Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

Data and Text Mining

Graph Embedding on Biomedical Networks: Methods, Applications, and Evaluations

Xiang Yue [1,*], Zhen Wang [1], Jingong Huang [2], Srinivasan Parthasarathy [1], Soheil Moosavinasab [3], Yungui Huang [3], Simon M. Lin [3], Wen Zhang [4], Ping Zhang [1,5], and Huan Sun [1,*]

[1]Department of Computer Science and Engineering, The Ohio State University, Columbus, OH, USA, [2]Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH, USA, [3]Research Information Solutions and Innovation, The Research Institute at Nationwide Children's Hospital, Columbus, OH, USA, [4]College of Informatics, Huazhong Agricultural University, Wuhan, Hubei, China, [5]Department of Biomedical Informatics, The Ohio State University, Columbus, OH, USA,

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Graph embedding learning which aims to automatically learn low-dimensional node representations has drawn increasing attention in recent years. To date, most recent graph embedding methods are mainly evaluated on social and information networks and have yet to be comprehensively studied on biomedical networks under systematic experiments and analyses. On the other hand, for a variety of biomedical network analysis tasks, traditional techniques such as matrix factorization (which can be seen as one type of graph embedding methods) have shown promising results, and hence there is a need to systematically evaluate more recent graph embedding methods (e.g., random walk-based and neural network-based) in terms of their usability and potential to further the state-of-the-art.

Results: We conduct a systematic comparison of existing graph embedding methods on three important biomedical *link prediction* tasks: drug-disease association (DDA) prediction, drug-drug interaction (DDI) prediction, protein-protein interaction (PPI) prediction, and one *node classification* task, i.e., classifying the semantic types of medical terms (nodes). Our experimental results demonstrate that the recent graph embedding methods are generally more effective than traditional embedding methods. Besides, compared with two state-of-the-art methods for DDAs and DDIs predictions, graph embedding methods without using any biological features achieve very competitive performance. Moreover, we summarize the experience we have learned and provide guidelines for properly selecting graph embedding methods and setting their hyper-parameters.

Availability: We develop an easy-to-use Python package with detailed instructions, BioNEV, available at: `https://github.com/xiangyue9607/BioNEV`, including all source code and datasets, to facilitate studying various graph embedding methods on biomedical tasks.

Contact: yue.149@osu.edu, sun.397@osu.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Graphs (a.k.a. networks) have been widely used to represent biomedical entities (as nodes) and their relations (as edges). Analyzing biomedical graphs can greatly benefit various important biomedical tasks, such as predicting potential drug indications (a.k.a. drug repositioning) based on drug-disease association graphs (Gottlieb *et al.*, 2011), detecting long non-coding RNA (lncRNA) functions based on lncRNA-protein interaction networks (Zhang *et al.*, 2018f), and assisting clinical decision making via disease-symptom graphs (Rotmensch *et al.*, 2017).

In order to analyze graph data, a surge of graph embedding (a.k.a. network embedding or graph representation learning) methods (Perozzi *et al.*, 2014; Tang *et al.*, 2015; Grover and Leskovec, 2016; Ribeiro *et al.*, 2017) have been proposed, where the goal is to learn a low-dimensional feature representation for each node in the graph. The feature representations are generally learned to preserve the structural information

* Preprint submitted to *Bioinformatics*.

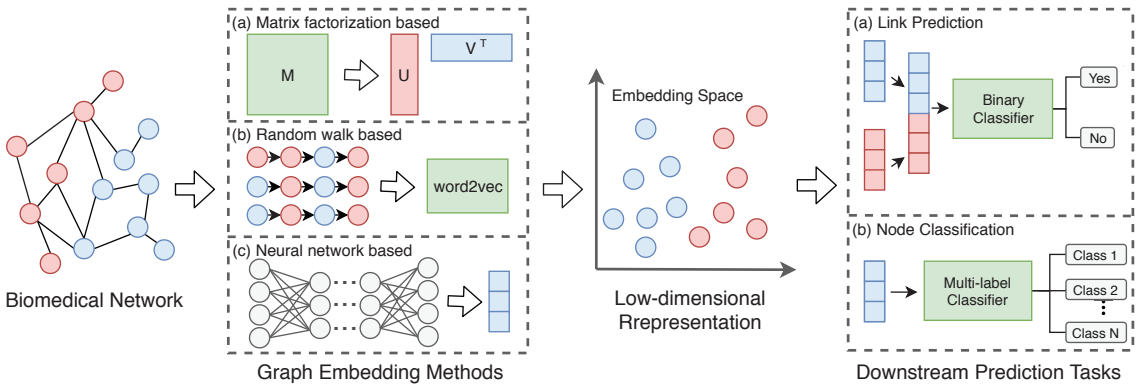


Fig. 1: Pipeline for applying graph embedding methods to biomedical tasks. Low-dimensional node representations are first learned from biomedical networks by graph embedding methods and then used as features to build specific classifiers for different tasks. For (a) matrix factorization-based methods, they take a data matrix (e.g., adjacency matrix) as the input to learn embeddings through matrix factorization. For (b) random walk-based methods, they first generate sequences of nodes through random walks and then feed the sequences into the word2vec model (Mikolov *et al.*, 2013) to learn node representations. For (c) neural network-based methods, their architectures and inputs vary from different models (see Section 2 for details).

of graphs, and thus can be used as features in building machine learning models for various downstream tasks, such as link prediction, community detection, node classification, and clustering (Wang *et al.*, 2018; Xie *et al.*, 2016). However, to date, these advanced approaches are mainly evaluated on non-biomedical networks such as social networks, citation networks, and user-item networks, and only a few studies have conducted evaluations on protein-protein interaction networks (Grover and Leskovec, 2016; Goyal and Ferrara, 2018).

Although there exist models developed for biomedical tasks that involve the general idea of graph embedding, many of them still focus on traditional techniques such as Locally Linear Embedding (LLE) (Zhang *et al.*, 2017a,b), Laplacian Eigenmap (LE) (Ezzat *et al.*, 2017) and Matrix Factorization (MF) (Zhang *et al.*, 2018d,e). Given that the recent graph embedding methods have been demonstrated more effective than those traditional methods in a wide range of non-biomedical tasks (Perozzi *et al.*, 2014; Tang *et al.*, 2015; Grover and Leskovec, 2016; Wang *et al.*, 2016), we conduct this work to investigate the effectiveness and potential of advanced graph embedding methods on biomedical tasks. Fig. 1 summarizes the pipeline for applying various graph embedding methods to biomedical tasks (e.g., link prediction and node classification).

Specifically, we first provide an overview of existing graph embedding methods and conduct a systematic comparison on three important biomedical prediction tasks: drug-disease association (DDA) prediction, drug-drug interaction (DDI) prediction, and protein-protein interaction (PPI) prediction. These three tasks focus on *link prediction* task, which predicts if there is a link (i.e., interaction/association/edge) between two nodes. In contrast to link prediction tasks, there are few widely studied *node classification* tasks in the biomedical literature. Here, we formulate one to evaluate graph embeddings for node classification: *Given a medical term-term co-occurrence graph where terms and their co-occurrence statistics are extracted from clinical notes in Electronic Medical Records (EMRs), we propose to classify the semantic types of each medical term.* This task aims to infer the semantic type information for free-form text terms to bridge the gap between unstructured text and structured knowledge in the medical domain, which is very important and meaningful to study.

For the above 4 tasks, we compile 5 datasets from commonly used biomedical databases and select 11 graph embedding methods (including both traditional and more recent methods) for comprehensive comparisons. By benchmarking them, we demonstrate that in general, the recently proposed graph embedding methods are more effective than the traditional embedding methods in various biomedical tasks. Moreover, we compare the graph embedding methods with two recent computational methods that are specially designed and among state-of-the-arts for DDAs and DDIs prediction, and demonstrate that the graph embedding methods can achieve very competitive or further improve the performance while being very general. Additionally, we provide insightful observations as well as suggestions for selecting proper graph embedding methods and setting their hyper-parameters for biomedical prediction tasks. Furthermore, we discuss new trends and directions (e.g., transfer learning in biomedical graph embedding) to encourage future work.

Although there are some existing studies that review the technical details of various graph embedding methods (Hamilton *et al.*, 2017; Zhang *et al.*, 2018a) and discuss the applications of graph embedding methods on biomedical graphs (Su *et al.*, 2018), few have systematically compared their performance on biomedical datasets.

To summarize, our contributions are threefold:

- We provide an overview of different types of graph embedding methods, and discuss how they can be used in 3 important biomedical *link prediction* tasks: DDAs, DDIs and PPIs prediction, and a meaningful biomedical *node classification* task, i.e., classify the semantic types of medical terms based on the co-occurrence graph constructed from clinical notes.
- We compile 5 benchmark datasets for all the above prediction tasks and use them to systematically evaluate 11 representative graph embedding methods selected from different categories (i.e., 5 matrix factorization-based, 3 random walk-based, 3 neural network-based). We discuss our observations from extensive experiments and provide some insights and guidelines for how to choose embedding methods (including their hyper-parameter settings).
- We develop an easy-to-use Python package with detailed instructions, BioNEV (Biomedical Network Embedding Evaluation), available at: `https://github.com/xiangyue9607/BioNEV`, including all source code and datasets, to facilitate studying various graph embedding methods on biomedical tasks.

2 Overview of Graph Embedding Methods

In this section, we provide a brief overview of different graph embedding methods, which are categorized into 3 groups: matrix factorization-based, random walk-based, and neural network-based (Fig. 1 provides a high-level illustration).

2.1 Matrix factorization-based methods

Matrix factorization has been widely adopted for data analysis. Essentially, it aims to factorize a data matrix into lower dimensional matrices and still keeps the manifold structure and topological properties hidden in the original data matrix. Pioneer work in this category dates back to the early 2000s, such as Isomap (Tenenbaum *et al.*, 2000), Locally Linear

Embedding (Roweis and Saul, 2000), and Laplacian Eigenmaps (Belkin and Niyogi, 2002). Traditional matrix factorization has many variants, such as Singular Value Decomposition (SVD) and Graph Factorization (GF) (Ahmed *et al.*, 2013). And they often focus on factorizing the 1st-order data matrix (e.g., adjacency matrix).

More recently, researchers focus on designing various high-order data proximity matrices to preserve the graph structure and propose various matrix factorization-based graph embedding learning methods. For example, GraRep (Cao *et al.*, 2015) considers the high-order proximity of the network and designs k-step transition probability matrices for factorization. HOPE (Ou *et al.*, 2016) also considers the high-order proximity. But different from GraRep, it adopts some well-known network similarity measures such as Katz Index and Common Neighbors to preserve network structures.

2.2 Random walk-based methods

Inspired by the word2vec (Mikolov *et al.*, 2013) model, a popular word embedding technique from Natural Language Processing (NLP), which tries to learn word representations from sentences, random walk-based methods are developed to learn node representations by generating "node sequences" through random walks in graphs. Specifically, given a graph and a starting node, random walk-based methods first randomly select one of the node's neighbors and then move to this neighbor. This procedure is repeated to obtain node sequences. Then the word2vec model is adopted to learn embeddings from sequences of nodes. In this way, neighborhood similarity and structural information can be preserved into latent features.

One of the initial works in this category is DeepWalk (Perozzi *et al.*, 2014) which performs truncated random walks on a graph. Compared to DeepWalk, node2vec (Grover and Leskovec, 2016) adopts a flexible biased random walk procedure that smoothly combines Breadth-first Sampling (BFS) and Depth-first Sampling (DFS) to generate node sequences. Further, struc2vec (Ribeiro *et al.*, 2017) is proposed for better modeling the structural identity (e.g., nodes in the network may perform similar functions). Specifically, struct2vec first constructs a multi-layer weighted graph that encodes the structural similarity between nodes where each layer k is defined by using the k-hop neighborhoods of the nodes. Then DeepWalk is performed on the multilayer graph to learn node representations in which nodes with high structural similarity are close to each other in the embedding space.

2.3 Neural network-based methods

Recent years have witnessed the success of neural network models in many fields. Various neural networks also have been introduced into graph embedding areas, such as Multilayer Perceptron (MLP) (Tang *et al.*, 2015), autoencoder (Cao *et al.*, 2016; Wang *et al.*, 2016; Kipf and Welling, 2016), Generative Adversarial Network (GAN) (Wang *et al.*, 2017a) and Graph Convolutional Network (GCN) (Kipf and Welling, 2016, 2017). Different methods adopt different neural architectures and use different kinds of graph information as input. For example, LINE (Tang *et al.*, 2015) directly models node embedding vectors by approximating the 1st-order proximity and 2nd-order proximity of nodes, which can be seen as a single-layer MLP model. DNGR (Cao *et al.*, 2016) applies the stacked denoising autoencoders on the positive pointwise mutual information (PPMI) matrix to learn deep low-dimensional node embeddings. SDNE (Wang *et al.*, 2016) adopts a deep autoencoder to preserve the second-order proximity by reconstructing the neighborhood structure of each node; meanwhile, it also incorporates Laplacian Eigenmaps proximity measure into the learning framework to exploit the first-order proximity. GAE (Kipf and Welling, 2016) utilizes a Graph Convolutional Networks (GCNs) encoder and an inner product decoder to learn node embeddings. GraphGAN (Wang *et al.*, 2017a) adopts Generative Adversarial Networks (GANs) to model the connectivity of nodes. The GAN framework includes a generator and a discriminator where the generator approximates the true connectivity distribution over all other nodes and generates fake samples, while the discriminator model detects whether the sampled node is from ground truth or generated by the generator.

3 Graph Embedding on Biomedical Networks

While graph embedding techniques have been widely used in many open-domain data mining tasks, they are not thoroughly evaluated on *biomedical graphs*. In this section, we select 11 representative graph embedding methods (5 matrix factorization-based, 3 random walk-based, 3 neural network-based), and evaluate how they perform on 3 popular biomedical *link prediction* tasks: drug-disease association prediction, drug-drug interaction prediction, protein-protein interaction prediction. Moreover, we discuss a meaningful *node classification* task, which is to classify the semantic types of medical terms based on their co-occurrence graph extracted from clinical notes, for further graph embedding methods evaluation.

3.1 Link prediction in biomedical networks

Discovering new interactions (links) is one of the most important tasks in the biomedical area. A considerable amount of efforts has been devoted to developing computational methods to predict potential interactions in various biomedical networks, such as the DDA network (Zhang *et al.*, 2017a), DDI network (Zhang *et al.*, 2015), and PPI network (Wang *et al.*, 2014). Developing such computational methods can help generate hypotheses of potential associations or interactions in biological networks.

The link prediction task can be formulated as: *Given a set of biomedical entities and their known interactions, we aim to predict other potential interactions between entities*. Traditional methods in the biomedical field put much effort on feature engineering which tries to develop biological features (e.g., chemical substructures, gene ontology) or graph properties (e.g., topological similarities). After that, supervised learning methods (e.g., SVM, Random Forest) or semi-supervised graph inference model (e.g., label propagation) are utilized to predict potential interactions. The assumption behind these methods is that entities sharing similar biological features or graph features may have similar connections.

However, deploying methods based on biological features typically faces two problems: 1) Biological features may not always be available and can be hard and costly to obtain. One popular approach to solve this problem is to remove those biological entities without features via pre-processing, which usually results in small-scale pruned datasets and thus is not pragmatic and useful in the real setting (Zhang *et al.*, 2018c). 2) Biological features, as well as hand-crafted graph features (e.g., node degrees), could be not precise enough to represent or characterize biomedical entities, and may fail to help build a robust and accurate model for many applications (Hamilton *et al.*, 2017).

Graph embedding methods that seek to learn node representations automatically open opportunities to solve the two problems mentioned above. Embedding ideas also have been employed in some recently proposed computational methods in the biomedical field. For example, in DDAs prediction, matrix factorization-based techniques (Yang *et al.*, 2014; Zhang *et al.*, 2018d; Dai *et al.*, 2015) are utilized to factorize the drug-disease association matrix and learn low-dimensional representations for drug/disease in the latent space. During factorization, regularization terms or constraints can be added to further improve the quality of latent representations. In DDIs prediction, Zhang *et al.* (2018b) propose manifold regularized matrix factorization in which Laplacian regularization is incorporated to learn a better drug representation. Besides, graph-based autoencoder is introduced for DDIs prediction (Zitnik *et al.*, 2018; Ma *et al.*, 2018) whose intuition is similar to GAE (Kipf and Welling, 2016). For predicting PPIs, Laplacian and SVD are commonly adopted (Zhu *et al.*,

Table 1. A summary of 11 representative graph embedding methods and existing work (if any) using them for a certain task. ✗ means that a method (row) has not been applied for a task (column). As we can see, recent graph embedding methods on biomedical tasks are under-investigated.

Method Catergory		Embdding Methods	Link Prediction Tasks			Node Classification Task
			drug-disease association prediction	drug-drug interaction prediction	protein-protein interaction prediction	medical term type classification
Traditional	Matrix Factorization-based	Laplacian	(Zhang *et al.*, 2018d)	(Zhang *et al.*, 2018b)	(Zhu *et al.*, 2013)	✗
		SVD	(Dai *et al.*, 2015)	✗	(You *et al.*, 2017)	✗
		GF	(Yang *et al.*, 2014) (Zhang *et al.*, 2018d)	(Zhang *et al.*, 2018b)	✗	✗
Recently Proposed		HOPE	✗	✗	✗	✗
		GraRep	✗	✗	✗	✗
	Random Walk-based	DeepWalk	✗	✗	✗	✗
		node2vec	✗	✗	✗	✗
		struc2vec	✗	✗	✗	✗
	Neural Network-based	LINE	✗	✗	✗	✗
		SDNE	✗	✗	(Wang *et al.*, 2017b)	✗
		GAE	✗	(Zitnik *et al.*, 2018) (Ma *et al.*, 2018)	✗	✗

2013; You *et al.*, 2017). Additionally, autoencoder (Wang *et al.*, 2017b) is also applied, which has a similar design as SDNE (Wang *et al.*, 2016).

3.2 Node classification in the medical term graph

In addition to the link prediction task with the application of graph embedding, *node classification* which aims to predict the class of unlabeled nodes given a partially labeled graph, is also one of the most important applications of graph embedding in graph analysis and knowledge discovery (Tang *et al.*, 2015; Grover and Leskovec, 2016).

With the development of modern hospital information systems and the rapid growth of the adoption of Electronic Medical Records (EMRs), multiple sources of clinical information (including diagnostic history, medications, and laboratory test results) are becoming available for biomedical researchers, which provides a great opportunity for the analysis of large-scale clinical data. However, a large amount of clinical information remains under-tapped and locked in the unstructured data (e.g., clinical notes, surgical records, discharge records) as EMRs (Hersh *et al.*, 2013). Some recent works try to extract medical phrases and their relations from clinical texts to make the buried information more structured and accessible (Lv *et al.*, 2016). However, the phrase mining methods mainly focus on extracting words or phrases from clinical texts and do not reveal the semantic information (e.g., semantic type or categories) of extracted phrases (e.g., pharmacological substance, sign or symptom) and leave this task to later phases. Hence, we formulate a node classification task (see Fig. 2): *Classify the semantic types of medical terms extracted from clinical texts.* In this work, we assume the clinical texts have been converted into a medical term-term co-occurrence graph as in (Finlayson *et al.*, 2014), where each node is an extracted medical terms and each edge is the co-occurrence count of two terms in a context window. We apply graph embedding methods to the co-occurrence graph to learn representations of medical terms. Afterward, a multi-label classifier can be trained based on the learned embeddings to classify the semantic types of medical terms.

3.3 Summary

Table 1 summarizes 11 representative graph embedding techniques by three categories and the existing works by applying them for certain tasks. As can be seen, existing methods for the 4 representative biomedical tasks primarily adopt the traditional techniques, e.g., Laplacian Eigenmaps, matrix factorization. On the other hand, more recent advanced graph embedding methods have been demonstrated to outperform traditional techniques in social/information networks (Tang *et al.*, 2015; Cao *et al.*, 2015; Wang *et al.*, 2016), but whether they can perform well in biomedical networks are yet unknown. Hence, we conduct comprehensive

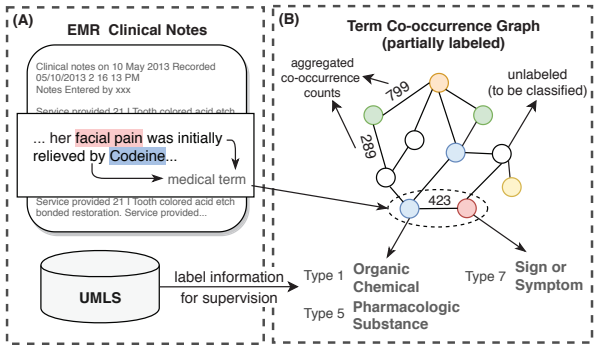


Fig. 2: Illustration of (A) how medical term-term co-occurrence graph is constructed and (B) node type classification in the graph. Our work assumes that the graph is given as in (Finlayson *et al.*, 2014) and mainly focuses on (B), i.e., testing various embedding methods on the classification performance.

experiments to evaluate those 11 graph embedding methods selected from three different categories on four representative biomedical tasks.

We follow the pipeline (shown in Fig. 1) of the widely adopted link prediction and node classification methods in general domains (Tang *et al.*, 2015; Grover and Leskovec, 2016): Graph embeddings are first learned and then used as feature inputs to build a binary classifier or multi-label classifier (e.g., Logistic Regression, SVM, MLP) to predict the unobserved links or the node labels.

4 Experiments

In this section, we introduce the details of 5 compiled datasets, including 2 DDA graphs, a DDI graph, a PPI graph, and a medical term-term co-occurrence graph, and use them as benchmark datasets to systematically evaluate the selected graph embedding methods.

4.1 Datasets

Drug-disease association (DDA) graph. We extract chemical-disease associations from the Comparative Toxicogenomics Database (CTD) (Davis *et al.*, 2018). CTD offers two kinds of associations: curated (verified) and inferred. Since our task is to infer potential chemical-disease associations, we only use curated ones as our golden instances. Finally, we obtain 92,813 edges between 12,765 nodes (9,580 chemicals and 3,185 diseases) in this graph (named as "CTD DDA").

Also, we construct another DDA network from National Drug File Reference Terminology (NDF-RT) in UMLS (Bodenreider, 2004). NDF-RT is produced by the U.S. Department of Veterans Affairs, and it models

drug characteristics including ingredients, physiologic effect, and related diseases. We extract drug-disease treatment associations using the *may treat* and *may be treated by* relationships in NDF-RT. This graph (named "NDF-RT DDA") contains 13,545 nodes (12,337 drugs and 1,208 diseases) and 56,515 edges.

Drug-drug interaction (DDI) graph. We collect verified DDIs from DrugBank (Wishart *et al.*, 2017), a comprehensive and freely accessible online database that contains detailed information about drugs and drug targets. We obtain 242,027 DDIs between 2,191 drugs and refer to this dataset as "DrugBank DDI".

Protein-protein interaction (PPI) graph. We extract *Homo sapiens* PPIs from STRING database (Szklarczyk *et al.*, 2014). Each PPI is associated with a confidence score that indicates its possibility to be a true positive interaction. To reduce noise, we only collect PPI whose confidence score is larger than 0.7. Finally, we obtain 359,776 interactions among 15,131 proteins and name this dataset as "STRING PPI".

Medical term-term co-occurrence graph. We adopt a publicly available set of medical terms with their co-occurrence statistics which are extracted by Finlayson *et al.* (2014) from 20 million clinical notes collected from Stanford Hospitals and Clinics (Lowe *et al.*, 2009) since 1995. Medical terms are extracted from raw clinical notes using an existing phrase mining tool (LePendu *et al.*, 2012) by matching with 22 clinically relevant ontologies such as SNOMED-CT and MedDRA. Co-occurrence frequencies between two terms are counted based on how many times they co-occur in the same temporal *bin* (i.e., a certain time-frame, see (Finlayson *et al.*, 2014) for more details). We select *perBin 1-day* dataset since it contains more medical terms compared to other bins. To filter very common medical terms (e.g., "medical history", "medication dose") that may influence the quality of embeddings, we convert the co-occurrence counts to the PPMI values (Levy and Goldberg, 2014) and remove the edges whose PPMI values are less than 2. We also adopt a subsampling (Mikolov *et al.*, 2013) strategy to further filter common terms and construct a medical term-term co-occurrence graph that contains 48,651 medical terms and 1,659,249 edges.

We keep the medical terms that can be mapped to the Unified Medical Language System (UMLS) Concept Unique Identifiers (CUI) and collect their corresponding semantic types (e.g., clinical drug, disease or syndrome) from UMLS. We select 31 different semantic types, with each having more than 20 samples. Finally, we obtain 25,120 nodes with label information. This dataset is called "Clin Term COOC".

The details of all datasets are summarized in Table 2.

Table 2. Statistics of the datasets, where the Density is defined as $\frac{2*\#Edges}{\#Nodes^2}$

Task Type	Dataset	#Nodes	#Edges	Density	#Node Labels
Link Prediction	CTD DDA	12,765	92,813	0.11%	-
	NDFRT DDA	13,545	56,515	0.06%	-
	DrugBank DDI	2,191	242,027	10.08%	-
	STRING PPI	15,131	359,776	0.31%	-
Node Classification	Clin Term COOC	48,651	1,659,249	0.14%	31

4.2 Experimental Set-up

We use OpenNE*, an open-source Python package for network embedding, to learn node embeddings for Laplacian Eigenmaps (Belkin and Niyogi, 2003), HOPE (Ou *et al.*, 2016), GF (Ahmed *et al.*, 2013), DeepWalk (Perozzi *et al.*, 2014), LINE (Tang *et al.*, 2015) and SDNE (Wang *et al.*, 2016). We run SVD using Numpy† and obtain struc2vec‡ (Ribeiro *et al.*, 2017) and GAE§ (Kipf and Welling, 2016) embeddings using the source code provided by their authors. More implementation details can be found in the Supplementary Materials.

For the link prediction tasks (Section 4.3), all the known interactions are positive samples and are split into the training set (80%) and testing set (20%). Since unknown interactions are far more than known ones, we randomly select disconnected edges as negative samples with an equal number of positive samples in both training and testing phase. After learning embeddings, for each node pair, we concatenate the embeddings of two nodes as edge features to build a simple Logistic Regression binary classifier using scikit-learn package (Pedregosa *et al.*, 2011). Area under ROC curve (*AUC*), *accuracy* and *F1* score are used to evaluate the performance of the classifiers, so as to evaluate different embedding methods.

For the node classification task (Section 4.4), we use the entire graph information to train the embeddings. Afterward, nodes with label information are split into the training set (80%) and the testing set (20%). The embedding vectors of nodes are directly treated as feature vectors and used to train *One-vs-Rest* Logistic Regression classifiers using the scikit-learn package. *Accuracy*, *Macro-F1* and *Micro-F1* are used to evaluate the performance of different embedding methods on the testing set.

For all embedding methods, the dimensionality of the learned embedding is set to 100 unless otherwise stated and we also discuss its impact on the performance. Moreover, we tune 1-2 significant hyper-parameters for some embedding methods via grid-search (see Section 4.5 for details). Other hyper-parameters for each method are set at their default values recommended by the corresponding papers.

4.3 Link Prediction Results

We conduct the link prediction task on the 4 compiled biomedical networks: CTD DDA, NDFRT DDA, DrugBank DDI, and STRING PPI. Table 3 shows the overall performance of different embedding methods on the four datasets.

Generally, compared to traditional techniques (e.g., Laplacian Eigenmaps, SVD, GF), the recently proposed embedding methods have largely improved the link prediction performance. For example, LINE achieves 3%-23% improvements in terms of the *AUC* value on the 4 datasets compared with Laplacian Eigenmaps. Struc2vec obtains 3%-15% gains of the *accuracy* on the 4 datasets respectively when compared with GF. The results demonstrate that the recently proposed graph embedding methods are more effective and could be used on various biological link prediction tasks to improve the prediction performance.

Furthermore, we have the following key observations and analyses:

- *For the matrix factorization-based methods*, since HOPE and GraRep are designed to capture the high-order proximity of graphs, they are usually more effective than traditional matrix factorization methods that only preserve the first-order of networks.
- *For the random walk-based methods*, generally, struc2vec performs better than DeepWalk and node2vec. This is not surprising because compared to DeepWalk and node2vec, struc2vec constructs a hierarchy weighted graph to measure the structural identity. Such hierarchy structure design incorporates both node degree distributions from the bottom as well as the entire network on the top, which can better capture the graph structure information and obtain better performansce.
- *For the neural network-based methods*, LINE achieves competitive prediction performance consistently, and only a little inferior compared to the best performing method on each dataset. It indicates that directly

* https://github.com/thunlp/OpenNE

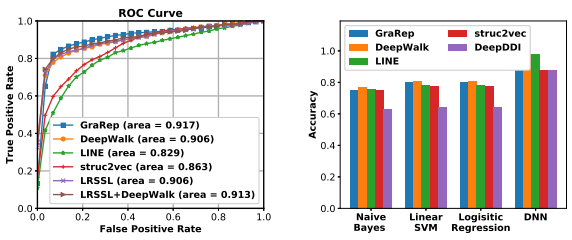
† http://www.numpy.org/

‡ https://github.com/leoribeiro/struc2vec

§ https://github.com/tkipf/gae

Table 3. Overall link prediction performance on the four compiled biomedical datasets. The best performing method in each category is in bold.

Method Catergory		Method Name	CTD DDA			NDFRT DDA			DrugBank DDI			STRING PPI		
			AUC	ACC	F1	AUC	ACC	F1	AUC	ACC	F1	AUC	ACC	F1
Traditional	Matrix factorization-based	Laplacian	0.8496	0.788	0.7972	0.9321	0.9191	0.923	0.7966	0.7183	0.727	0.6175	0.5824	0.5809
		SVD	0.934	0.8527	0.8513	0.7741	0.7014	0.6948	0.9191	0.8374	0.8373	0.8673	0.7938	0.7894
		GF	0.8824	0.8083	0.8055	0.7274	0.6642	0.6604	0.8832	0.8031	0.8101	0.8152	0.7456	0.7461
Recently Proposed		HOPE	0.9507	0.8845	0.8855	0.9498	0.9273	0.9304	0.9246	0.8443	0.8457	0.8388	0.7635	0.7632
		GraRep	**0.9596**	**0.8987**	**0.8994**	**0.9632**	**0.9321**	**0.9347**	**0.9254**	**0.845**	**0.8461**	**0.8958**	**0.8254**	**0.8252**
	Random walk-based	DeepWalk	0.9326	0.8677	0.866	0.7902	0.7208	0.7216	0.924	0.843	0.845	**0.8899**	**0.8178**	**0.8185**
		node2vec	0.9071	0.8332	0.8297	0.7451	0.6777	0.6776	0.9028	0.8209	0.8209	0.8002	0.7313	0.7328
		struc2vec	**0.9631**	**0.9002**	**0.9000**	**0.9568**	**0.9147**	**0.9137**	**0.9055**	**0.8246**	**0.8283**	0.8809	0.8090	0.8091
	Neural network-based	LINE	**0.9623**	**0.9028**	**0.9029**	**0.9604**	**0.934**	**0.9357**	0.9092	0.828	0.8319	0.8552	0.784	0.7918
		SDNE	0.9317	0.8645	0.8647	0.9466	0.9036	0.9052	0.9107	0.832	0.8372	**0.8944**	**0.8236**	**0.8236**
		GAE	0.9245	0.8387	0.8371	0.7337	0.6549	0.6464	**0.9185**	**0.8356**	**0.8389**	0.8535	0.7807	0.7864



(a) (b)

Fig. 3: (a) Comparison with the state-of-the-arts for drug-disease association prediction (LRSSL) and (b) drug-drug interaction prediction (DeepDDI). Graph embedding methods achieve competitive performance against them.

modeling edge information by a single-layer MLP is an effective way to learn node embeddings. SDNE and GAE also obtain satisfying prediction performance, which demonstrates that autoencoders and graph convolutional networks can also be useful for capturing graph structural information.

Comparison with state-of-the-art studies. To further demonstrate the effectiveness of graph embedding methods, we compare them with the state-of-the-art methods for two link prediction tasks: drug-disease association prediction and drug-drug interaction prediction.

For the DDAs prediction, we select LRSSL (Liang *et al.*, 2017) as our baseline. LRSSL is a Laplacian regularized sparse subspace learning framework which aims to project different drug features into a common subspace. Three drug feature profiles (i.e., chemical substructure, target domain and target annotation) are used in the training process. To fairly compare with LRSSL, we adopt the code and dataset used in their original paper. To learn graph embeddings without modeling biological features, we run four representative graph embedding methods: GraRep, DeepWalk, LINE, and struc2vec on LRSSL's drug-disease association graph. Following the same train/test split, training and evaluation process of link prediction in Section 4.2, we plot the ROC Curves to illustrate the performance of different methods better. As seen in Fig. 3(a), graph embedding methods achieve competitive performance compared with LRSSL. Further, we use learned DeepWalk embedding vectors as the 4th feature for the LRSSL method and improve the LRSSL performance.

For the DDIs prediction, we compare the embedding methods with a recent method DeepDDI (Ryu *et al.*, 2018). DeepDDI first adopts Principal Component Analysis (PCA) to reduce the dimension of the drug features (i.e., drug substructure) and then feeds features into a deep neural network (DNN) classifier. To fairly compare DeepDDI with graph embedding methods and reduce the bias caused by different classifiers, we compare methods under 4 classifiers, Naive Bayes, Linear SVM, Logistic Regression and 8-layer DNN (exactly the same one as in the original paper).

Table 4. Overall node classification performance on the "Clin Term COOC" dataset.

Category	Method	Accuracy	Micro-F1	Macro-F1
Matrix Factorization-based	Laplacian	0.2711	0.3071	0.0742
	SVD	0.3627	0.4242	0.1927
	GF	0.3025	0.3542	0.1308
	HOPE	0.3364	0.3906	0.1689
	GraRep	0.3563	0.4118	0.1705
Random Walk-based	DeepWalk	0.3830	0.4381	0.1898
	node2vec	0.4144	0.4704	0.2240
	struc2vec	0.2253	0.2577	0.0393
Neural Network-based	LINE	0.4013	0.4568	0.2141
	SDNE	0.2588	0.2995	0.0521

*The source code of GAE provided by the authors does not support a large-scale graph (nodes > 40k). We omit its performance here.

More implement details can be found in the Supplementary Materials. As seen in Fig. 3(b), graph embeddings outperform the drug features-based model or obtain very competitive performance under each classifier, which demonstrates the power of graph embedding methods.

4.4 Node Classification Results

Apart from biological link prediction tasks, node classification task is also another critical task in biomedical graph analysis. Here, we focus on classifying the semantic types of medical terms given their co-occurrence graph extracted from clinical notes. Table 4 shows the performance of different embedding methods, and we make the following key observations:

• *For the matrix factorization-based methods*, it is a little surprising that the traditional method SVD achieves better performance, even surpassing HOPE and GraRep. The reason may be that the high-order proximity in word/phrase co-occurrence networks sometimes is not so essential. Directly modeling the first-order proximity (i.e., co-occurrence) would be good enough to classify the nodes.

• *For the random walk-based methods*, node2vec performs better since it aims to capture different functions of nodes (i.e., homophily and structural equivalence) via a more flexible biased random walk. Struc2vec performs worse on this term co-occurrence graph as it mainly focuses on modeling the structural identity of nodes; however, a clear structural role may not exist in the medical term co-occurrence graph, which leads to worse performance.

• *For the neural network-based methods*, LINE achieves competitive performance, which demonstrates that directly modeling edge information is an effective way to learn the embedding for the node classification task. On the other hand, the deep autoencoder-based method SDNE performs worse on this graph. The reason may be that when the scale of the input data (i.e., adjacency vector) is large, the reconstruction loss of the autoencoder is too large to be optimized, and thus it is hard to learn good embeddings.

4.5 Influence of Hyper-parameters

The hyper-parameters can have a significant impact on machine learning models. In this section, we investigate the influence of some important hyper-parameters in various embedding methods. To be specific, we first evaluate how different embedding dimensions can affect the prediction performance. Fig. 4 shows the impact of embedding dimensionality on the prediction performance for "CTD DDA" and "Clin Term COOC" datasets (results on other datasets are in the Supplementary Materials). Generally, the prediction performance becomes better when the embedding dimensionality increases, which is intuitive since higher dimensionality can encode more useful information. However, it is also expected that the time cost for training the classifier increases as well.

Further, we select 1-2 sensitive hyper-parameters from 6 embedding methods, which have been pointed out to be important ones by their authors of embedding methods. Table S1 (in the Supplementary Materials) shows the selected hyper-parameters in different embedding methods as well as their meanings. We tune these hyper-parameters by grid search. We provide some high-level guidelines on setting hyper-parameters for practitioners (results and guidelines are both discussed in the Supplementary Materials).

4.6 Summary of Experimental Results

In summary, we can see that, in general, the recently proposed graph embedding methods outperform traditional methods in various biomedical tasks and thus more attention is expected to be paid on these more advanced embedding methods for future biomedical graph analysis.

For matrix factorization-based methods, we observe that modeling high-order proximity (e.g., HOPE, GraRep) is generally useful for link prediction tasks but may be less meaningful for the node classification task. For random walk-based methods, struc2vec is more suitable for link prediction tasks while node2vec performs better in the node classification task. Also, DeepWalk is robust for various datasets and tasks. For neural network-based methods, LINE usually achieves competitive performance against the best performing method on each dataset. SDNE and GAE can achieve good performance on relatively smaller datasets but may not perform well on large-scale datasets.

More details of the datasets, implementation, experiment results, guidelines can be found in the Supplmentary Materials.

5 Future Directions

Modeling external information in graph embedding learning. In addition to the graph structure, external information can also help build computational models for biomedical networks. Among the most commonly used ones are the biological features of entities (e.g., drug substructures). For example, (Zhang *et al.*, 2018d) incorporate drug and disease features into matrix factorization to learn better representations. There may also exist partial label information on graphs (e.g., semantic types are partly available for nodes in a medical term co-occurrence graph). Incorporating those features and labels into advanced graph embedding models can potentially further improve the performance. There have been a surge of *attributed graph embedding* methods that explore this direction. For example, DDRW (Li *et al.*, 2016) and MMDW (Tu *et al.*, 2016) jointly optimize the objective of DeepWalk with a Support Vector Machine (SVM) classification loss to incorporate label information. We leave benchmarking such *attributed network embedding* methods on biomedical graphs as our future work.

Transfer learning for graph embedding. Recent studies in Computer Vision and Natural Language Processing show that transfer learning helps improve model performance on different tasks (Shin *et al.*, 2016; Howard and Ruder, 2018). General patterns are captured during pre-trained processes and can be “transferred” into new prediction tasks. There also exist some pre-trained embeddings of biomedical entities (Choi *et al.*, 2016; Beam *et al.*, 2018) which allow us to adopt similar ideas of "transfer learning" to learn graph embeddings. We can initialize the embedding vector for each node on a graph with its pre-trained embedding (e.g., by looking for the corresponding entity in (Choi *et al.*, 2016; Beam *et al.*, 2018)) rather than by random initialization, and then continue training various graph embedding methods as before (which is often referred to as “fine-tuning”). The pre-trained embeddings can be seen as "coarse embeddings" since they are usually pre-trained on a large general corpus and have not been optimized for downstream tasks yet. Nevertheless, they can contain some additional semantic information that may not be able to be learned from a downstream task graph (e.g., due to its small scale). By fine-tuning, such additional semantic information can be "transferred" into the finally learned embeddings. We experiment with this transfer learning idea on the "CTD DDA" graph. As seen from Table S3 in the Supplementary Materials, the link prediction performance has been improved using the pre-trained embeddings from (Beam *et al.*, 2018). Currently, the number of released biomedical entities with pre-trained embeddings is still limited and entities without pre-trained embeddings have to be initialized randomly. However, with the increasing volume of biomedical data, more and more entities can have pre-trained embeddings, and the idea of *pre-training -then- fine-tuning* can be more promising.

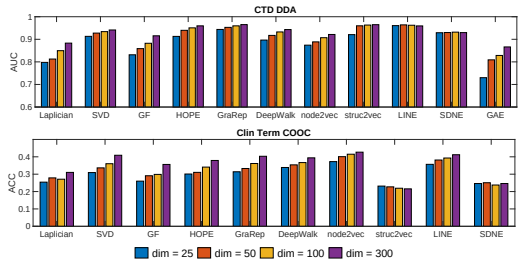


Fig. 4: The influence of dimensionality of learned embeddings from different embedding methods.

6 Conclusion

This paper provides an overview of various graph embedding techniques and evaluates their performance on four biomedical network analysis tasks (i.e., DDAs prediction, DDIs prediction, PPIs prediction, and medical term semantic type classification). We compile 5 datasets for these 4 tasks and use them to benchmark 11 representative graph embedding methods. Through extensive experiments, we demonstrate that the more recent and advanced graph embedding methods (e.g., node2vec, LINE, struc2vec) usually outperform the traditional methods (e.g., matrix factorization) and deserve further investigations for future biomedical graph analysis. Besides, we provide some general guidelines for practitioners to properly select embedding methods and their hyper-parameters and also discuss potential directions (e.g., transfer learning for graph embedding) as the future work.

References

Ahmed, A., Shervashidze, N., Narayanamurthy, S., Josifovski, V., and Smola, A. J. (2013). Distributed large-scale natural graph factorization. In *WWW*, pages 37–48. ACM.

Beam, A. L., Kompa, B., Fried, I., Palmer, N. P., Shi, X., Cai, T., and Kohane, I. S. (2018). Clinical concept embeddings learned from massive sources of medical data. *arXiv preprint arXiv:1804.01486*.

Belkin, M. and Niyogi, P. (2002). Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*, pages 585–591.

Belkin, M. and Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, **15**(6), 1373–1396.

Bodenreider, O. (2004). The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, **32**(suppl_1), D267–D270.

Cao, S., Lu, W., and Xu, Q. (2015). Grarep: Learning graph representations with global structural information. In *CIKM*, pages 891–900. ACM.

Cao, S., Lu, W., and Xu, Q. (2016). Deep neural networks for learning graph representations. In *AAAI*, pages 1145–1152.

Choi, Y., Chiu, C. Y.-I., and Sontag, D. (2016). Learning low-dimensional representations of medical concepts. *AMIA*, **2016**, 41.

Dai, W., Liu, X., Gao, Y., Chen, L., Song, J., Chen, D., Gao, K., Jiang, Y., Yang, Y., Chen, J., *et al.* (2015). Matrix factorization-based prediction of novel drug indications by integrating genomic space. *Computational and mathematical methods in medicine*, **2015**.

Davis, A. P., Grondin, C. J., Johnson, R. J., Sciaky, D., McMorran, R., Wiegers, J., Wiegers, T. C., and Mattingly, C. J. (2018). The comparative toxicogenomics database: update 2019. *Nucleic Acids Research*, page gky868.

Ezzat, A., Wu, M., Li, X.-L., and Kwoh, C.-K. (2017). Drug-target interaction prediction using ensemble learning and dimensionality reduction. *Methods*, **129**, 81–88.

Finlayson, S. G., LePendu, P., and Shah, N. H. (2014). Building the graph of medicine from millions of clinical narratives. *Scientific data*, **1**, 140032.

Gottlieb, A., Stein, G. Y., Ruppin, E., and Sharan, R. (2011). Predict: a method for inferring novel drug indications with application to personalized medicine. *Molecular systems biology*, **7**(1), 496.

Goyal, P. and Ferrara, E. (2018). Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems*, **151**, 78–94.

Grover, A. and Leskovec, J. (2016). node2vec: Scalable feature learning for networks. In *KDD*, pages 855–864. ACM.

Hamilton, W. L., Ying, R., and Leskovec, J. (2017). Representation learning on graphs: Methods and applications. *arXiv preprint arXiv:1709.05584*.

Hersh, W. R., Weiner, M. G., Embi, P. J., Logan, J. R., Payne, P. R., Bernstam, E. V., Lehmann, H. P., Hripcsak, G., Hartzog, T. H., Cimino, J. J., *et al.* (2013). Caveats for the use of operational electronic health record data in comparative effectiveness research. *Medical care*, **51**(8 0 3), S30.

Howard, J. and Ruder, S. (2018). Universal language model fine-tuning for text classification. In *ACL*, volume 1, pages 328–339.

Kipf, T. N. and Welling, M. (2016). Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*.

Kipf, T. N. and Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In *ICLR*.

LePendu, P., Iyer, S. V., Fairon, C., and Shah, N. H. (2012). Annotation analysis for testing drug safety signals using unstructured clinical notes. *Journal of biomedical semantics*, **3**(1), S5.

Levy, O. and Goldberg, Y. (2014). Linguistic regularities in sparse and explicit word representations. In *CoNLL*, pages 171–180.

Li, J., Zhu, J., and Zhang, B. (2016). Discriminative deep random walk for network classification. In *ACL*, volume 1, pages 1004–1013.

Liang, X., Zhang, P., Yan, L., Fu, Y., Peng, F., Qu, L., Shao, M., Chen, Y., and Chen, Z. (2017). Lrssl: predict and interpret drug–disease associations based on data integration using sparse subspace learning. *Bioinformatics*, **33**(8), 1187–1196.

Lowe, H. J., Ferris, T. A., Hernandez, P. M., and Weber, S. C. (2009). Stride–an integrated standards-based translational research informatics platform. In *AMIA Annual Symposium Proceedings*, volume 2009, page 391. AMIA.

Lv, X., Guan, Y., Yang, J., and Wu, J. (2016). Clinical relation extraction with deep learning. *IJHIT*, **9**(7), 237–248.

Ma, T., Xiao, C., Zhou, J., and Wang, F. (2018). Drug similarity integration through attentive multi-view graph auto-encoders. *arXiv preprint arXiv:1804.10850*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *NIPS*.

Ou, M., Cui, P., Pei, J., Zhang, Z., and Zhu, W. (2016). Asymmetric transitivity preserving graph embedding. In *KDD*, pages 1105–1114. ACM.

Pedregosa, F., Varoquaux, G., Gramfort, A., and et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830.

Perozzi, B., Al-Rfou, R., and Skiena, S. (2014). Deepwalk: Online learning of social representations. In *KDD*, pages 701–710. ACM.

Ribeiro, L. F., Saverese, P. H., and Figueiredo, D. R. (2017). struc2vec: Learning node representations from structural identity. In *KDD*, pages 385–394. ACM.

Rotmensch, M., Halpern, Y., Tlimat, A., Horng, S., and Sontag, D. (2017). Learning a health knowledge graph from electronic medical records. *Scientific reports*, **7**(1), 5994.

Roweis, S. T. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *science*, **290**(5500), 2323–2326.

Ryu, J. Y., Kim, H. U., and Lee, S. Y. (2018). Deep learning improves prediction of drug–drug and drug–food interactions. *PNAS*, **115**(18), E4304–E4311.

Shin, H.-C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., and Summers, R. M. (2016). Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, **35**(5), 1285–1298.

Su, C., Tong, J., Zhu, Y., Cui, P., and Wang, F. (2018). Network embedding in biomedical data science. *Briefings in Bioinformatics*.

Szklarczyk, D., Franceschini, A., Wyder, S., Forslund, K., Heller, D., Huerta-Cepas, J., Simonovic, M., Roth, A., Santos, A., Tsafou, K. P., *et al.* (2014). String v10: protein–protein interaction networks, integrated over the tree of life. *Nucleic acids research*, **43**(D1), D447–D452.

Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., and Mei, Q. (2015). Line: Large-scale information network embedding. In *WWW*, pages 1067–1077. ACM.

Tenenbaum, J. B., De Silva, V., and Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *science*, **290**(5500), 2319–2323.

Tu, C., Zhang, W., Liu, Z., Sun, M., *et al.* (2016). Max-margin deepwalk: Discriminative learning of network representation. In *IJCAI*, pages 3889–3895.

Wang, D., Cui, P., and Zhu, W. (2016). Structural deep network embedding. In *KDD*, pages 1225–1234. ACM.

Wang, D. D., Wang, R., and Yan, H. (2014). Fast prediction of protein–protein interaction sites based on extreme learning machines. *Neurocomputing*, **128**, 258–266.

Wang, H., Wang, J., Wang, J., Zhao, M., Zhang, W., Zhang, F., Xie, X., and Guo, M. (2017a). Graphgan: Graph representation learning with generative adversarial nets. *arXiv preprint arXiv:1711.08267*.

Wang, H., Zhang, F., Hou, M., Xie, X., Guo, M., and Liu, Q. (2018). Shine: Signed heterogeneous information network embedding for sentiment link prediction. In *WSDM*, pages 592–600. ACM.

Wang, Y.-B., You, Z.-H., Li, X., Jiang, T.-H., Chen, X., Zhou, X., and Wang, L. (2017b). Predicting protein–protein interactions from protein sequences by a stacked sparse autoencoder deep neural network. *Molecular BioSystems*, **13**(7), 1336–1344.

Wishart, D. S., Feunang, Y. D., Guo, A. C., Lo, E. J., Marcu, A., Grant, J. R., Sajed, T., Johnson, D., Li, C., Sayeeda, Z., *et al.* (2017). Drugbank 5.0: a major update to the drugbank database for 2018. *Nucleic acids research*, **46**(D1), D1074–D1082.

Xie, J., Girshick, R., and Farhadi, A. (2016). Unsupervised deep embedding for clustering analysis. In *ICML*, pages 478–487.

Yang, J., Li, Z., Fan, X., and Cheng, Y. (2014). Drug–disease association and drug-repositioning predictions in complex diseases using causal inference–probabilistic matrix factorization. *JCIM*, **54**(9), 2562–2569.

You, Z.-H., Li, X., and Chan, K. C. (2017). An improved sequence-based prediction protocol for protein-protein interactions using amino acids substitution matrix and rotation forest ensemble classifiers. *Neurocomputing*, **228**, 277–282.

Zhang, D., Yin, J., Zhu, X., and Zhang, C. (2018a). Network representation learning: A survey. *IEEE transactions on Big Data*.

Zhang, P., Wang, F., Hu, J., and Sorrentino, R. (2015). Label propagation prediction of drug-drug interactions based on clinical side effects. *Scientific reports*, **5**, 12339.

Zhang, W., Yue, X., Chen, Y., Lin, W., Li, B., Liu, F., and Li, X. (2017a). Predicting drug-disease associations based on the known association bipartite network. In *BIBM*, pages 503–509. IEEE.

Zhang, W., Yue, X., Liu, F., Chen, Y., Tu, S., and Zhang, X. (2017b). A unified frame of predicting side effects of drugs by using linear neighborhood similarity. *BMC systems biology*, **11**(6), 101.

Zhang, W., Chen, Y., Li, D., and Yue, X. (2018b). Manifold regularized matrix factorization for drug-drug interaction prediction. *JBI*, **88**, 90–97.

Zhang, W., Yue, X., Huang, F., Liu, R., Chen, Y., and Ruan, C. (2018c). Predicting drug-disease associations and their therapeutic function based on the drug-disease association bipartite network. *Methods*, **145**, 51–59.

Zhang, W., Yue, X., Lin, W., Wu, W., Liu, R., Huang, F., and Liu, F. (2018d). Predicting drug-disease associations by using similarity constrained matrix factorization. *BMC bioinformatics*, **19**(1), 233.

Zhang, W., Huang, F., Yue, X., Lu, X., Yang, W., Li, Z., and Liu, F. (2018e). Prediction of drug-disease associations and their effects by signed network-based nonnegative matrix factorization. In *BIBM*, pages 798–802. IEEE.

Zhang, W., Yue, X., Tang, G., Wu, W., Huang, F., and Zhang, X. (2018f). Sfpel-lpi: Sequence-based feature projection ensemble learning for predicting lncrna-protein interactions. *PLoS computational biology*, **14**(12), e1006616.

Zhu, L., You, Z.-H., and Huang, D.-S. (2013). Increasing the reliability of protein–protein interaction networks via non-convex semantic embedding. *Neurocomputing*, **121**, 99–107.

Zitnik, M., Agrawal, M., and Leskovec, J. (2018). Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics*, **34**(13), i457–i466.