

Drug-Drug Interaction Prediction Based on Knowledge Graph Embeddings and Convolutional-LSTM Network

Md. Rezaul Karim

Fraunhofer FIT, Aachen, Germany
RWTH Aachen University, Germany

Michael Cochez

Fraunhofer FIT, Aachen, Germany
RWTH Aachen University, Germany

Joao Bosco Jares

Fraunhofer FIT, Aachen, Germany
RWTH Aachen University, Germany

Mamtaz Uddin

Clinical Pharmacy and Pharmacology
University of Dhaka, Bangladesh

Oya Beyan

RWTH Aachen University, Germany
Fraunhofer FIT, Aachen, Germany

Stefan Decker

Fraunhofer FIT, Aachen, Germany
RWTH Aachen University, Germany

ABSTRACT

Interference between pharmacological substances can cause serious medical injuries. Correctly predicting so-called drug-drug interactions (DDI) does not only reduce these cases but can also result in a reduction of drug development cost. Presently, most drug-related knowledge is the result of clinical evaluations and post marketing surveillance; resulting in a limited amount of information. Existing data-driven prediction approaches for DDIs typically rely on a single source of information, while using information from multiple sources would help improve predictions. Machine learning (ML) techniques are used, but the techniques are often unable to deal with skew in the data. Hence, we propose a new ML approach for predicting DDIs based on multiple data sources. For this task we use 12,000 drug features from DrugBank, PharmGKB, and KEGG drugs, which are integrated using Knowledge Graphs (KGs). To train our prediction model, we first embed the nodes in the graph using various embedding approaches. We found that the best performing combination was a ComplEx embedding method creating using PyTorch-BigGraph (PBG) with a Convolutional-LSTM network and classic machine learning based prediction models. The model averaging ensemble method of three best classifiers yields up to 0.94, 0.92, 0.80 for AUPR, F1-score, and MCC, respectively during 5-fold cross-validation tests.

KEYWORDS

Drug-drug interactions; Linked data; Knowledge graphs; Graph embeddings; Conv-LSTM network; Model averaging ensemble.

ACM Reference Format:

Md. Rezaul Karim, Michael Cochez, Joao Bosco Jares, Mamtaz Uddin, Oya Beyan, and Stefan Decker. 2019. Drug-Drug Interaction Prediction Based on Knowledge Graph Embeddings and Convolutional-LSTM Network. In *10th ACM Int'l Conference on Bioinformatics, Computational Biology and Health Informatics (ACM-BCB '19)*, September 7–10, 2019, Niagara Falls, NY, USA. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/3307339.3342161>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Others than the author(s) must be honored. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ACM-BCB '19, September 7–10, 2019, Niagara Falls, NY, USA
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6666-3/19/09...\$15.00
<https://doi.org/10.1145/3307339.3342161>

1 INTRODUCTION

Drug-drug interactions (DDIs) are (often) preventable causes of medical injuries that occur when a drug causes a pharmacokinetic (PK) or pharmacodynamic (PD) effect on the body when it is taken together with another drug [41, 46]. They are a common cause of adverse drug reactions (ADRs) and increased healthcare costs [9]. The majority of ADRs are caused by unintended DDIs and occasionally arise through co-prescription of drugs. While it would be ideal for identifying all possible DDIs during clinical trials, interactions are frequently reported after the drugs are approved for clinical use. ADRs are a significant threat to public health, as shown in a study by Shtar et al. They found that about 6.7% of hospital readmission occurred because of ADRs with a fatality rate of 0.32% in the USA in 2014. In that year, as many as 807,270 cases of serious ADRs were reported in the United States, resulting in 123,927 lost lives [38].

For example, acetylsalicylic acid, commonly known as aspirin, is a drug used for the treatment of pain and fever due to various causes. This medicine has both anti-inflammatory and antipyretic effects, which inhibits platelet aggregation and is used in the prevention of blood clots and myocardial infarction. However, the risk or severity of hypertension can be increased (e.g., negative drug-drug interaction) when acetylsalicylic acid is combined with 1-benzylimidazole [48]. Predicting potential DDIs reduces unanticipated drug interactions, lowers drug development costs, and can be used to optimize the drug design process. Thus, the study of DDIs and ADRs is important in both drug development and clinical application, especially for co-administered medications. Since the majority of ADRs occur between pairs of drugs, they have become the focus of research and clinical studies [46]. To further reduce costs and to make the analysis of large amounts of interactions possible, automated methods for identifying ADRs are needed. Current approaches involve clinical evaluation of drugs and post-marketing surveillance. Here, features are extracted from drug properties such as targets, side-effects, chemical properties, fingerprint, and drug indications. Then statistical methods and various supervised ML algorithms (like, e.g., decision tree (DT), Naive Bayes (NB), k-nearest neighbors (k-NN), logistic regression (LR), support vector machine (SVM), random forest (RF), and gradient boosting trees (GBT)) [1] are employed.

Deep learning-based approaches, which can utilize deep features, are mostly unexplored in the context of DDIs prediction. While a deep architecture like a convolutional neural network (CNN) is good at reducing frequency variations by acting as a feature

extractor [53], a long short-term memory (LSTM) network is good at temporal modeling and learning orderly sequences from a large feature space [37]. By combining these two deep architectures, the convolutional-LSTM(Conv-LSTM) can capture both locally and globally important drug features which we found to lead to more accurate DDI predictions [37]. However, the features which have been traditionally used for these approaches form either a large and sparse binary matrix or a dense, but small similarity matrix, making them not ideal for training ML models [8]. Further, an increasing amount of drug and small molecules data is being generated, and state-of-the-art approaches still rely on the analysis on a limited number of data sources only, e.g. DrugBank.

To incorporate multiple data sources, Knowledge Graphs are a powerful tool, and many biomedical knowledge bases have been published in this form. In this graph, the nodes represent different entities like drugs, diseases, protein targets, substructures, side effects, and pathways. See e.g., [8, 46] for examples on using Knowledge Graphs for DDI prediction. Once the data is in the form of a Knowledge Graph, we have to extract information from it as features for our interaction predictors. To do this, we use embedding methods which project each node in the graph to a dense vector.

In our work, we consider more sources of DDIs as others. Scientific literature that has predicted DDIs very accurately is often ignored as ground truth in related work. In this paper, we collected DDI information from DrugBank [48], the Kyoto Encyclopedia of Genes & Genomes (KEGG) [23], TWOSIDES [43], and scientific literature. Then, we created an integrated KG using data from DrugBank, KEGG drug, PharmGKB [47], and OFFSIDES [43] (excluding data already in the above mentioned DDI data). To transform the information from this graph in a format suitable for the prediction models, we applied different KG embedding techniques. Then, we trained several ML models as baselines and also performed experiments with the Conv-LSTM model. The key contributions of this paper can be summarized as follows:

- We have created a dataset with 2,898,937 drug-drug interaction pairs; we believe that this is the largest available.
- We have prepared a large-scale integrated KG about DDIs with data from DrugBank, KEGG, OFFSIDES, and PharmGKB having 1.2 billion triples.
- We have evaluated different KG embeddings techniques with different settings to train and evaluate ML models.
- We provide a comprehensive evaluation with details analysis of the outcome and comparison with the state-of-the-art approaches and baseline models.
- We found that a combined CNN and LSTM network called Conv-LSTM for predicting DDIs leads to the highest accuracy.

This paper is structured as follows: section 2 discusses related works with their emerging use cases and potential limitations. Section 3 details the proposed approach, including problem formulation, data collections, KG construction, graph embeddings, network constructions, and training. The results of our experiments can be found in Section 4, where we also discuss the key findings of the study. Section 5 provides some explanations of the importance, highlights the limitations of the study reported, and discusses some future works before concluding the paper. To avoid confusion, we use the terms node and drug interchangeably throughout the paper.

2 RELATED WORK

Till date, DDI prediction is a non-trivial research problem in pharmacology, and numerous approaches have been proposed to predict novel DDIs by employing various data sources. Traditional work relies on in vitro and in vivo experiments and focuses on small sets of specific drug pairs and had laboratory limitations [15]. With the emergence of available biomedical data, researchers moved the focus towards automatically populating and completing biomedical KGs using large-scale structured databases and text publicly available [8]. In this scope, the Bio2RDF project made 35 life sciences datasets as linked open data (LOD) in RDF, in which similar entities are mapped in different KGs and built large heterogeneous graphs that also contain biomedical drug-related facts. Although these approaches made available numerous biomedical KGs, they often contain incomplete and inaccurate data that impede their application in the field of safe medicine development [8]. Lately, ML and text mining based approaches were used in which pharmacological similarities of drugs as features are used by regarding the DDIs prediction task as a link prediction problem.

Different drug similarity metrics are used for inferring DDIs and their associated recommendations in which LR is trained to calculate the maximum likelihood by using known DDIs [17]. Similarly, DDIs using phenotypic, chemical, biological, therapeutic, structural, and genomic similarities of drugs are used for predicting DDIs [9]. Other investigations used pharmacological and graph qualities between drugs [7] or drug structural similarities and interaction networks incorporating PK and PD knowledge [41] using LR. Peng et al. [28] developed a Bayesian network, which combines molecular drug similarity and drug side-effect similarity to predict the combined effect of drugs. Lately, Andrej K. et al. [24] formulated the DDIs prediction problem as a binary classification problem to predict unknown DDIs in 5 arbitrary databases such as DrugBank, KEGG, NDF-RT, SemMedDB, and Twosides. In these works, supervised ML approaches such as DT, NB, k-NN, LR, SVM, RF, and GBT are mostly used for predicting DDIs from topological and semantic similarity features. However, feature-based approaches only predict binary DDIs or those that have been pre-defined in structured databases and suffer from robustness caused by data sparsity and vast computation requirements. Similarity-based approaches, in contrast, do not allow for the calculation of various similarities for many drugs due to lack of drug information [8].

Several other approaches are proposed using biomedical KGs and text embedding in which graph embedding is utilized [46] to overcome issues such as data incompleteness and the sparsity problem. These learned embeddings are then mostly used for predicting DDIs. Other works have utilized text mining [15, 31, 42] to predict and evaluate new DDIs in which either drug-related data from scientific literature was discovered from large health information exchange repository [15] or automated reasoning has been developed to derive new enzyme-based DDIs from MEDLINE abstracts [42]. With the abundance of biomedical data characterizing drugs and their associated targets, these methods cannot fuse multiple sources of information and perform inference over the network of drugs effectively. Therefore, approaches employing KGs embeddings and ML-approaches [1, 8, 19, 46] have emerged. In particular, approaches based on KG embeddings are powerful predictors and

outperform state-of-the-art approaches for inferring new DDIs. Most of the embedding methods are translation-based; embeddings are built by treating relations as translations from the head entity to tail entity. In these methods, the vector embeddings are created such that $\mathbf{h} \oplus \mathbf{r} \approx \mathbf{t}$ where (h, r, t) is a triple of knowledge base (i.e., the relation r holds between h and t). For this equation, there are various options for the \oplus operator (see e.g., [6, 16, 22]).

With these ideas in mind, the DDIs prediction framework called Tiresias [1] is proposed in which various sources of drug-related data and knowledge are used as inputs and provides DDIs predictions as outputs using an LR classifier. The process starts with semantic integration of data into a KG describing drug attributes and relationships with various related entities such as enzymes, chemical structures, and pathways. The KG is then used to compute several similarity measures between the drugs in a scalable and distributed framework. A recent approach called PRD [46] was proposed for predicting DDIs in which graph embedding techniques such as TransE, TransD, TransH, HolE are employed to overcome the data incompleteness and sparsity issues [1]. First, a large-scale drug KG is created from different sources containing biomedical texts, which are then embedded into a common low dimensional space into a continuous vector space in which both entities and relations were considered. The learned embeddings are subsequently used to predict the DDIs using a rich DDI triple encoder (RDTE) network in which an encoder incorporates the drug-related information to obtain the DDI relation representation. The decoder then reconstructs the embedding vector from the latent representation and is used to predict the labels for potential DDIs.

However, most translation-based embeddings are limited in their capacity to model complex and diverse objects, including important properties of relations, such as symmetric, transitive, one-many, many-to-one, and many-many relations in KGs [16]. As KG embeddings techniques show state-of-the-art performance, generating quality feature vectors using appropriate embedding methods plays a significant role. A more recent work [8] employed KG embedding methods to extract feature vector representation of drugs using LOD to predict potential DDIs. The effects of DDIs prediction accuracy using LR, NB, and RF is also investigated on a single source (the Bio2RDF DrugBank v4 dataset) with different embedding methods such as RDF2Vec, TransE, and TransD.

In these approaches, DDIs information extraction from biomedical texts and drug event reports using text mining (TM) and then inferring DDIs by integrating knowledge from several sources are two typical steps [24, 32, 43]. Although, the way DDIs are extracted and the predictions made vary across methods. Numerous approaches have extracted DDIs from biomedical text using knowledge-rich and knowledge-poor features [18] or from lexical, syntactical, and semantic-based features [5]. Other approaches, focus on classifying DDIs in which an SVM is trained using drug features generated by similarity measures [4, 33] or by exploiting linguistic information [10]. Apart from these, other approaches have employed text mining for extracting DDIs from a semantically annotated corpus of documents describing DDIs from DrugBank and MEDLINE abstracts [21, 36].

3 MATERIALS AND METHODS

In this section, we discuss our methods in detail, including the problem formulation, data collection and integration, KG embeddings, the Conv-LSTM network construction, and the network training with hyperparameter optimization. The last step is inferencing DDI predictions. Figure 1 shows the workflow of the proposed approach.

3.1 Problem formulation

Since DDIs form a complex network in which nodes refer to drugs and links refer to potential interactions, we approach the DDIs prediction task as a link prediction problem similar to Shtar et al. [38]. Given a directed DDI KG as $G = (V, E)$ in which each edge $e = (u, v) \in E$ represents an interaction between drugs u and v . Let N denotes the number of drugs, we can define the DDIs matrix $Y \in \{0, 1\}^{N \times N}$ as follows:

$$y_{u,v} = \begin{cases} 1, & \text{if interaction exists between drugs } u \text{ and } v \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

In eq. (1), a value of 1 for $y_{u,v}$ indicates an existing interaction between drugs u and v . However, a value of 0 does not mean that an interaction does not exist in the KG, but it could be that the interaction has not yet been discovered [38]. Next, we proceed to DDIs extraction to be followed by KG construction.

3.2 DDIs extraction and KG construction

Since creating an integrated KG and extracting DDIs are the two most important steps in our approach, we first focus on the data and knowledge source selection. We constructed our integrated knowledge graph based on the drugs and drug-target related data from DrugBank, KEGG drug, and PharmGKB. On the other hand, OFFSIDES, TWOSIDES, and scientific literature from MEDLINE are used for finding DDI with enough evidence.

3.2.1 Data collection. At the current time, there is no automated method or data source available which would provide complete DDI information. Moreover, the available data is spread over multiple sources. Therefore, we rely on several sources for collecting drug-related data. The *DrugBank* database is a bioinformatics and cheminformatics resource that combines detailed drug-related information, including chemical, pharmacological, and pharmaceutical data with comprehensive drug target information.

The *PharmGKB* database¹ contains 12,664 drug entries including 2,588 approved small molecule drugs, 1,287 approved biotech drugs, 130 nutraceuticals and over 6,305 experimental drugs [48]. The *PharmGKB* database is a repository for genomic, molecular, and cellular phenotype data. It also contains clinical information and the impact of genetic variation on drug response about people who have participated in pharmacogenomics research studies. PharmGKB contains genes, diseases, drugs, and pathways related data as well as detailed information on 470 genetic variants affecting drug metabolism.

The *KEGG* databases contain metabolic pathways that are hyperlinked to metabolite and protein/enzyme information. As of May 2019, KEGG drug database has 10,979 drugs related information and 501,689 DDIs relations. Finally, the *OFFSIDES* database, which contains drug effects mined from adverse event reports based on

¹DrugBank v5.1.3, April 02, 2019

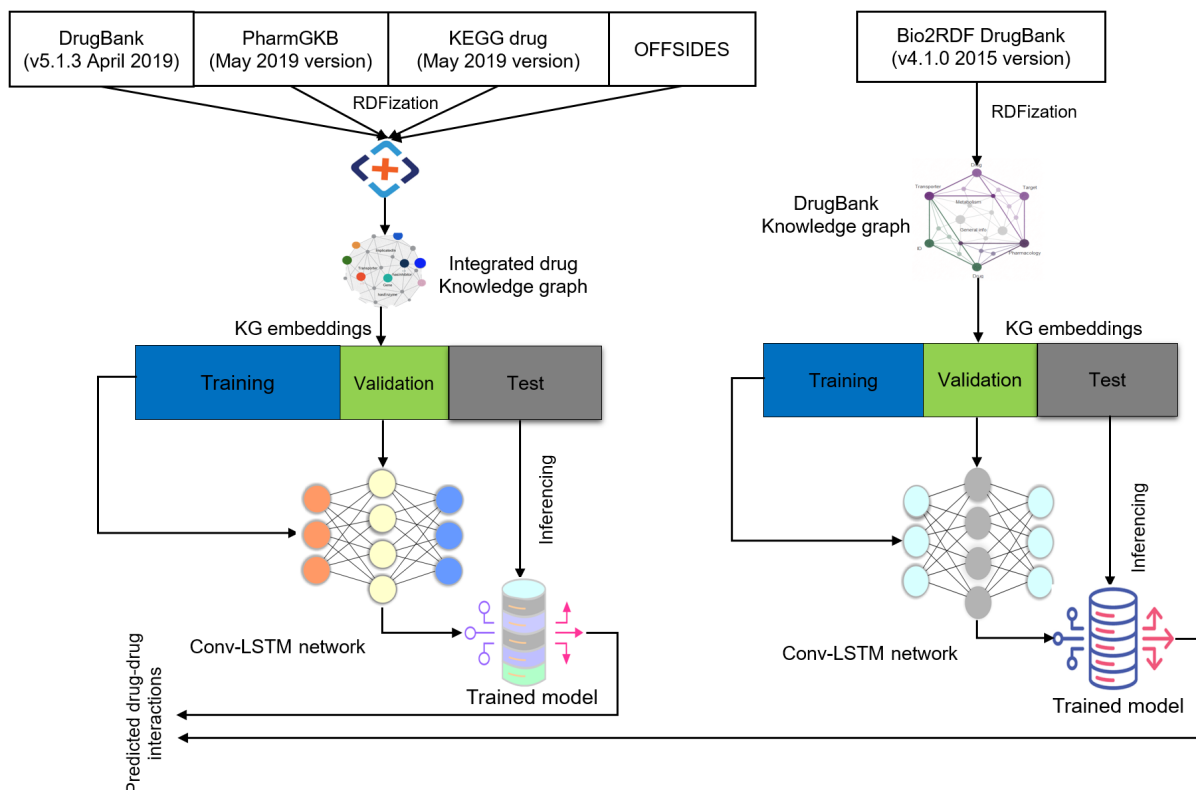


Figure 1: workflow of the proposed approach for predicting DDIs

PharmGKB [24, 32], reports 438,802 drug side-effects. From these sources, we create two datasets: i) the DDI dataset which contains drug-drug interaction pairs, ii) the knowledge graph which we will later use as background knowledge for interactions. The latter does not contain any explicit information about the interactions.

3.2.2 DDI extraction. We employ a semi-supervised approach for extracting DDIs from the sources mentioned above. We parsed the DDI information from the provided XML file from DrugBank and compiled an edge list of drug identifier combinations, which gives us 2,641,889 pairwise DDIs and 2,630,796 unique DDIs spanning 12,112 drugs. Although the KEGG drug database has 10,979 drugs related and 501,689 DDIs relations, mapping to DrugBank identifiers (IDs) results in only 58,205 interactions because of missing mappings.

Data from TWOSIDES [44], which is a comprehensive source of polypharmacy ADRs is also used, but interactions are restricted to those that cannot be ascribed unambiguously to either drug alone. Therefore, a collection of the drug pairs for the interacting compounds built in literature [24] is used in which the PubChem IDs are used to map TWOSIDES IDs to DrugBank IDs. This way, we obtained a list of 19,020 DDIs between 351 compounds and 63,473 distinct pairwise DDIs between 645 drugs.

Dhami et al. [14] have identified that a few DDIs reported in the DrugBank dataset are less evident to interact with each other. We relied on multi-source evidence for those drug pairs and removed from the DrugBank DDI list the contradictory ones. Next, Zhang et

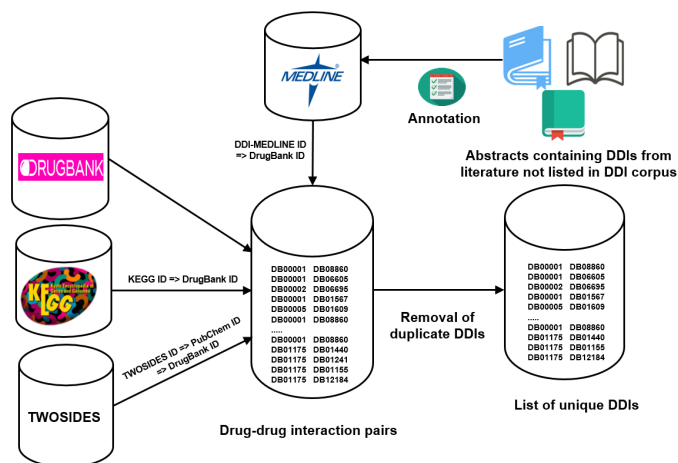


Figure 2: extraction of DDIs from different scientific sources

al. [50] reported 145,068 DDIs² based on label propagation prediction using clinical side-effects. We added these interactions to our dataset. Finally, Sridhar et al. [40] listed top-ranked ten predictions for interactions unknown in DrugBank using their PSL model; also, these pairs were added to our dataset.

²<https://astro.temple.edu/~tua87106/ddi.html>

Table 1: Statistics of drug-drug interactions

Database/source	#DD-interactions
DrugBank	2,630,796
KEGG Drug	40,540
TWOSIDES	82,493
MEDLINE [21], other sources [14, 40, 50, 51]	145,108
Total DDIs	2,898,937

Besides these, we also incorporate the interactions from the 227 MEDLINE abstracts from the DDI corpus [21, 36]. This contributed 327 DDIs based on 1,826 pharmacological substances. Additionally, some abstracts are also used that are not listed in the DDI corpus e.g. [14, 40, 50, 51]. For these, the annotation guidelines³ developed by domain experts were used. A certified pharmacist verified these annotations. The overall DDI dataset consists of information from all these sources combined. An overview of the process is shown in fig. 2 and statistics are collected in table 1 where it should be noted that duplicates between the data sources are removed to obtain the final number of interactions (2,898,937).

3.2.3 KGs construction and integration. To create our integrated knowledge graph, we used data from DrugBank, KEGG, OFFSIDES, and PharmGKB. Although PharmGKB does not contain DDI information, it publishes lexicons of known drug names and synonyms as well as gene and disease terms and data on genetic pathways. The dataset is directed at clinicians and researchers. Previously, Bio2RDF [3] created a large RDF graph that interlinks data from major databases containing biological entities such as drugs, proteins, pathways, and diseases. Using that data would have been an option, but the latest version (i.e., v4.0) is already rather outdated. Instead, we collected the raw DrugBank, KEGG drug, PharmGKB, and OFFSIDES data from the respective portals and converted them into RDF using a modified version of Bio2RDF scripts⁴. Then each RDF KG was uploaded to a blazegraph RDF triplestore in named graph⁵. Then similar to literature [46], federated SPARQL queries are executed based on the ‘billion triples benchmark’ [35] to extract selected triples. For our dataset five types of drug-related entities, namely drugs, genes, proteins, pathways and enzymes, and phenotype (i.e., disease, side-effects), are included. Further, nine types of biological relations are considered: (drug, **hasTarget**, protein), (drug, **hasEnzyme**, gene), (drug, **hasEnzyme**, protein), (drug, **hasTransporter**, protein), (drug, **hasTransporter**, gene⁶), (protein, **isPresentIn**, pathway), (gene, **isPresentIn**, pathway), and (pathway, **isImplicatedIn**, phenotype).

Before the extraction, mappings are created based on `owl:sameAs` and `owl:equivalentProperty` axioms in which respective drug identifiers are mapped to DrugBank IDs as shown in fig. 2. Although genes contain the information needed to make functional molecules called proteins, we considered relation around genes and protein distinct, since PharmGKB contains information about both genes and proteins. The extracted triples are formed the triples

Table 2: Statistics of the data sources

Knowledge graph	#Triples	#Entities	#Relation types
DrugBank	7,740,864	2,116,569	72
KEGG	308,690	107,916	41
OFFSIDES	438,802	875,985	12
PharmGKB	2,793,078	1,583,910	135
Total	11,281,434	4,587,380	260

in our drug KG in the form (subject, predicate, object), indicating that the subject has the specified relation to the object. Since this integrated knowledge graph should not contain any explicit information about drug-drug interactions, there is no information in the form of `drugbank_vocabulary:ddi-interactor-in` and `kegg_vocabulary:Interaction` from the DrugBank and KEGG drug KG, respectively. The number of triples, entities, and relation types for the individual KGs and the integrated KG are given in table 2. Next, we will elaborate on how we prepared this data as an input for our classifiers.

3.3 Knowledge graph embeddings

We used the information of our knowledge graph for predicting the interaction between each pair of drugs. However, ML classifiers do typically expect their input as fixed-length vectors. Hence, we perform a KG embedding procedure to encode the information from the graph into dense vectors. KG embedding consists of three steps: representing entities and relations, defining a scoring function, and learning entity and relation representation [29]. We used RDF2Vec[34], SimpleIE [25], TransE [6], KGloVe [11], CrossE [52], and PBG [27] for the KG embeddings. These representations represent the neighborhood of a node as well as the kind of relations that exist to the neighboring nodes. Since these methods do not incorporate literal information into the embedding, literals are removed from the KG.

RDF2Vec works by first generating a corpus of text by performing uniform random walks starting from each entity in the graph [12]. Then, the corpus C of edge-labeled random walks are used as the input for learning embeddings of each node using the skip-gram (SG) word2vec [30] model⁷. From a given a sequence of drug facts $(w_1, w_2, \dots, w_n) \in C$, the SG model aims to maximize the average log probability L_p (see eq. (2)) according to the context within the fixed-size window, in which c represents a context.

$$L_p = \frac{1}{N} \sum_{n=1}^N \sum_{-c \leq j \leq c, j \neq 0} \log p(w_{n+j}|w_n) \quad (2)$$

To define $p(w_{n+j}|w_n)$, we use negative sampling by replacing $\log p(w_o|w_I)$ with a function to discriminate target words (w_o) from a noise distribution $P_n(w)$ drawing k words from $P_n(w)$:

$$\log \sigma(v_{w_o}^\top v_{w_I}) + \sum_{i=1}^k \mathbb{E}_{w_i \sim P_n(w)} \left[\log \sigma(-v_{w_i}^\top v_{w_I}) \right] \quad (3)$$

³http://hulat.inf.uc3m.es/DrugDDI/annotation_guidelines_ddi_corpus.pdf

⁴<https://github.com/rezacsedu/DDI-prediction-KG-embeddings-Conv-LSTM/scripts>

⁵Zenodo download link: <https://zenodo.org/deposit/3270566>

⁶e.g. Polymorphisms in the ABC drug has transporter gene MDR1

⁷Literature [8, 12] have observed better performance using SG than the CBOW model

The embedding of a concept s occurring in corpus C is the vector v_s in eq. (3) derived by maximizing eq. (2). Besides RDF2Vec, we also trained *TransE* embeddings as a representative of the translation-based KG embedding methods. Here, every entity and relation is embedded as a low-dimensional vector, where the relations are represented as the translation from the head entity to tail entity. The *CrossE* embedding method, which explicitly simulates crossover interactions, is also used. Although both general embeddings for each entity and relation and multiple triple specific embeddings can be generated using *CrossE*, we used only the general embeddings. The *Simple* embedding method, which allows two embeddings of each entity to be learned independently, is also employed. The embeddings learned through *Simple* are interpretable, which help to incorporate drug-related background knowledge into the embeddings. A reported advantage of *Simple* is that it outperforms tensor factorization techniques, especially for link prediction problems.

Learning the representations in a KG relies on contrasting positive instances with negative ones. However, KGs typically include only positive relation instances [39]. A solution to this issue is using implicit negative evidence in which instances that have not been observed in the KGs are considered negative. Kotnis and Nastase et al. [26] employed several negative sampling approaches and observed the impact on the learned embeddings. They found that the “corrupting positives” method is leading to the best result in a link prediction task. This corruption produces negative instances that are closer to the positive ones than those produced through random sampling. Also several methods which we used in this work (*TransE*, *CrossE*, *Simple*, and *RDF2Vec*) generate negative instances by corrupting positive samples.

KGloVe [11] has some similarity with *RDF2Vec*, but uses a different technique to identify global patterns for creating vector space embeddings. First, a co-occurrence matrix is created by personalized computing PageRank for each node. Then, this process is repeated for the graph with all edges reversed. These two matrices are summed together and normalized. Finally, this matrix is used as an input to the *GloVe* word embedding algorithm. In this work, we have only used unbiased walks. As a last model, we train *CompLex* [45] using the *PBG* implementation [27] because our integrated KG contains many triples. Technically, *CompLex* uses only the Hermitian dot product for creating embeddings⁸. The overall embedding method using *CompLex* is arguably simpler but since the composition of complex embeddings can handle a large variety of binary relations (among them symmetric and antisymmetric ones) research has exposed that complex embeddings can outperform several other models.

We trained the *CompLex* embedding model of *PBG* to i) create the embeddings faster, scalable for large graphs, and parallelize the training, and ii) to observe if the embeddings generated by this model are useful for predicting DDIs. Given these dense representations, we can now feed the information from the graph into our machine learning models. To represent the feature vector of a drug pair, we concatenate the embedding vectors of each drug in the pair in which the negative samples are generated by corrupting positive edges by sampling either a new source or a destination for each existing edge [27].

⁸Complex counterpart of the standard dot product between real vectors

3.4 Network construction

We train various baseline ML models, which we will use as baselines later. Here, we describe the more complex neural network architecture which gave the best results we obtained. We construct a so-called *Conv-LSTM* network [37] by combining both CNN and LSTM layers as shown in fig. 3. While CNN uses convolutional filters to capture local relationship values in drug features, the LSTM network can carry overall relationships from the features extracted by CNN. The *Conv-LSTM* has shown good performance on diverse prediction tasks such as hate speech detection from text [53], for precipitation nowcasting [37], and for monocular depth prediction [13]. Consequently, it has been able to capture the most significant drug features in our case.

We extended the *Conv-LSTM* network proposed in literature [37] in which each input X_1, X_2, \dots, X_t , cell outputs C_1, C_2, \dots, C_t , hidden states $\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_t$, and gates i_t, f_t, o_t of the network are 2D tensors whose dimensions are spatial dimensions of the drug features. *Conv-LSTM* determines the future state of a certain cell in the input hyperspace by the inputs and past states of its local neighbors. This is achieved by using a conv operator in the state-to-state and input-to-state transitions as represented as follows [37]:

$$i_t = \sigma(W_{xi} * X_t + W_{hi} * \mathcal{H}_{t-1} + W_{ci} \circ C_{t-1} + b_i) \quad (4)$$

$$f_t = \sigma(W_{xf} * X_t + W_{hf} * \mathcal{H}_{t-1} + W_{cf} \circ C_{t-1} + b_f) \quad (5)$$

$$C_t = f_t \circ C_{t-1} + i_t \circ \tanh(W_{xc} * X_t + W_{hc} * \mathcal{H}_{t-1} + b_c) \quad (6)$$

$$o_t = \sigma(W_{xo} * X_t + W_{ho} * \mathcal{H}_{t-1} + W_{co} \circ C_t + b_o) \quad (7)$$

$$\mathcal{H}_t = o_t \circ \tanh(C_t) \quad (8)$$

In the above equations, $*$ denotes the conv operator, and \circ is the entrywise multiplication of two matrices of same dimensions. The second LSTM layer emits an output ‘ \mathcal{H} ’, which is then reshaped (i.e., flatten) into a feature sequence and fed into fully-connected layers to predict the DDIs at the next step and as an input at the next time step. The first layer is the embedding layer, which maps a drug sample as a ‘sequence’ into a real vector domain. Then the embedding representation with a shape of 100x300 is fed into a 1D convolutional layer, which has 100 filters and a kernel-size of 4.

The output of each conv layer is then passed to the dropout layer to regularize learning to avoid overfitting. Intuitively, these can be thought of as forcing the classifier not to rely on any trivial individual drug features. The conv layer convolves the input feature space into a 100x100 representation, which is further down-sampled by the 1D max pooling layer (MPL) having a pool size of 4 along the embedding dimension, producing an output of shape 25x100. Where each of the 25 dimensions can be considered as an ‘extracted feature.’ The MPL flattens the output space by taking the highest value in each timestep dimension, which produces a 1x100 vector containing drug features that are highly indicative of interest. Contrarily, LSTM layer treats flattened feature vector’s dimension as timesteps and outputs 100 hidden units per timestep. Then using a global MPL, the most influential features are fed into a fully-connected layer after passing through another dropout layer and finally to a softmax layer which generates the probability distribution over the classes. Additionally, we introduce Gaussian noise [49] into each conv, LSTM, and dense layer to improve the model generalization.

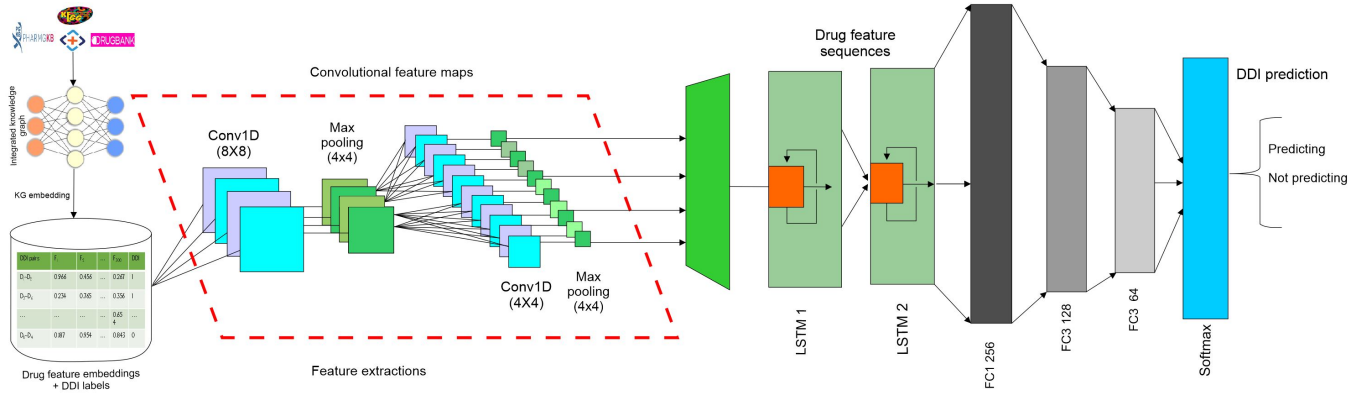


Figure 3: A schematic representation of the Conv-LSTM network, which starts from taking input into an n-dimensional embedding space and passing to both CNN and LSTM layers before getting the vector representation of the most important features to be fed through dense, dropout, Gaussian noise, and Softmax layers for predicting possible drug-drug interactions.

3.5 Network training

Since all the classifiers need both negative and positive samples for the link prediction problem, previous studies have randomly chosen negative samples from unknown interactions [40, 50]. However, setting all the unknown interactions as negative samples creates a data imbalance issue. Consequently, performance metrics, such as AUPR and F1-score, get influenced [8]. Other research has tackled this issue through random undersampling from the unknown interactions at a ratio corresponding to the positive set [9], or inferring negatives by unsupervised clustering analysis [20].

The open source implementations of PBG⁹, CrossE¹⁰, TransE¹¹, and SimpleIE¹² were used for the KG embedding with the default parameters provided. On the other hand, the modified version of KGloVe¹³ is used, which converged at 400th iteration. While RDF2Vec was trained using skip-gram by setting window size = 5 with graph walk at depth 5 and 500 walks per entity. Each embedding methods is employed by setting the dimension of the feature vector to 300 by varying negative samples. By filtering out the drugs that have no calculated feature vector, we were able to extract the features for 12,439 drugs out of 12,664 drugs. The embeddings generated by RDF2Vec, TransE, PBG, KGloVe, CrossE, and SimpleIE are then used to train the Conv-LSTM network for the link prediction similar to [2], in which we aim to estimate the probability that a relation or link with label l exists between vertices v_1 and v_2 given their vector representation, $V(v_1)$ and $V(v_2)$: $p(v_1, v_2, l) \in E | \langle V(v_1), V(v_2) \rangle$.

$$L = \sum_{i,j \in Y} -y_{i,j} \log \hat{y}_{i,j} - (1 - y_{i,j}) \log (1 - \hat{y}_{i,j}) \quad (9)$$

First-order gradient-based optimization techniques Adam, AdaGrad, RMSprop, and AdaMax with varying learning rates and different batch sizes are used to learn model parameters, which tries

to optimize the binary cross-entropy loss eq. (9). The hyperparameters optimization is done based on random search and cross-validation in which the model is trained on a batch size of 128 wherein each of 5 runs 70% of the data is used for the training, 30% for evaluating the network, and 10% from the training set is randomly used for the validation.

4 EXPERIMENTS

The evaluation code¹⁴ was written in Python. The software stack consists of Scikit-learn, PyTorch, and Keras with the TensorFlow backend. The network training is carried out on an Nvidia GTX 1080i GPU with CUDA and cuDNN enabled. We also trained LR, KNN, NB, SVM, RF, and GBT as ML baseline models. Similar to Conv-LSTM network, we perform the hyperparameter optimization for these classifiers through a random search and 5-fold cross-validation tests. For the experiment, 80% of the data is used for the training using 5-fold cross-validation and evaluate the optimized model on 20% held-out data in which the best hyperparameters were produced through a random search. Although AUC score is used commonly as a performance metric in previous studies, literature has emphasized that it might not be sufficiently accurate for imbalanced data [8, 24]. Therefore, we used the area under the precision-recall curve (AUPR), and Matthias correlation coefficient (MCC) along with the AUC and F1-score to measure the performance of the classifiers. Finally, we use the model averaging ensemble (MAE) of top-3 models to report the final prediction.

4.1 Analysis of DDIs predictions

Table 3 summarizes the results of the prediction task based on different embedding methods. A general observation is that the Conv-LSTM model outperformed all baseline models, in the best case resulting in an AUPR of 0.93. Also, overall LR, NB, KNN, and SVM models performed worst. Although these algorithms are intrinsically simple, have low variance, and less prone to over-fitting, feature selector based on these may be discriminating drug-related features very aggressively, which forces these classifiers to lose some

⁹<https://github.com/facebookresearch/PyTorch-BigGraph>

¹⁰<https://github.com/wencolani/CrossE>

¹¹<https://github.com/xjdwj/TransE-Pytorch>

¹²<https://github.com/baharefatemi/SimpleIE>

¹³<https://github.com/miselo/globalRDFEmbeddingsISWC>

¹⁴Source code: <https://github.com/rezacsedu/DDI-prediction-KG-embeddings-Conv-LSTM>

Table 3: Prediction comparison across embedding methods.
 * signifies the embedding method giving best accuracy.

Embedding	Model	AUPR	F1-score	MCC
PBG*	LR	0.74	0.72	0.53
	NB	0.73	0.70	0.55
	SVM	0.80	0.81	0.69
	KNN	0.69	0.71	0.51
	GBT	0.88	0.85	0.72
	RF	0.91	0.92	0.75
	Conv-LSTM	0.93	0.91	0.79
	MAE	0.94	0.92	0.80
SimpleIE	LR	0.76	0.72	0.59
	NB	0.73	0.72	0.55
	SVM	0.80	0.81	0.66
	KNN	0.73	0.73	0.53
	GBT	0.87	0.86	0.74
	RF	0.89	0.88	0.77
	Conv-LSTM	0.91	0.90	0.78
	MAE	0.92	0.91	0.79
KGloVe	LR	0.75	0.73	0.54
	NB	0.72	0.71	0.53
	SVM	0.78	0.79	0.68
	KNN	0.71	0.69	0.53
	GBT	0.87	0.85	0.71
	RF	0.89	0.86	0.73
	Conv-LSTM	0.87	0.89	0.74
	MAE	0.89	0.90	0.75
TransE	LR	0.72	0.71	0.57
	NB	0.69	0.70	0.51
	SVM	0.75	0.74	0.64
	KNN	0.63	0.59	0.49
	GBT	0.83	0.82	0.69
	RF	0.84	0.85	0.71
	Conv-LSTM	0.86	0.87	0.73
	MAE	0.87	0.88	0.74
CrossE	LR	0.65	0.68	0.47
	NB	0.70	0.71	0.50
	SVM	0.72	0.73	0.55
	KNN	0.69	0.66	0.52
	GBT	0.81	0.82	0.65
	RF	0.82	0.83	0.66
	Conv-LSTM	0.85	0.84	0.67
	MAE	0.86	0.85	0.69
RDF2Vec	LR	0.65	0.68	0.47
	NB	0.69	0.70	0.48
	SVM	0.71	0.72	0.52
	KNN	0.67	0.64	0.50
	GBT	0.79	0.80	0.61
	RF	0.80	0.81	0.62
	Conv-LSTM	0.83	0.82	0.67
	MAE	0.84	0.85	0.69

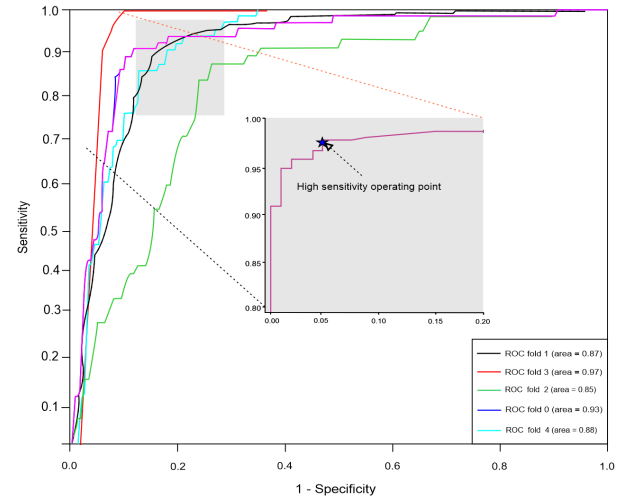


Figure 4: ROC curves of cross-validated Conv-LSTM model

useful drug features which result in worse performance. Among the tree-based classifiers, RF performs the best, showing an F1-score of 0.91, which is the best among the ML baselines. The model averaging ensemble of top-3 models (i.e., GBT, RF, and Conv-LSTM) boosts the performance by 1.5% compared to the best Conv-LSTM model in terms of F1 score.

Interestingly, the MCC scores by all classifiers also show the prediction was strongly correlated with the ground truth (measured using a Pearson product-moment correlation coefficient we obtained 0.70), probably because the embeddings generated by the embedding methods are learnable quality drug features. The AUC score generated by the Conv-LSTM network is found to be the highest, which is at least 3% better than the second-best score by the RF classifier, where the LR classifier performed the worst. The ROC curve in fig. 4 shows consistent AUC scores across the folds, which signifies that the predictions are much better than random guessing.

4.2 Comparison of graph embedding methods

As seen in table 3, the classifiers work better with drug features generated by the PBG, SimpleIE, and KGloVe methods. In particular, the GBT, RF, and Conv-LSTM classifiers perform consistently best on the embeddings generated by PBG in terms of F1, MCC, and AUPR scores. In contrast, using features generated by the RDF2Vec method, we experience the worst DDIs prediction accuracy. This is different from earlier work [8], where the best results were obtained with RDF2Vec with uniform weighting setting.

We suspect that the classifiers did not benefit much from more training samples in our case. Therefore, we validate this by calibrating the best performing Conv-LSTM classifier against different embedding methods for which the output probability of the classifier can be directly interpreted as a confidence level in terms of ‘fraction of positives’; the result is illustrated in fig. 5. As seen, the Conv-LSTM classifier gave a probability value between 0.82 to 0.93, which means 93% predictions belong to true positive predictions generated by the embeddings from PBG.

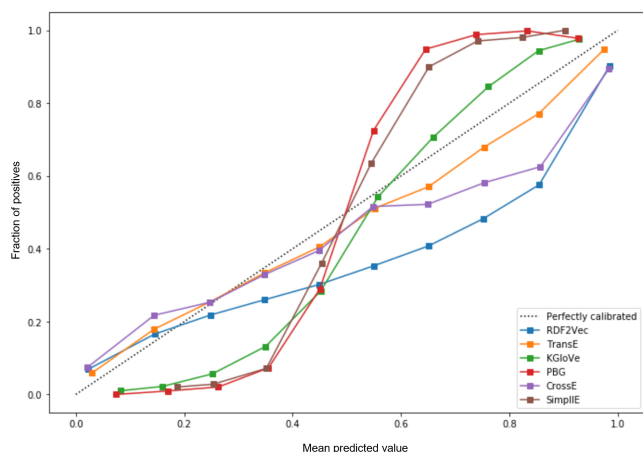


Figure 5: Calibrating Conv-LSTM with embedding methods

4.3 Effects of number of drug samples

To understand the effects of having more training samples, and to understand whether our classifiers suffer more from variance errors or bias errors, we observed the learning curves of top-3 classifiers (i.e., RF, GBT, and Conv-LSTM) and SVM (a linear model) for varying numbers of training samples. As shown in fig. 6, for SVM the validation and training scores converge to a low value with increasing size of the training set. Consequently, SVM did not benefit much from more training samples. However, RF and GBT are tree-based ensemble methods, and the Conv-LSTM network can learn more complex concepts from the drug features. This results in a lower bias, which can be observed from higher training scores than the validation scores for the maximum number of drug samples, i.e., adding more training samples does increase generalization.

4.4 Influence of negative samples

Inspired by [45], we investigated the influence of the number of negatives per positive training sample, which we call σ , for ComplEx. As we already varied σ per training sample and set it to 15, we further varied again the σ in [5, 10, 20, 25] and collected the embeddings for each setting again. Then we observed if the Conv-LSTM network performs better with the larger σ . Generating more negatives samples moderately improves the results. In particular, with 20 negative triples, we observed about 1% accuracy boost in terms of AUPR. Embedding training also converged slightly faster. Further increasing σ to 25 results in a drop in AUPR.

4.5 Comparison with state-of-the-art

Since our approach of data collection and preparation are different from other approaches and we have more samples, a one-to-one comparison was not viable –especially with Tiresias framework [1], PRD[46], and INDI [17]. Kastrin et al. [24] used data from multiple sources, but evaluated and inferred unknown DDIs from TWO-SIDES only. With that dataset, they achieved the best AUPR score of 0.93 using RF and GBT classifiers.

The Tiresias framework, which uses both pharmacological similarities from embedding features, has reported an F1-score of 0.85

and AUPR of 0.92. Their pharmacological similarity features are equivalent to INDI, which also uses the DrugBank v4.0 dataset. INDI evaluated the performance of DDI prediction on a total of 37,212 true DDIs. They obtained AUC scores of 0.93 and an F1-score of 0.89, omitting the interaction type (i.e., PD or PK) using a 10-fold cross validation setting. Remzi et al. [8] observed an F1-score of 0.867 and AUPR of 0.918 using DrugBank v4.0 dataset.

With our approach, evaluations against several baseline models yield an AUPR of up to 0.94, an F1-score of 0.92, and an MCC of 0.80 during 5-fold cross-validation tests. This signifies that a KG-based approach using multiple data sources is comparable to current state-of-the-art methods. To show the benefit of using a more robust classifier, we further trained the Conv-LSTM network with DrugBank v4.0 dataset as shown in fig. 1. During a 5-fold cross-validation test, we observed slightly better accuracy, namely an F1-score of 0.895 and AUPR of 0.926, which outperforms [8].

5 CONCLUSION AND OUTLOOK

Adverse drug reactions are very dangerous and lead to a significant number of the hospital (re-)admissions and even deaths. Many of these reactions are due to drug-drug interactions. Preferably, all drug-drug interactions should be known upfront to ensure that preventable cases do not occur. However, it is not feasible to investigate all possible interactions, and hence approaches able to predict possible interactions are investigated. In this paper, we proposed the use of knowledge graphs to integrate drug-related data from different sources. This way, we have integrated background knowledge about drugs, diseases, pathways, proteins, enzymes, chemical structures, etc. Since this background data is in a format which cannot be used as a direct input for typical classifiers, we applied several node embedding techniques to create a dense vector representation for each node in the KG. These representations are then fed into various traditional ML classifiers and a specifically designed neural network architecture based on a convolutional-LSTM.

Our core observations are that i) We could outperform the baseline classifiers, as well as earlier state-of-the-art models, consistently with our proposed architecture. We obtained up to 0.94, 0.92, 0.80 for AUPR, F1-score, and MCC, respectively, during 5-fold cross-validation tests showing high confidence at predicting potential DDIs. ii) From the embedding models we used, the PBG model did perform best, but also SimpleE and KGloVe gave reasonable results.

One limitation of our approach is the inability to provide explanations for the predicted DDIs. The embedding creates latent features, which are like a black-box model. As future research directions, we see i) the possibility to include even more data to the background. For example, NDFRT, SemMedDB, and SIDE; also a large ablation study to measure the influence of each of these additions would be useful, ii) also including explicit information about negative drug-drug interaction, as well as a prediction of the interaction type, iii) providing explanations for the interactions, iv) further investigation of other models to perform predictions on graphs, and v) interaction with other (non-drug) substances like food.

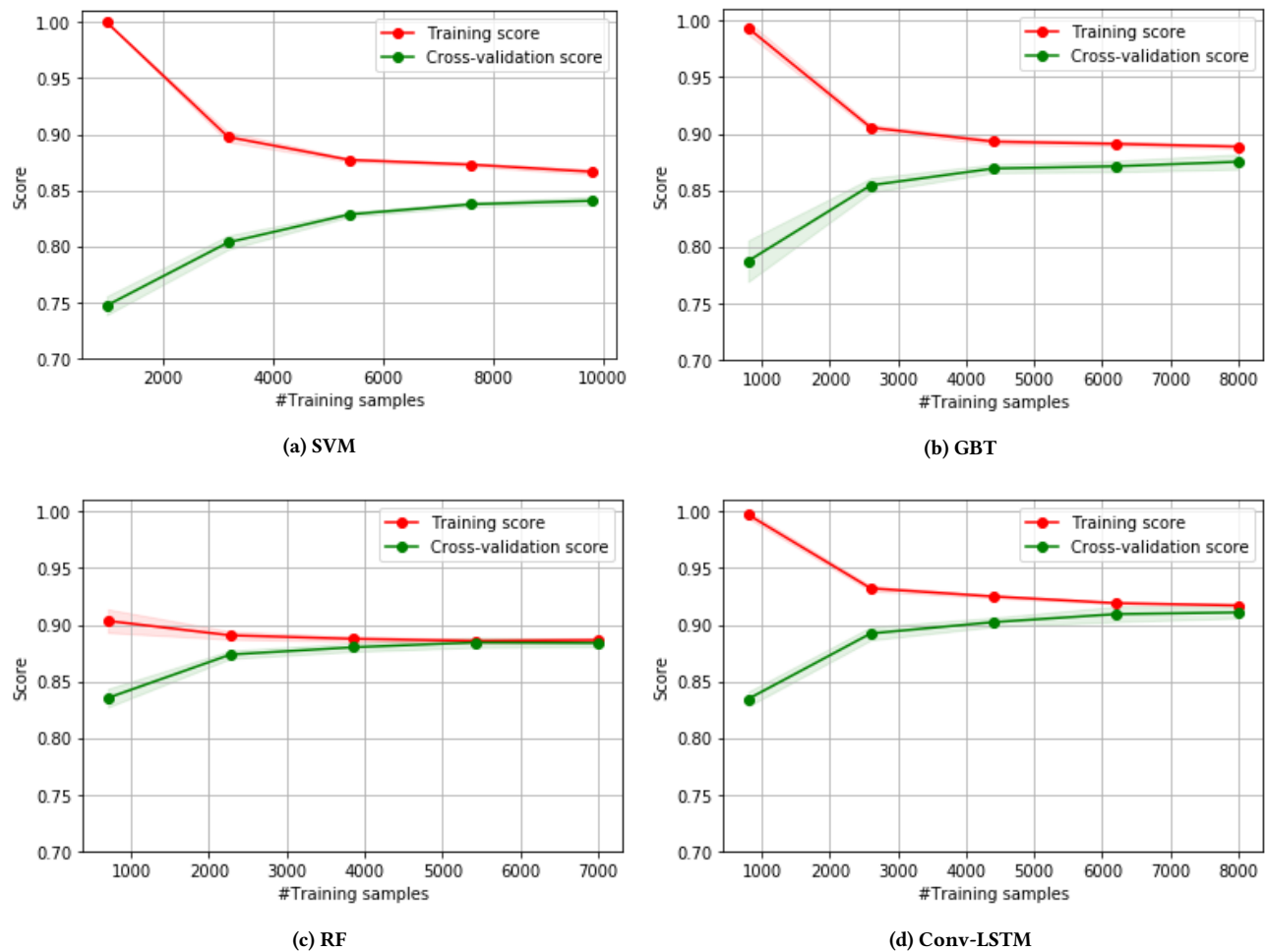


Figure 6: Learning curves showing the validation and training scores of top-3 and SVM classifier

REFERENCES

- [1] Ibrahim Abdelaziz, Achille Fokoue, Ping Zhang, and Mohammad Sadoghi. 2017. Large-scale structural and textual similarity-based mining of knowledge graph to predict drug–drug interactions. *Journal of Web Semantics* 44 (2017), 104–117.
- [2] Mona Alshahrani, Mohammad Asif Khan, Omar Maddouri, Akira R Kinjo, NÁzria Queral-Rosinach, and Robert Hoehndorf. 2017. Neuro-symbolic representation learning on biological knowledge graphs. *Bioinformatics* 33, 17 (04 2017), 2723–2730. <https://doi.org/10.1093/bioinformatics/btx275>
- [3] François Belleau, Marc-Alexandre Nolin, Nicole Tourigny, Philippe Rigault, and Jean Morissette. 2008. Bio2RDF: towards a mashup to build bioinformatics knowledge systems. *Journal of biomedical informatics* 41, 5 (2008), 706–716.
- [4] Jari Björne, Suwisa Kaewphan, and Tapio Salakoski. 2013. UTurku: Drug Named Entity Recognition and Drug-Drug Interaction Extraction Using SVM Classification and Domain Knowledge. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. Association for Computational Linguistics, Atlanta, Georgia, USA, 651–659.
- [5] Tamara Babić, Juliane Fluck, and Martin Hofmann-Apitius. 2013. SCAI: Extracting drug–drug interactions using a rich feature vector. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. Association for Computational Linguistics, Atlanta, Georgia, USA, 675–683.
- [6] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *Advances in Neural Information Processing Systems* 26, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger (Eds.). Curran Associates, Inc., 2787–2795. <http://papers.nips.cc/paper/5071-translating-embeddings-for-modeling-multi-relational-data.pdf>
- [7] Aurel Cami, Shannon Manzi, and Ben Y Reis. 2013. Pharmacointeraction network models predict unknown drug–drug interactions. *PloS one* 8, 4 (2013), e61468.
- [8] Remzi Celebi, Erkan Yasar, Huseyin Uyar, Ozgur Gumus, Oguz Dikenelli, and Michel Dumontier. 2018. Evaluation of Knowledge Graph Embedding Approaches for Drug-Drug Interaction Prediction using Linked Open Data. (2018).
- [9] Feixiong Cheng and Zhongming Zhao. 2014. Machine learning-based prediction of drug–drug interactions by integrating drug phenotypic, therapeutic, chemical, and genomic properties. *Journal of the American Medical Informatics Association* 21, e2 (2014), e278–e286.
- [10] Md. Faisal Mahbub Chowdhury and Alberto Lavelli. 2013. FBK-irst : A Multi-Phase Kernel Based Approach for Drug-Drug Interaction Detection and Classification that Exploits Linguistic Information. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. Association for Computational Linguistics, Atlanta, Georgia, USA, 351–355.
- [11] Michael Cochez, Petar Ristoski, and Heiko Paulheim. 2017. Global RDF Vector Space Embeddings. In *The Semantic Web – ISWC 2017: 16th International Semantic Web Conference, Vienna, Austria, October 21–25, 2017*, Claudia d’Amato, Miriam Fernandez, et al. (Eds.). Springer International Publishing, Cham, 190–207. https://doi.org/10.1007/978-3-319-68288-4_12
- [12] Michael Cochez, Petar Ristoski, Simone Paolo Ponzetto, and Heiko Paulheim. 2017. Biased Graph Walks for RDF Graph Embeddings. In *Proceedings of the 7th International Conference on Web Intelligence, Mining and Semantics (WIMS '17)*.

- ACM, New York, NY, USA, Article 21, 12 pages. <https://doi.org/10.1145/3102254.3102279>
- [13] Arun CS Kumar, Suchendra M Bhandarkar, and Mukta Prasad. 2018. Depthnet: A recurrent neural network architecture for monocular depth prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. The Computer Vision Foundation, Salt Lake City, Utah, 283–291.
 - [14] Devendra Singh Dhami, Gautam Kunapuli, Mayukh Das, David Page, and Sri-ram Natarajan. 2018. Drug-Drug Interaction Discovery: Kernel Learning from Heterogeneous Similarities. *Smart Health* 9 (2018), 88–100.
 - [15] Jon D Duke, Xu Han, Zhiping Wang, Abhinata Subhadarshini, Shreyas D Karnik, Xiaochun Li, Stephen D Hall, Yan Jin, J Thomas Callaghan, Marcus J Overhage, et al. 2012. Literature based drug interaction prediction with clinical assessment using electronic medical records: novel myopathy associated drug interactions. *PLoS computational biology* 8, 8 (2012), e1002614.
 - [16] Jun Feng, Minlie Huang, Mingdong Wang, Mantong Zhou, Yu Hao, and Xiaoyan Zhu. 2016. Knowledge Graph Embedding by Flexible Translation. In *Proceedings of the Fifteenth International Conference on Principles of Knowledge Representation and Reasoning (KR'16)*. AAAI Press, 557–560.
 - [17] Assaf Gottlieb, Gideon Y Stein, Yoram Oron, Eytan Ruppin, and Roded Sharan. 2012. INDI: a computational framework for inferring drug interactions and their associated recommendations. *Molecular systems biology* 8, 1 (2012), 592.
 - [18] Negacy Hailu, Lawrence E. Hunter, and K. Bretonnel Cohen. 2013. UCoLorado_SOM: Extraction of Drug-Drug Interactions from Biomedical Text using Knowledge-rich and Knowledge-poor Features. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. Association for Computational Linguistics, Atlanta, Georgia, USA, 684–688.
 - [19] Sophie Hallstedt, Nikita Makarov, Hossein Samieadeh, Maria Pellegrino, Martina Garofalo, and Michael Cochez. 2018. Strategies to connect RDF Graphs for Link Prediction using Drug-Disease Knowledge Graphs. <https://doi.org/10.6084/m9.figshare.7429094.v1> Poster presented at the 11th International Conference Semantic Web Applications and Tools for Life Sciences (SWAT4HCLS 2018).
 - [20] Pathima Nusrath Hameed, Karin Verspoor, Snezana Kusljic, and Saman Halgamauge. 2017. Positive-unlabeled learning for inferring drug interactions based on heterogeneous attributes. *BMC bioinformatics* 18, 1 (2017), 140.
 - [21] Maria Herrero-Zazo, Isabel Segura-Bedmar, and Thierry Declercq. 2013. The DDI corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of biomedical informatics* 46, 5 (2013), 914–920.
 - [22] Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Knowledge Graph Embedding via Dynamic Mapping Matrix. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, Beijing, China, 687–696. <https://doi.org/10.3115/v1/P15-1067>
 - [23] Minoru Kanehisa, Susumu Goto, Mao Tanabe, and Mika Hirakawa. 2009. KEGG for representation and analysis of molecular networks involving diseases and drugs. *Nucleic acids research* 38, suppl_1 (2009), D355–D360.
 - [24] Andrej Kastrin, Polonca Ferik, and Brane Leskošek. 2018. Predicting potential drug-drug interactions on topological and semantic similarity features using statistical learning. *PLoS one* 13, 5 (2018), e0196865.
 - [25] Seyed Mehran Kazemi and David Poole. 2018. Simple Embedding for Link Prediction in Knowledge Graphs. In *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.). Curran Associates, Inc., 4284–4295. <http://papers.nips.cc/paper/7682-simple-embedding-for-link-prediction-in-knowledge-graphs.pdf>
 - [26] Bhushan Kotnis and Vivi Nastase. 2017. Analysis of the impact of negative sampling on link prediction in knowledge graphs. *arXiv preprint arXiv:1708.06816* (2017).
 - [27] Adam Lerer, Ledell Wu, Jiajun Shen, Timothee Lacroix, Luca Wehrstedt, Abhijit Bose, and Alex Peysakhovich. 2019. PyTorch-BigGraph: A Large-scale Graph Embedding System. *arXiv preprint arXiv:1903.12287* (2019).
 - [28] Peng Li, Chao Huang, Yingxue Fu, Jinan Wang, Ziyin Wu, Jinlong Ru, Chunli Zheng, Zihu Guo, Xuetong Chen, Wei Zhou, et al. 2015. Large-scale exploration and analysis of drug combinations. *Bioinformatics* 31, 12 (2015), 2007–2016.
 - [29] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning Entity and Relation Embeddings for Knowledge Graph Completion. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI'15)*. AAAI Press, 2181–2187. <http://dl.acm.org/citation.cfm?id=2886521.2886624>
 - [30] Tomas Mikolov, Kai Chen, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).
 - [31] Bethany Percha and Russ Altman. 2012. Discovery and explanation of drug-drug interactions via text mining. In *Biocomputing 2012*. World Scientific, 410–421.
 - [32] Bethany Percha and Russ B Altman. 2013. Informatics confronts drug–drug interactions. *Trends in pharmacological sciences* 34, 3 (2013), 178–184.
 - [33] Majid Rastegar-Mojarad, Richard D. Boyce, and Rashmi Prasad. 2013. UWM-TRIADS: Classifying Drug-Drug Interactions with Two-Stage SVM and Post-Processing. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. Association for Computational Linguistics, Atlanta, Georgia, USA, 667–674.
 - [34] Petar Ristoski and Heiko Paulheim. 2016. RDF2Vec: RDF Graph Embeddings for Data Mining. In *The Semantic Web – ISWC 2016*, Paul Groth, Elena Simperl, Alasdair Gray, Marta Sabou, Markus Krötzsch, Freddy Lecue, Fabian Flöck, and Yolanda Gil (Eds.). Springer International Publishing, Cham, 498–514.
 - [35] Muhammad Saleem, Ali Hasnain, and Axel Ngonga Ngomo. 2018. Largerdf-bench: a billion triples benchmark for sparql endpoint federation. *Journal of Web Semantics* 48 (2018), 85–125.
 - [36] Isabel Segura-Bedmar, Paloma Martinez, and Maria Herrero-Zazo. 2014. Lessons learnt from the DDIExtraction-2013 shared task. *Journal of biomedical informatics* 51 (2014), 152–164.
 - [37] Xingjian SHI, Zhouong Chen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun WOO. 2015. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. In *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett (Eds.). Curran Associates, Inc., Montreal, Canada, 802–810.
 - [38] Guy Shtat, Lior Rokach, and Bracha Shapira. 2019. Detecting drug-drug interactions using artificial neural networks and classic graph similarity measures. *arXiv preprint arXiv:1903.04571* (2019).
 - [39] Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. In *Advances in neural information processing systems*. 926–934.
 - [40] Dhanya Sridhar, Shobeir Fakhraei, and Lise Getoor. 2016. A probabilistic approach for collective similarity-based drug–drug interaction prediction. *Bioinformatics* 32, 20 (2016), 3175–3182.
 - [41] Takako Takeda, Ming Hao, Tiejun Cheng, Stephen H Bryant, and Yanli Wang. 2017. Predicting drug–drug interactions through drug structural similarities and interaction networks incorporating pharmacokinetics and pharmacodynamics knowledge. *Journal of cheminformatics* 9, 1 (2017), 16.
 - [42] Luis Tari, Saadat Anwar, Shanshan Liang, James Cai, and Chitta Baral. 2010. Discovering drug–drug interactions: a text-mining and reasoning approach based on properties of drug metabolism. *Bioinformatics* 26, 18 (2010), i547–i553.
 - [43] Nicholas P Tattonetti, P Ye Patrick, Roxana Daneshjou, and Russ B Altman. 2012. Data-driven prediction of drug effects and interactions. *Science translational medicine* 4, 125 (2012), 125ra31–125ra31. <https://doi.org/10.1126/scitranslmed.3003377>
 - [44] Nicholas P Tattonetti, P Ye Patrick, Roxana Daneshjou, and Russ B Altman. 2012. Data-driven prediction of drug effects and interactions. *Science translational medicine* 4, 125 (2012), 125ra31–125ra31.
 - [45] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Eric Gaussier, and Guillaume Bouchard. 2016. Complex Embeddings for Simple Link Prediction. In *Proceedings of The 33rd International Conference on Machine Learning (Proceedings of Machine Learning Research)*, Maria Florina Balcan and Kilian Q. Weinberger (Eds.), Vol. 48. PMLR, New York, New York, USA, 2071–2080.
 - [46] Meng Wang. 2017. Predicting Rich Drug-Drug Interactions via Biomedical Knowledge Graphs and Text Jointly Embedding. *arXiv preprint arXiv:1712.08875* (2017).
 - [47] Michelle Whirl-Carrillo, Ellen M McDonagh, JM Hebert, Li Gong, K Sangkuhl, CF Thorn, Russ B Altman, and Teri E Klein. 2012. Pharmacogenomics knowledge for personalized medicine. *Clinical Pharmacology & Therapeutics* 92, 4 (2012), 414–417.
 - [48] David S Wishart, Yannick D Feunang, An C Guo, Elvis J Lo, Ana Marcu, Jason R Grant, Tanvir Sajed, Daniel Johnson, Carin Li, Zinat Sayeeda, et al. 2017. DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic acids research* 46, D1 (2017), D1074–D1082.
 - [49] Junyuan Xie, Linli Xu, and Enhong Chen. 2012. Image Denoising and Inpainting with Deep Neural Networks. In *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (Eds.). Curran Associates, Inc., 341–349. <http://papers.nips.cc/paper/4686-image-denoising-and-inpainting-with-deep-neural-networks.pdf>
 - [50] Ping Zhang, Fei Wang, Jianying Hu, and Robert Sorrentino. 2015. Label propagation prediction of drug-drug interactions based on clinical side effects. *Scientific reports* 5 (2015), 12339.
 - [51] Wen Zhang, Yanlin Chen, Feng Liu, Fei Luo, Gang Tian, and Xiaohong Li. 2017. Predicting potential drug-drug interactions by integrating chemical, biological, phenotypic and network data. *BMC bioinformatics* 18, 1 (2017), 18.
 - [52] Wen Zhang, Bibek Paudel, Wei Zhang, Abraham Bernstein, and Huajun Chen. 2019. Interaction Embeddings for Prediction and Explanation in Knowledge Graphs. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (WSDM '19)*. ACM, New York, NY, USA, 96–104. <https://doi.org/10.1145/3289600.3291014>
 - [53] Ziqi Zhang and Lei Luo. 2018. Hate speech detection: A solved problem? The challenging case of long tail on Twitter. *Semantic Web Pre-press, Preprint* (2018), 1–21.