] develops a deep learning-based method that reduces the dimension of drug features, based on a principal component analysis.
- (5) **KG-ddi:** [Karim *et al.*, 2019] is a KG-based method for DDI prediction, by encompassing over 12,000 drug features from integrated knowledge graph of multiple data sources, and it adopts a CNN-LSTM model using the embeddings.

In the experiment, for the comparison with graph embedding methods (e.g., MF/RW/NN-based), we re-implement BioNEV [Yue *et al.*, 2019] on our machine, and we generate the embedding of dimension 100 as drug feature for each dataset. For the comparison with DeepDDI, we construct the DNN architecture and modify the output of the original implementation from multi-label to a binary class. For KGE-ddi, we re-implement the RDF2Vec to generate 300-dimension embeddings of each dataset as drug features and load pre-trained model for prediction. The other settings are as the same as original work.

Metrics. We denote $y_{i,j}$, $\hat{y}_{i,j}$ as the ground truth, and predicted values for drug-drug pairs dataset, respectively. We evaluate the prediction performance using several metrics, including ACC, AUPR, AUC-ROC, F1 scores.

Experimental settings. We use Bio2RDF-based method to construct the knowledge graph for each dataset. For both datasets, we randomly divide all approved DDIs as positive samples into training, validation and testing sets in a 8/1/1 ra-

| Parameter | Setting | Parameter | Setting |
|---------------|---------|-------------------|---------|
| Batch size | 4096 | Dimension | 32 |
| Learning rate | 1e-2 | Number of depth | 2 |
| L2 weight | 1e-7 | Neighborhood size | 16 |

Table 2: The hyper-parameter settings for the two datasets.

tio, and randomly sample the complement set of positive samples as negative samples, with an equal number of positive and negative samples in all phase. We adopt Adam algorithm to optimize all trainable parameters through a random search and 5-fold cross-validation tests. Meanwhile, we set the number of epoch 50 for training, and other hyper-parameter settings are provided in Table 2, which are optimized by AUC-ROC on a validation set.

4.2 Results and Analysis

In this section, we compare the performance of the proposed method with the baselines. Table 3 reports the average ACC, AUPR, AUC-ROC, and F1 scores across 5 runs on DrugBank and KEGG-drug datasets, respectively. From this table, we found that KGNN significantly outperforms the baselines across the two datasets. More specifically, KGNN achieved at least 2.76% on ACC, 4.78% on AUPR, 2.13% on AUC-ROC and 2.34% on F1 higher performance than other methods. For example, on DrugBank dataset, KGNN_{concat} achieved a ACC score of 0.9561 with 11.18% absolute gain compared to GreRep (the second best method). This is because KGNN explores both drug features and related entities in knowledge graph, while the others like Laplacian, DeepWalk and struc2vec only learn from similar drug features. Compared with NN-based methods, which leverages graph features with similar connections for DDI prediction, KGNN also achieved superior results (e.g., 19.62% improvement of AUPR on KEGG-drug dataset). This is due to the fact that the drug embeddings in KGNN can better capture the semantic similarity of relations than the graph embeddings used in NN-based methods. Moreover, with comparison to DeepDDI and KG-ddi, KGNN achieved stable performances across datasets and our three variants all achieved similar results. This is a very encouraging result. The reason could be that (i) compared to DeepDDI, our method jointly considers topological neighborhood structures and related entities in knowledge graph, which benefits to the performance; (ii) compared to KG-ddi, our model incorporates GNN model to obtain the topological neighborhood representations of drug and related entities, which can obtain more high-order structures and semantic relations than embedding-based methods for modeling drugs between drug pairs. Overall, it is essentially a non-trivial achievement in terms of DDI prediction.

4.3 Ablation Study

As mentioned before, existing models for DDI prediction have leveraged multiple data sources to learn the representation for drug, while they often ignore the neighborhood information and related correlations in knowledge graph. Therefore, this work considers topological neighborhood structures to learn the interaction between drug pairs. More precisely, the core idea of KGNN is to fully leverage both high-order

| Metrics | Methods | MF-based | | RW-based | | NN-based | | DL-based | KG-based | KGNN _x | | | |
|---------|---------|-----------|--------|----------|-----------|----------|--------|----------|----------|-------------------|-----------------|------------|---------------|
| | | Laplacian | GreRep | DeepWalk | struc2vec | LINE | SDNE | GAE | DeepDDI | KG-ddi | <i>neighbor</i> | <i>sum</i> | <i>concat</i> |
| ACC | | 0.7183 | 0.8443 | 0.8349 | 0.7882 | 0.8280 | 0.8303 | 0.7491 | 0.8123 | 0.7867 | 0.9354 | 0.9538 | 0.9561 |
| | | 0.8029 | 0.8718 | 0.8547 | 0.8436 | 0.8655 | 0.8674 | 0.7586 | 0.8229 | 0.8154 | 0.8846 | 0.8882 | 0.8950 |
| AUPR | | 0.7533 | 0.9115 | 0.9070 | 0.8672 | 0.8915 | 0.8782 | 0.7403 | 0.9193 | — | 0.9801 | 0.9890 | 0.9892 |
| | | 0.8261 | 0.9055 | 0.9011 | 0.8861 | 0.8968 | 0.8967 | 0.7571 | 0.8442 | — | 0.9207 | 0.9247 | 0.9533 |
| AUC-ROC | | 0.7966 | 0.9230 | 0.9181 | 0.8735 | 0.9092 | 0.9029 | 0.8085 | 0.9261 | 0.7867 | 0.9824 | 0.9902 | 0.9912 |
| | | 0.8736 | 0.9305 | 0.9208 | 0.9086 | 0.9264 | 0.9249 | 0.8334 | 0.8994 | 0.8154 | 0.9418 | 0.9453 | 0.9518 |
| F1 | | 0.7270 | 0.8461 | 0.8357 | 0.7962 | 0.8318 | 0.8373 | 0.7889 | 0.8466 | 0.7843 | 0.9366 | 0.9544 | 0.9566 |
| | | 0.8079 | 0.8748 | 0.8570 | 0.8476 | 0.8695 | 0.8704 | 0.7888 | 0.7966 | 0.8152 | 0.8869 | 0.8909 | 0.8982 |

Table 3: Performance of KGNN against comparative approaches. First/second row of each method corresponds to results reported on **DrugBank** and **KEGG-drug** dataset respectively. Note that the standard deviation scores is omitted due to the space limitation.

structures and semantic relations from knowledge graph, by using multiple aggregation operations, to better learn the drug and entity representations. To study the effectiveness of the central idea, we implemented several variants of our model, named $KGNN_x$, where the subscript x attached denotes different aggregation operations (recall Section 3.4). The last three columns in Table 3 show the comparison results. It can be seen that $KGNN_{concat}$ outperforms other variants in all metrics. Moreover, as $KGNN_{neighbor}$ only considers neighborhood information of knowledge graph, it performs worse than $KGNN_{concat}$ and $KGNN_{sum}$ in both dataset. This shows that combining topological neighborhood representations with semantic relations between drug and related entities is benefit to improving the DDI prediction performance.

4.4 Case Study

In this section, we examine the influence of several key hyper-parameters on the performance of proposed KGNN. We fixed other parameters when studying one of parameters. Other settings are kept the same as that in Section 4.1. Figure 3 reports the average and standard deviation of ACC, AUPR, AUC-ROC, and F1 scores on the KEGG-drug dataset.

Impact of neighborhood size. Firstly, we vary the neighborhood size k and observe that KGNN achieves the best performance when $k = 16$. This is because a too small k does not have enough capacity to incorporate neighborhood information, while a too large value is prone to be misled by noises.

Impact of depth of receptive field. Secondly, we investigate the influence of depth of receptive field H by varying from 1 to 6 (Out of system memory when H is equal to 7). We observe that the performance of our model in all metrics decreases starting from $H = 3$, as a larger H brings massive noises to the model. This is also in accordance with our intuition, since a long relation chain makes little sense when inferring drug-entity similarities. This implies that an H of 1 or 2 is often enough for real cases, according to the experiment results.

Impact of dimension of embedding. Lastly, we examine the influence of dimension of embedding d by varying from 8 to 512. The result is rather intuitive: it can boost the performance with a proper d that can encode enough information of drugs and entities from KG, while a too large value adversely suffers from over-fitting.

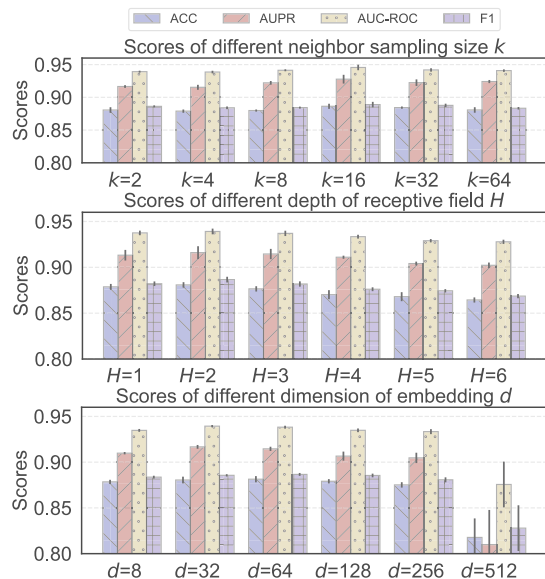


Figure 3: Results of KGNN with varying size of k , H , and d .

5 Conclusion

In this paper, we propose a novel model, called KGNN (knowledge graph neural network), for drug-drug interaction prediction. KGNN extends spatial-based GNN approaches to the knowledge graph by multiple aggregating neighborhood information selectively, which is able to learn both topological structure information and semantic relation of knowledge graph, as well as the neighborhood of drug and related entities. We implement the proposed method and conduct experimental comparisons on two widely used datasets. The experimental results show that KGNN outperforms the classic and state-of-the-art DDI prediction models.

Acknowledgments

This work is supported in part by the National Key R&D Program of China (No. 2018YFB0204302), the NSFC (No. 61972425, U1811264, 61872309, 61772570), the Pearl River S&T Nova Program of Guangzhou (No. 201806010056), and Guangdong NSF for Distinguished Young Scholar (No. 2018B030306025). X. Lin is supported by the China Scholarships Council (No. 201906130128). X. Lin thanks Philip S. Yu for hosting his visit at University of Illinois at Chicago.

References

- [Asada *et al.*, 2018] Masaki Asada, Makoto Miwa, and Yutaka Sasaki. Enhancing drug-drug interaction extraction from texts by molecular structure information. In *ACL*, pages 680–685, 2018.
- [Belkin and Niyogi, 2003] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.
- [Cao *et al.*, 2015] Shaosheng Cao, Wei Lu, and Qionikai Xu. Grarep: Learning graph representations with global structural information. In *CIKM*, pages 891–900, 2015.
- [Celebi *et al.*, 2018] Remzi Celebi, Erkan Yasar, Huseyin Uyar, Ozgur Gumus, Oguz Dikenelli, and Michel Dumontier. Evaluation of knowledge graph embedding approaches for drug-drug interaction prediction using linked open data. In *SWAT4HCLS 2018*, pages 1–10, 2018.
- [Chu *et al.*, 2019] Xu Chu, Yang Lin, Yasha Wang, Leye Wang, Jiangtao Wang, and Jingyue Gao. Mlrda: a multitask semi-supervised learning framework for drug-drug interaction prediction. In *IJCAI*, pages 4518–4524, 2019.
- [Deac *et al.*, 2019] Andreea Deac, Yu-Hsiang Huang, Petar Veličković, Pietro Liò, and Jian Tang. Drug-drug adverse effect prediction with graph co-attention. *arXiv*, 2019.
- [Hamilton *et al.*, 2017] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *NeurIPS*, pages 1024–1034, 2017.
- [Huang *et al.*, 2019] Kexin Huang, Cao Xiao, Trong Nghia Hoang, Lucas M Glass, and Jimeng Sun. Caster: Predicting drug interactions with chemical substructure representation. *arXiv preprint arXiv:1911.06446*, 2019.
- [Jin *et al.*, 2017] Bo Jin, Haoyu Yang, Cao Xiao, Ping Zhang, Xiaopeng Wei, and Fei Wang. Multitask dyadic prediction and its application in prediction of adverse drug-drug interaction. In *AAAI*, 2017.
- [Karim *et al.*, 2019] Md Rezaul Karim, Michael Cochez, Joao Bosco Jares, Mamta Uddin, Oya Beyan, and Stefan Decker. Drug-drug interaction prediction based on knowledge graph embeddings and convolutional- lstm network. In *ACM BCB*, pages 113–123, 2019.
- [Kipf and Welling, 2016] Thomas N Kipf and Max Welling. Variational graph auto-encoders. In *NeurIPS*, 2016.
- [Le *et al.*, 2018] Yuquan Le, Zhi-Jie Wang, Zhe Quan, Jiawei He, and Bin Yao. Acv-tree: A new method for sentence similarity modeling. In *IJCAI*, pages 4137–4143, 2018.
- [Lin *et al.*, 2019] Xuan Lin, Zhe Quan, Zhi-Jie Wang, Huang Huang, and Xiangxiang Zeng. A novel molecular representation with bigru neural networks for learning atom. *Briefings in bioinformatics*, 11 2019.
- [Ma *et al.*, 2018] Tengfei Ma, Cao Xiao, Jiayu Zhou, and Fei Wang. Drug similarity integration through attentive multi-view graph auto-encoders. In *IJCAI*, 2018.
- [Perozzi *et al.*, 2014] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *SIGKDD*, pages 701–710, 2014.
- [Quan *et al.*, 2018] Zhe Quan, Xuan Lin, Zhi-Jie Wang, Yan Liu, Fan Wang, and Kenli Li. A system for learning atoms based on long short-term memory recurrent neural networks. In *BIBM*, pages 728–733, 2018.
- [Quan *et al.*, 2019a] Zhe Quan, Yan Guo, Xuan Lin, Zhi-Jie Wang, and Xiangxiang Zeng. Graphcpi: Graph neural representation learning for compound-protein interaction. In *BIBM*, pages 717–722, 2019.
- [Quan *et al.*, 2019b] Zhe Quan, Zhi-Jie Wang, Yuquan Le, Bin Yao, Kenli Li, and Jian Yin. An efficient framework for sentence similarity modeling. *TASLP*, 27(4):853–865, 2019.
- [Ribeiro *et al.*, 2017] Leonardo FR Ribeiro, Pedro HP Saverese, and Daniel R Figueiredo. struc2vec: Learning node representations from structural identity. In *SIGKDD*, pages 385–394, 2017.
- [Ryu *et al.*, 2018] Jae Yong Ryu, Hyun Uk Kim, and Sang Yup Lee. Deep learning improves prediction of drug-drug and drug-food interactions. *PNAS*, 115(18):E4304–E4311, 2018.
- [Tang *et al.*, 2015] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *WWW*, pages 1067–1077, 2015.
- [Vilar *et al.*, 2012] Santiago Vilar, Rave Harpaz, Eugenio Uriarte, Lourdes Santana, Raul Rabadan, and Carol Friedman. Drug-drug interaction through molecular structure similarity analysis. *JAMIA*, 19(6):1066–1074, 2012.
- [Vilar *et al.*, 2014] Santiago Vilar, Eugenio Uriarte, Lourdes Santana, Tal Lorberbaum, George Hripcsak, Carol Friedman, and Nicholas P Tatonetti. Similarity-based modeling in large-scale prediction of drug-drug interactions. *Nature protocols*, 9(9):2147, 2014.
- [Wang *et al.*, 2016] Daixin Wang, Peng Cui, and Wenwu Zhu. Structural deep network embedding. In *SIGKDD*, pages 1225–1234, 2016.
- [Wang *et al.*, 2019] Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. Knowledge graph convolutional networks for recommender systems. In *WWW*, pages 3307–3313, 2019.
- [Yue *et al.*, 2019] Xiang Yue, Zhen Wang, Jingong Huang, Srinivasan Parthasarathy, Soheil Moosavinasab, Yungui Huang, Simon M Lin, Wen Zhang, Ping Zhang, and Huan Sun. Graph embedding on biomedical networks: Methods, applications, and evaluations. *Bioinformatics*, 2019.
- [Zitnik *et al.*, 2018] Marinka Zitnik, Monica Agrawal, and Jure Leskovec. Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics*, 34(13):i457–i466, 2018.