*Bioinformatics*, 37(17), 2021, 2651–2658
doi: 10.1093/bioinformatics/btab169
Advance Access Publication Date: 15 March 2021
Original Paper

Gene expression

# MUFFIN: multi-scale feature fusion for drug–drug interaction prediction

**Yujie Chen, Tengfei Ma, Xixi Yang, Jianmin Wang , Bosheng Song * and Xiangxiang Zeng***

School of Computer Science and Engineering, Hunan University, Changsha 410012, China

*To whom correspondence should be addressed.

Associate Editor: Alfonso Valencia

Received on November 25, 2020; revised on February 5, 2021; editorial decision on March 5, 2021; accepted on March 11, 2021

## Abstract

**Motivation:** Adverse drug–drug interactions (DDIs) are crucial for drug research and mainly cause morbidity and mortality. Thus, the identification of potential DDIs is essential for doctors, patients and the society. Existing traditional machine learning models rely heavily on handcraft features and lack generalization. Recently, the deep learning approaches that can automatically learn drug features from the molecular graph or drug-related network have improved the ability of computational models to predict unknown DDIs. However, previous works utilized large labeled data and merely considered the structure or sequence information of drugs without considering the relations or topological information between drug and other biomedical objects (e.g. gene, disease and pathway), or considered knowledge graph (KG) without considering the information from the drug molecular structure.

**Results:** Accordingly, to effectively explore the joint effect of drug molecular structure and semantic information of drugs in knowledge graph for DDI prediction, we propose a multi-scale feature fusion deep learning model named MUFFIN. MUFFIN can jointly learn the drug representation based on both the drug-self structure information and the KG with rich bio-medical information. In MUFFIN, we designed a bi-level cross strategy that includes cross- and scalar-level components to fuse multi-modal features well. MUFFIN can alleviate the restriction of limited labeled data on deep learning models by crossing the features learned from large-scale KG and drug molecular graph. We evaluated our approach on three datasets and three different tasks including binary-class, multi-class and multi-label DDI prediction tasks. The results showed that MUFFIN outperformed other state-of-the-art baselines.

**Availability and implementation**: The source code and data are available at https://github.com/xzenglab/MUFFIN.

**Contact:** boshengsong@hnu.edu.cn or xzeng@hnu.edu.cn

## 1 Introduction

Drug–drug interaction (DDI)-induced adverse drug reaction (ADR) may increase morbidity and mortality (Giacomini *et al.*, 2007; Plumpton *et al.*, 2016). Thus, the identification of potential DDIs is essential. In recent years, several methods have been employed for DDI prediction. The fundamental methods for predicting DDIs are the traditional laboratory-based methods. Considering that these lines of methods are labor-intensive, time-consuming and costly (Whitebread *et al.*, 2005), the ability to discover potential DDIs is greatly limited. Therefore, accurate and dependable computational methods should be discovered.

Machine learning, an emerging computational approach in recent years, has been widely used to predict DDIs. Existing machine learning-based methods predict potential DDIs by exploiting different drug-related similarity features, such as molecular structure (Takeda *et al.*, 2017; Vilar *et al.*, 2012; 2014), side effects (Gottlieb *et al.*, 2012; Tatonetti *et al.*, 2012a), phenotypic similarity (Li *et al.*, 2015) and genomic similarity (Zhou et al., 2020b). However, these works rely heavily on the handcraft feature and domain knowledge. Recent deep learning-based methods can automatically learn abstract features with high robustness and generalization ability from a large volume of data, mitigating the limitations caused by traditional machine learning. However, previous works need large labeled data which may have false positive samples. And they typically either focused on the structure information or SMILES sequences (Toropov *et al.*, 2005) of the drugs without considering the rich semantic information related to drugs (Deac *et al.*, 2019; Huang *et al.*, 2020; Mohamed *et al.*, 2020; Ryu *et al.*, 2018), or utilized knowledge graph (KG) with rich bio-medical information without considering drug molecular structure information (Karim *et al.*, 2019; Lin *et al.*, 2020; Zitnik et al., 2018).

Although these methods have achieved strong performance, they do not consider the synergistic effect between the drug chemical structure and KG, thus limiting its predictive capability. Moreover, most state-of-art works consider is the presence of an interaction between drugs by considering the DDI prediction as a binary classification task while ignoring the significant research on specific types of adverse reactions between drugs. For example, KGNN (Lin *et al.*, 2020) determined the existence of an interaction between drugs, while in our model, we predicted the specific type of the interaction. For instance, we determined whether *aspirin* may decrease the excretion rate of *goserelin* and possibly increase the serum level.

Considering the above limitations, we propose a novel multi-scale feature fusion (MUFFIN) model, a deep learning framework for DDI prediction using drug chemical structure and bio-medical KG. We designed a bi-level cross strategy that can jointly learn the fusion representation of internal (chemical structure) and external (KG) features of the drugs from convolutional neural network (CNN)-based (Krizhevsky *et al.*, 2017) cross- and scalar-level perspectives. The bi-level architecture can effectively combine the multi-modal features through the multi-granularity feature fusion process, thus improving the capability of DDI prediction. Moreover, we evaluated the MUFFIN model on three different DDI prediction tasks, namely, binary-class, multi-class and multi-label tasks. Experimental results demonstrate that MUFFIN achieved the best performance on three tasks, thus supporting the significance of the combination of chemical structure and knowledge features from KG. The main contributions of this article can be summarized as follows:

- We propose a MUFFIN model, which is a novel deep learning-based feature fusion framework for binary-class, multi-class and multi-label DDI prediction. It can effectively integrate the features extracted from drug molecular structure and knowledge graph.
- We introduced a **bi-level** architecture, including cross- and scalar-level modules, which can fuse internal and external features from different granularity. (i) **Cross-level** is the extraction and aggregation of local features (based on the CNN) and global features by performing cross-product operation for different features; (ii) **Scalar-level** extract many fine-grained fusion features through the element-wise product.
- We compared MUFFIN with several state-of-art works and variants of our model for ablation study. Experimental results have shown that our work outperformed the baselines on three different DDI prediction tasks.

## 2 Related work

In recent years, many works have been proposed to resolve the DDI prediction problem by using the chemical structure of drugs. Vilar *et al.* (2012) combined the DDI and drug structural similarity matrices to generate a DDI interaction similarity matrix, and thus identifying DDI candidates. Cheng and Zhao (2014) utilized four similarities, including drug phenotypic, therapeutic, chemical structure and genomic properties, and combined with five machine learning-based models (Naive Bayes, decision tree, k-nearest neighbor, logistic regression and support vector machine) to deal with the DDI prediction task. Ryu *et al.* (2018) utilized the similarity of drug chemical structure as a feature and then fed the drug–drug pairs into the deep neural network (DNN) to predict the interaction type. Huang *et al.* (2020) developed an end-to-end model that generated a functional representation by using sub-structure information extracted from the drug SMILES string to predict DDIs.

Besides the methods we mentioned that are based on the chemical structure of the drug, some of the works utilized the topological information of drugs in bio-medical networks (Zeng *et al.*, 2019; Zhou *et al.*, 2020a). Zhang *et al.* (2017) presented an ensemble model that uses eight different types of drug data and a known DDI network to predict DDIs. Zitnik et al. (2018) developed a graph convolutional network (GCN) with drugs, targets and the side effects, and considered the DDI prediction as multi-relational link prediction task. Ma *et al.* (2018) designed a multi-view graph autoencoder with an attention mechanism to predict DDIs by considering the integration of known drug attributes, such as side effects, indications and interactions. Further, for large drug-related data in KG, Karim *et al.* (2019) integrated multi-source datasets to KG and then leveraged the ComplEx (Trouillon *et al.*, 2016), a KG embedding method, with convolutional LSTM network to predict DDIs. Lin *et al.* (2020) transferred DrugBank (Wishart *et al.*, 2018) and KEGG (Kanehisa and Goto, 2000) dataset into KG form through Bio2RDF (Belleau *et al.*, 2008) tool, and then utilized representation obtained by aggregating the neighbor information selectively to address the DDI prediction problem.

However, most of these works simply consider is the presence of an interaction between drugs without considering the positive effect of identifying specific side-effects between drugs with interaction. Meanwhile, these methods use structural information and knowledge information separately without considering their complementary role. Accordingly, we aim to design a new fusion strategy that can generate powerful drug representation for binary-class, multi-class and multi-label DDI prediction tasks by making full use of features extracted from drug molecular graph and large-scale bio-medical KG.

## 3 Materials and methods

Our problem formulation is summarized in Section 3.1. Section 3.2 introduces the framework of our proposed MUFFIN. Then, Section 3.3 describes the generation of drug representations. Section 3.4 develops a bi-level fusion strategy that leverages both structure and drug-related knowledge information to further learn drug representations. Section 3.5 then signifies how this representation can be leveraged to accurately predict DDI types.

### 3.1 Problem formulation

In our study, we present drug set as $D = \{d_1, d_2, \ldots, d_{N_d}\}$ and its corresponding molecular structure graph set as $G_{drug} = \{g_1, g_2, \ldots, g_{N_d}\}$, where $N_d$ is the total number of drugs. For the binary-class prediction task, we define a DDI matrix $Y$ with each element $y_{ij} \in \{0, 1\}$ indicates the existence of experimental evidence that $d_i$ and $d_j$ interact (i.e. $y_{ij} = 1$) or the absence of evidence for interaction (i.e. $y_{ij} = 0$). For the multi-class prediction task, we consider all types $R_D$ of DDI pairs (81 types of DDI relations are defined in our work). And for the multi-label task, 200 different DDI types are considered.

We present $G_{kg} = \{(h, r, t)|h, t \in E, r \in R\}$ as KG, where $E$ denotes entity set, and $R$ represents relation set in KG. Each triple $(h_i, r_i, t_i)$ describes is the presence of a connection between $h_i$ and $t_i$ with $r_i$ as the relation (e.g. Loxoprofen, drug-target and COX2), where $h_i, t_i \in E, r_i \in R, i \in \{1, 2, \ldots, N_{kg}\}$, and $N_{kg}$ denotes the total number of triples in our KG.

For the DDI prediction problem, given the $G_{kg}$ and DDI relation matrix $Y$ or DDI interaction pairs, we aim to learn a prediction function $\hat{y}_{ij} = \mathcal{F}_1((d_i, d_j)|\theta, G_{kg}, Y)$ and a mapping $\mathcal{F}_2 : D \times D \to R_D$ from drug pair $(d_i, d_j)$ to specific type for binary-class and multi-class, respectively. Note that $\theta$ denotes the model parameters, and $\hat{y}_{ij}$ denotes the probability of interaction between drug pair $(d_i, d_j)$.

### 3.2 Overview of MUFFIN

The framework of MUFFIN is illustrated inFigure 1. Our framework consists of three modules. In **representation learning module**, we adopted message passing neural network (MPNN) and a knowledge graph representation method (e.g. TransE) to extract the molecular structure feature and semantic features from molecular graph and knowledge graph, respectively. In **feature fusion module**, we designed a bi-level strategy, including cross- and scalar-level units. In cross-level unit, we crossed both features and then used CNN and flatten operation to learn the local and global features, respectively. In scalar-level unit, we utilized element-wise product to get fine-grained interactive feature between two different features. In **classifier module**, we concatenated the features learned from

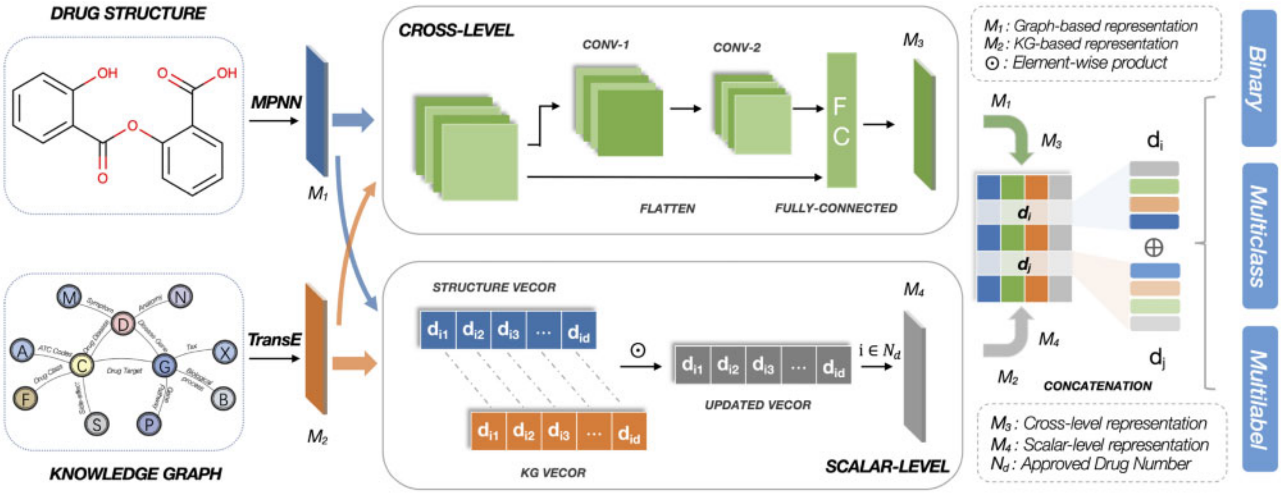


**Fig. 1.** MUFFIN workflow: (**a**) Representation Learning Module (left part): MUFFIN leverages the drug structure information (1-D SMILES converted to the 2-D molecular graph) and feeds them into MPNN to learn the graph-based structure representation $M_1$. For the entities in KG, MUFFIN uses the KGE method (e.g. TransE) to obtain the KG-based representation $M_2$. (**b**) Feature Fusion Module (middle part): the upper part implements the cross-level fusion strategy, both the representations are used as inputs of the cross-product operation, and then we utilize the CNN-based network and flatten operation to obtain the local and global features. The cross-level representation $M_3$ is received through a concatenation operation and a fully connected layer. The lower part is the scalar-level fusion strategy, which uses the element-wise product to obtain the scalar-level representation $M_4$. (**c**) Classifier Module (right part): the four-part representations are concatenated and then fed into the fully connected layer for binary-class, multi-class and multi-label DDI prediction

above modules, and then used various classifiers to predict DDIs according to different classification tasks. Next, we introduce the details of our framework.

## 3.3 Representation learning module

**Graph-based representation.** For each drug $d_i \in D$, we constructed a molecular graph $g_i \in G_{drug}$ according to its SMILES string, and $g_i = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ represent atoms, and $\mathcal{E}$ represent chemical bonds. We adopted message passing neural networks (Gilmer *et al.*, 2017) to generate the structure representation of $d_i$. The process involves a message-passing phase and a readout phase. In the message-passing phase, we performed k iterations and updated the representation of node $p$ by aggregating its neighbor information. Formally, the first phase can be described as follows:

$$z_p^k = U_{(k-1)}\left(z_p^{(k-1)}, \sum\nolimits_{q\epsilon N(p)} M_{(k-1)}\left(z_p^{(k-1)}, z_q^{(k-1)}, o_{pq}\right)\right) \tag{1}$$

where $U_{(k-1)}$ denotes node update functions, $M_{(k-1)}$ denotes message functions, $N(p)$ denotes the neighbors of $p$ in graph $g_i$, $o_{pq} \in R^d$ denotes the representation of edge between node $p$ and node $q$, and $z_p^{(k-1)} \in R^d$ denotes the representation of node $p$ after $k-1$ iterations. Note that the message function is defined as $M(z_p, z_q, o_{pq}) = \mathcal{W}(o_{pq})z_q$, where $\mathcal{W}(\cdot)$ denotes a neural network mapping the edge vector $o_{pq}$ to $d \times d$ matrix. The graph-based drug representation $z_i \in R^d$ can be acquired by the readout phase based on the node representation generated after $k$ iterations, and the second phase can be described as follows:

$$z_i = \frac{1}{\mathcal{V}_\rangle} \sum\nolimits_{p \in \mathcal{V}_\rangle} \left(z_p^k\right) \tag{2}$$

**KG-based representation.** For each entity and relation in $G_{kg}$, we employed TransE (Bordes *et al.*, 2013), a widely used method on KG embedding (KGE), to obtain the KG-based representation. Specifically, it learns entity and relation embeddings by optimizing the translation principle $\boldsymbol{e_h} + \boldsymbol{e_r} \approx \boldsymbol{e_t}$, where $\boldsymbol{e_h}, \boldsymbol{e_t}, \boldsymbol{e_r} \in R^d$, if a triplet $(h, r, t)$ exists in $G_{kg}$. The TransE model is trained to minimize the margin-based loss function which is described as follows:

$$L_{kg} = \sum\nolimits_{\varphi} max\left(0, f(e_h, e_r, e_t) + \gamma - f(e_{h'}, e_r, e_{t'})\right) \tag{3}$$

$$\varphi = (h, r, t) \in T \cup (h', r, t') \in T'_{(h,r,t)} \tag{4}$$

$$T'_{(h,r,t)} = \{(h', r, t)h' \in E\} \cup \{(h, r, t')t' \in E\} \tag{5}$$

where $\boldsymbol{e_h}$, $\boldsymbol{e_r}$ and $\boldsymbol{e_t}$ are the embeddings of the head entity, relation and the tail entity, respectively, $T$ and $T$' are the positive and negative sets of triples, and $\gamma$ denotes the margin parameter. For the negative sets of triplets $T$', it is formulated by replacing the entities or relations in positive triplets with entities and relations randomly sampled from $G_{kg}$.

## 3.4 Feature fusion module

We adopted a bi-level strategy to fuse graph- and KG-based representations. The fused features were used to express the interactive information on multi-faceted drug features. Before bi-level fusion operation, a fully connected layer was used to transfer the feature vectors into the same common space. Then, the vectors were denoted as $\boldsymbol{u}$ and $\boldsymbol{v}$ for $\boldsymbol{z}$ and $\boldsymbol{e}$, respectively. Specifically, $\boldsymbol{z}$ and $\boldsymbol{e}$ represent the graph- and KG-based representations of all drugs, as shown in Section 3.3. The process can be formulated as follows:

$$u = W_z z + b_z \tag{6}$$

$$v = W_e e + b_e \tag{7}$$

where $W_z$ and $W_e$ are trainable weights, and $b_z$ and $b_e$ are biases, respectively.

**Cross-level.** We denoted the graph- and KG-based $d_i \in D$ representations after transformation as vector $u_i = \{u_{i1}, u_{i2}, \ldots, u_{id}\}$ and $v_i = \{v_{i1}, v_{i2}, \ldots, v_{id}\}$, respectively. $u_i(v_i)$ denotes the $i$th row in vector $\boldsymbol{u}(\boldsymbol{v})$. We first constructed a cross matrix $C_i$ by cross-product operation, and this matrix expresses the interaction between $u_i$ and $v_i$ as follows:

$$C_i = u_i \otimes v_i = \begin{bmatrix} u_{i1}v_{i1} & u_{i1}v_{i2} & \cdots & u_{i1}v_{id} \\ \cdots & \cdots & \cdots\ \cdots & \cdots \\ u_{id}v_{i1} & u_{id}v_{i2} & \cdots & u_{id}v_{id} \end{bmatrix} \tag{8}$$

where $C_i \in R^{d \times d}$, $\otimes$ denotes the cross-product operator, and $d$ is the dimension of the drug vector. We then used the CNN model with a pooling layer to learn the local interactive feature and flatten the cross-matrix $C_i$ to learn the global feature. We concatenated both the local feature $a_{i_l}$ and global feature $a_{i_g}$ as $[a_{i_l} \parallel a_{i_g}]$, then fed them into a fully connected layer as a cross-level feature denoted as $a_i$. The specific process can be described as follows:

$$a_{i_l} = Pooling(ReLU(W_c * C_i + b_c)) \tag{9}$$

$$a_{i_g} = Flatten(C_i)\,(10) \tag{10}$$

$$a_i = MLP([a_{i_l} \parallel a_{i_g}]) \tag{11}$$

where $W_c$ denotes the filters, $*$ denotes the convolution operator, and $b_c$ denotes a bias vector. Here, we adopted *ReLU* and *max pooling* as non-linear activation and pooling functions, respectively. The pooling operation can reduce computational capacity and memory consumption and then improve the expression ability of the network. *MLP* is multilayer perceptron. And $\parallel$ denotes the concatenation operator that concatenates the local and global features.**Scalar-level.** We used element-wise product operation to encode the feature interaction between $u_i$ and $v_i$ learned from graph- and KG-based representations, and then fed the element-wise vector into a fully connected layer to obtain scalar-level fusion feature $s_i$. The process is described as follows:

$$s_i = MLP(u_i \odot v_i) = MLP(u_{i1}v_{i1}, u_{i2}v_{i2}, \ldots, u_{id}v_{id}) \tag{12}$$

where $s_i \in R^d$, $\odot$ denotes the element-wise product operator.

### 3.5 Classifier module

We concatenated the four-part drug representations as the representation of $d_i$, including the graph-based representation $u_i$, the KG-based representation $v_i$, the cross-level representation $a_i$, and the scalar-level representation $s_i$. The process can be described as follows:

$$x_i = [u_i \parallel a_i \parallel s_i \parallel v_i] \tag{13}$$

where $x_i \in R^{4d}$ denotes the final representation of $d_i$.

For DDI prediction tasks, we concatenated the representations of a pair of drugs and then fed them into fully connected layers to predict the DDI probability score as follows:

$$\hat{y}_{ij} = \sigma(MLP([x_i \parallel x_j])) \tag{14}$$

where $\hat{y}_{ij}$ represents the possibility of drug pair interaction in the binary-class prediction task, while in the multi-class and multi-label DDI prediction tasks, it represents the probability score of each relation type. Note that in binary-class and multi-label task, $\sigma$ is the *sigmoid* function, and for multi-class one, $\sigma$ refers to the *softmax* function.

### 3.6 Training

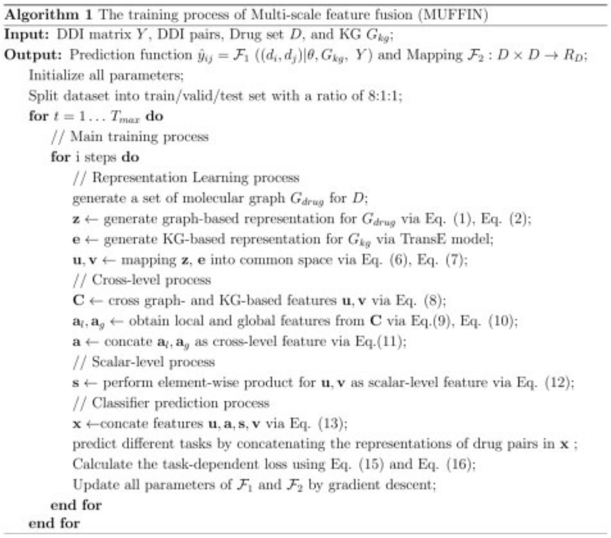
During training, we optimized the MUFFIN parameters by minimizing the cross-entropy loss in the binary-class and multi-label prediction tasks, as described below:

$$\ell_b = -\left[y_{ij} log \hat{y}_{ij} + (1 - y_{ij}) \log (1 - \hat{y}_{ij})\right] \tag{15}$$

where $y_{ij} \in \{0, 1\}$ denotes the interaction label for drug pair $(d_i, d_j)$ in binary-class task, and in multi-label task, each element of $y_{ij}$ is the one-hot vector with 200 elements (i.e. 200 DDI types). For the multi-class prediction task, the loss is defined as follows:

$$\ell_m = -\sum_{c=1}^{N_c} y_c \log \hat{y}_c \tag{16}$$

where $N_c$ is the number of multi-class DDI types, $y_c \in \{0, 1\}$ describes whether current type $c$ is the same as the true label of sample pair, and $\hat{y}_c$ indicates the probability that the observed sample $(d_i, d_j)$ belongs to type $c$. The entire training algorithm of MUFFIN is detailed in Algorithm 1. Note that in Algorithm 1, bold characters represent matrix representation of all drugs, such as $C = [C_1, \ldots, C_{i-1}, C_i, C_{i+1} \cdots, C_{N_d}]$, where $N_d$ is the total number of drugs.



```
Algorithm 1 The training process of Multi-scale feature fusion (MUFFIN)
Input: DDI matrix $Y$, DDI pairs, Drug set $D$, and KG $G_{kg}$;
Output: Prediction function $\hat{y}_{ij} = \mathcal{F}_1((d_i, d_j)|\theta, G_{kg}, Y)$ and Mapping $\mathcal{F}_2 : D \times D \rightarrow R_D$;
  Initialize all parameters;
  Split dataset into train/valid/test set with a ratio of 8:1:1;
  for $t = 1 \ldots T_{max}$ do
    // Main training process
    for i steps do
      // Representation Learning process
      generate a set of molecular graph $G_{drug}$ for $D$;
      $\mathbf{z} \leftarrow$ generate graph-based representation for $G_{drug}$ via Eq. (1), Eq. (2);
      $\mathbf{e} \leftarrow$ generate KG-based representation for $G_{kg}$ via TransE model;
      $\mathbf{u}, \mathbf{v} \leftarrow$ mapping $\mathbf{z}, \mathbf{e}$ into common space via Eq. (6), Eq. (7);
      // Cross-level process
      $\mathbf{C} \leftarrow$ cross graph- and KG-based features $\mathbf{u}, \mathbf{v}$ via Eq. (8);
      $\mathbf{a}_l, \mathbf{a}_g \leftarrow$ obtain local and global features from $\mathbf{C}$ via Eq.(9), Eq. (10);
      $\mathbf{a} \leftarrow$ concate $\mathbf{a}_l, \mathbf{a}_g$ as cross-level feature via Eq.(11);
      // Scalar-level process
      $\mathbf{s} \leftarrow$ perform element-wise product for $\mathbf{u}, \mathbf{v}$ as scalar-level feature via Eq. (12);
      // Classifier prediction process
      $\mathbf{x} \leftarrow$concate features $\mathbf{u}, \mathbf{a}, \mathbf{s}, \mathbf{v}$ via Eq. (13);
      predict different tasks by concatenating the representations of drug pairs in $\mathbf{x}$ ;
      Calculate the task-dependent loss using Eq. (15) and Eq. (16);
      Update all parameters of $\mathcal{F}_1$ and $\mathcal{F}_2$ by gradient descent;
    end for
  end for
```

## 4 Experiments

In this section, we design various experiments to evaluate the performance of our model.

### 4.1 Datasets description

In our experiments, we use various datasets to verify the effectiveness of our model.

- **Binary-class DDIs and KG Dataset.** DRKG (Ioannidis *et al.*, 2020), a comprehensive biological KG, includes 97 238 entities belonging to 13 entity-types and 5 874 261 triplets belonging to 107 edge-types. In our experiment, we extract binary-class DDI data from DRKG, in which the relation types are defined as <DRUGBANK::ddi-interactor-in::Compound: Compound> and <Hetionet::DrD>. The rest triples in DRKG were treated as KG dataset for the entire experiment.
- **Multi-class DDIs.** We collected the DDI multi-relation data from DeepDDI (Ryu *et al.*, 2018), which is extracted from DrugBank (Wishart *et al.*, 2018). It consists of 192 284 DDIs with 86 relation types. In line with the KG extracted from DRKG, we left a total of 172 426 DDI pairs with 81 relationships, which eliminates relationships with less than 10 samples.
- **Multi-label DDIs.** We used TWOSIDES (Tatonetti *et al.*, 2012b) as our multi-label DDI data source. To map drugs into the KG, we collected 99 002 DDIs with 200 different relations, ensuring each type has approximately 10 000 samples.

The details of our experiment datasets are shown in Table 2.

### 4.2 Baselines

To evaluate the effectiveness of our model, we compared MUFFIN with the following state-of-art works:

- **DeepWalk** (Perozzi *et al.*, 2014) generates a node sequence by using random walk and then used the word2vec (Mikolov *et al.*, 2013) method to learn the node representations. The representations of each node in drug pair were concatenated and then fed into the classifier for DDI prediction.
- **LINE** (Tang *et al.*, 2015) proposes a neural network-based method, which directly modeled the node embedding by considering both the first-order proximity (local structure) and second-order

proximity (global structure). Similar to DeepWalk, this method does not use any information about drugs, except for known DDIs.

- **DeepDDI** (Ryu *et al.*, 2018) designs a DNN framework with drug structural similarity as input to predict the interaction type between drugs. Moreover, the structural similarity was calculated using Tanimoto coefficient with extended-connectivity fingerprints of diameter 4 (ECFP4) (Rogers and Hahn, 2010). Considering that DeepDDI is originally used to predict the multi-class of drug pair, for the binary-class prediction task, we set the number of output neurons of DNN to 1. For the multi-class (multi-label) prediction task, the number of output neurons was set to 81 (200).
- **KGDDI** (Karim *et al.*, 2019) integrates multi-source datasets into KG, and utilizes the ComplEx embedding method to embed the nodes in KG, and then feeds them into a neural network based on CNN-LSTM to predict the relationship between drugs.
- **KGNN** (Lin *et al.*, 2020) adopts GCN to selectively aggregate the neighbor information with high-level layer information for learning the node representation in the KG.

To validate the effectiveness of each component in MUFFIN, we de-signed six different variants for the ablation study, and the details are described as follows:

- **MUFFIN_ST** simply uses MPNN to learn the drug representation based on structure feature and predicts the probability of each interaction types between given drug pairs through concatenating the drug embedding from drug pairs.
- **MUFFIN_KG** uses the KGE method (e.g. TransE) to learn the representation of each node in KG.
- **MUFFIN_cross** removes the scalar-level component on the basis of MUFFIN (our model) for DDI prediction.
- **MUFFIN_scalar** removes the cross-level component by comparison with MUFFIN.
- **MUFFIN_concat** simply considers the concatenation of the two parts of the feature learned from representation learning layer.
- **MUFFIN_sum** uses the sum operation to fuse two different features.

## 4.3 Evaluation metrics

We considered the metrics commonly used in binary-class and multi-class tasks, including Precision, Recall, F1, Accuracy, which are defined as follows:

- **Accuracy:** average accuracy over different classes can be defined as$Accuracy = (1/N_c)\sum_{i=1}^{N_c}(TP_i+TN_i)/(TP_i+FN_i+FP_i+TN_i)$, where $TP$, $TN$, $FP$ and $NP$ indicate the true positive, true negative, false positive and false negative, respectively.
- **Precision:** precision is the fraction of correct predicted interactions among all predicted interactions. The precision averaged over all classes is $Precision = (1/N_c)\sum_{i=1}^{N_c} TP_i/(TP_i + FP_i)$.
- **Recall:** recall is the ratio of the number of correctly predicted DDI to the total number of DDIs that exist. $Recall = (1/N_c)\sum_{i=1}^{N_c} TP_i/(TP_i + FN_i)$ is considered the averaged recall on different labels.
- **F1 score:** F1 is a trade-off between precision and recall. And the harmonic mean of average precision and recall score can be defined as $F1 = (2 \cdot Precision \cdot Recall)/(Precision + Recall)$.

Note that for the binary-class, $N_c$ is defined as 1. For the multi-label task, we use AUC and AUPR as performance metrics. AUC is the area under the receiver operating characteristic curve and AUPR denotes the area under the precision-recall curve.

## 4.4 Experiment setup

In this article, we represented KG entities, relations and drug structure embeddings as 100-dimensional vectors. For molecular graphs of drugs, we use the pretrained-GNN to obtain the graph-based representation (Hu *et al.*, 2019). And for the cross-level component, we constructed two CNN layers with pooling layers, and the filter and kernel sizes were set to eight and five, respectively. The hidden size of the last fully connected layers was set to 2048. And we set the number of output neurons according to the classification task (the binary classification task is 1, the multi-class is 81 and the multi-label is 200). The experimental learning rate was set to 0.0001. The model was trained for 200 epochs. The hyper-parameter settings of baselines were the same as reported in their original papers, and all experiments were based on five-fold cross-validation.

## 4.5 Evaluation results

The upper part of Table 1 shows the performance of MUFFIN and baselines we mentioned before on binary-class, multi-class and multi-label DDI tasks. In comparison with all baselines, MUFFIN

**Table 1.** Results of baselines, MUFFIN and the variants of MUFFIN

| Method | Binary-class | | | | | Multi-class | | | | Multi-label | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Accuracy | Precision | Recall | F1 | AUC | Mean-Accuracy | Macro-Precision | Macro-Recall | Macro-F1 | AUC | AUPR |
| DeepWalk | 0.8130 | 0.7970 | 0.8390 | 0.8170 | 0.8850 | 0.8000 | 0.8220 | 0.7101 | 0.7469 | 0.8708 | 0.6160 |
| LINE | 0.7810 | 0.7710 | 0.8000 | 0.7850 | 0.8620 | 0.7506 | 0.6870 | 0.5451 | 0.5804 | 0.8621 | 0.6043 |
| DeepDDI | 0.9166 | 0.9121 | 0.9214 | 0.9167 | 0.9748 | 0.8768 | 0.7986 | 0.7593 | 0.7662 | 0.8301 | 0.5032 |
| KGDDI | 0.8926 | 0.8936 | 0.8925 | 0.8925 | 0.9620 | 0.8923 | 0.7945 | 0.7667 | 0.7666 | 0.8906 | 0.6527 |
| KGNN | 0.9034 | 0.9058 | 0.8999 | 0.9029 | 0.9701 | 0.9127 | 0.8583 | 0.8170 | 0.8291 | 0.8948 | 0.6584 |
| MUFFIN_ST | 0.9166 | 0.9079 | 0.9329 | 0.9201 | 0.9772 | 0.9393 | 0.9257 | 0.9081 | 0.9113 | 0.8874 | 0.6353 |
| MUFFIN_KG | 0.9261 | 0.9175 | 0.9360 | 0.9265 | 0.9797 | 0.9298 | 0.9023 | 0.8810 | 0.8849 | 0.9054 | 0.6796 |
| MUFFIN_scalar | 0.9657 | 0.9642 | 0.9670 | 0.9656 | 0.9947 | 0.9515 | 0.9417 | 0.9268 | 0.9300 | 0.9114 | 0.6935 |
| MUFFIN_cross | 0.9890 | 0.9894 | 0.9884 | 0.9889 | 0.9991 | 0.9621 | 0.9493 | 0.9396 | 0.9411 | 0.9153 | 0.6982 |
| MUFFIN_concat | 0.9529 | 0.9473 | 0.9587 | 0.9530 | 0.9908 | 0.9465 | 0.9357 | 0.9188 | 0.9227 | 0.9109 | 0.6947 |
| MUFFIN_sum | 0.9509 | 0.9522 | 0.9490 | 0.9506 | 0.9900 | 0.9503 | 0.9381 | 0.9242 | 0.9269 | 0.9108 | 0.6941 |
| **MUFFIN (ours)** | **0.9913** | **0.9912** | **0.9913** | **0.9912** | **0.9994** | **0.9648** | **0.9568** | **0.9482** | **0.9495** | **0.9160** | **0.7033** |

*Note*: We adopted DRKG, DrugBank and TWOSIDES as the datasets of binary-class, multi-class and multi-label tasks, respectively. Bold numbers signify the best performance for each metric column.

showed the best performance. Moreover, for binary-class and multi-class tasks, the performance of MUFFIN was improved by at least 7.47% on accuracy, 7.91% on precision, 6.99% on recall, 7.45% on F1, 2.46% on AUC, 5.21% on mean accuracy, 9.85% on macro-precision, 13.12% on macro-recall and 12.04% on macro-F1. For multi-label prediction task, MUFFIN outperformed the best baseline by up to 2.12% on AUC and 4.49% on AUPR. These findings demonstrate the effectiveness of our model.

Specifically, (i) DeepWalk and LINE showed the worst results com-pared with other baselines, because they did not consider any drug information except for known DDIs; (ii) DeepDDI showed a relatively low performance than that of our model, because it merely utilized structure similarity information as a feature; and (iii) KGDDI and KGNN models, which have relatively poor results, considered the rich semantic information of drugs and the high-order relationship of nodes without leveraging the auxiliary features based on the molecular structure graph. Compared with all baselines, our model considers the multi-modal data, including KG and molecular structure graph and shows excellent performance in all three different tasks.

## 4.6 Ablation study

We further compared MUFFIN with its six variants, and the results are shown in the lower part of Table 1. MUFFIN_KG and MUFFIN_ST have poor performance compared with the other variants that combine the features extracted from the molecular structure graph and bio-medical KG. This finding occurred mainly because they used single drug information as a feature to predict DDIs. The performance of MUFFIN_scalar and MUFFIN_cross was lower compared with that of MUFFIN, thus supporting the effectiveness of bi-level fusion strategy that combines cross- and scalar-level modules. To verify the performance of our model relative to the traditional fusion strategy, including concatenation and summation, we made minor modifications based on the MUFFIN model. Table 2 shows that MUFFIN outperforms MUFFIN_concat and MUFFIN_sum in terms of all criteria on three different prediction tasks, thus further supporting the effectiveness of the proposed fusion strategy.

**Table 2.** Statistics of the datasets

| Data Name | Number | Data name | Number |
|---|---|---|---|
| Approved Drugs | 2322 | DDI (binary-class) | 1 178 210 |
| KG Entities | 96 766 | DDI (multi-class) | 172 426 |
| KG Relations | 105 | DDI (multi-label) | 99 002 |
| KG Triples | 4 488 504 | – | – |

In summary, our MUFFIN model has the best performance compared with all competing methods and variants, which completely signifies that the combination of drug structure and KG features is important in all prediction tasks, and the proposed fusion strategy is beneficial to downstream DDI prediction tasks.

## 4.7 Parameter analysis

In this section, we investigate the effect of different key parameters settings on MUFFIN by fixing other parameters. Figure 2 shows the performance of different negative samples in KG and embedding dimension in terms of different metrics. To study the influence of different parameters on the same dataset, we showed the performance of binary-class and multi-class prediction tasks.

**Impact of the negative sample size.** Figure 2a and b shows the influence of different number of negative samples for each positive sample during KG training. Compared with 32, 64, 256 sample size, we found that in two different tasks, 128 is the relatively best choice in our model. We observed that MUFFIN can learn more useful information with enough negative samples, however, with the increase

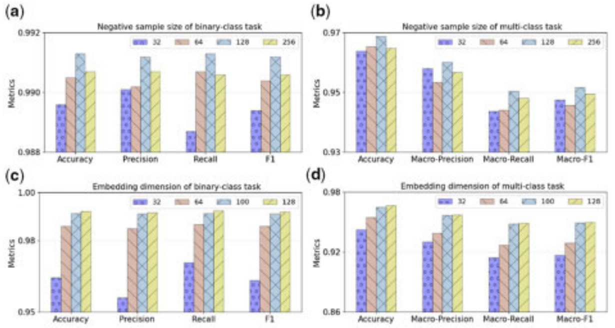


**Fig. 2.** Model performance under different negative sample size and embedding dimension settings.

**Table 3.** Top 10 drugs that may interact with Quinidine

| Drug name | Interaction type | Evidence |
|---|---|---|
| Bretylium | Bretylium may increase the hypotensive activities of Quinidine. | PMID:<br>386786 |
| Potassium bicarbonate | Potassium bicarbonate may decrease the excretion rate of Quinidine which could result in a higher serum level. | PMID:<br>5360313<br>8739822 |
| Sulfasalazine | Sulfasalazine may decrease the excretion rate of Quinidine which could result in a higher serum level. | DrugBank |
| Nebivolol | Nebivolol may increase the hypotensive activities of Quinidine. | Drugs.com |
| Eliglustat | The metabolism of Eliglustat can be decreased when combined with Quinidine. | Drugs.com<br>PMID:<br>31849452 |
| Romidepsin | The risk or severity of QTc prolongation can be increased when Romidepsin is combined with Quinidine. | PMID:<br>32449208 |
| Neratinib | Neratinib may decrease the excretion rate of Quinidine which could result in a higher serum level. | Unconfirmed |
| Potassium | Potassium may decrease the excretion rate of Quinidine which could result in a higher serum level. | PMID:<br>8930193<br>6025406 |
| Vernakalant | The risk or severity of QTc prolongation can be increased when Vernakalant is combined with Quinidine. | Unconfirmed |
| Iothalamic acid | Iothalamic acid may decrease the excretion rate of Quinidine which could result in a higher serum level. | DrugBank |

in the number of negative samples, more noise is introduced in the KG representation learning process.

**Impact of embedding dimension.** Figure 2c and d shows the results of different embedding dimensions. We investigated the influence of different embedding dimension by varying it from 32 to 128. We observed that with the increase of embedding dimension, the performance of the model is improved, but the improvement of performance is gradually narrowed. Thus, we selected 100-dimension as our embedding size in our experiment.

### 4.8 Case study: quinidine

In this section, we conduct a case study to demonstrate the strong performance of MUFFIN. We analyzed the predicted results in detail, and mainly focused on the top 10 drug pairs for Quinidine in Table 3. For these 10 drug pairs, we apply PubMed, DrugBank and Drug Interactions Checker tool provided by Drugs.com, to find the evidence support for them. From Table 3, we can observe that 8 of 10 DDI pairs have evidence, which further indicates the effectiveness of MUFFIN. For example, we predicted that the metabolism of *eliglustat* can be decreased when combined with *quinidine*, which is reported in the literature (Wang *et al.*, 2019).

## 5 Conclusion

In this article, we propose MUFFIN, a new computational framework with a novel multi-scale fusion strategy for binary-class, multi-class and multi-label DDI prediction tasks. It fully exploits the features extracted from both drug molecular structure graph and biomedical knowledge graph DRKG. According to the ablation study, we demonstrated that the combination of features is necessary. Moreover, the proposed bi-level strategy can effectively combine the multi-modal features. Compared with several state-of-art works on three different DDI prediction tasks, the experimental results on three real-world datasets showed the effectiveness of our model.

In our work, we mainly focus on the feature fusion strategy of drugs. However, data redundancy still exists in DRKG, in future work, we will use entity alignment technique to improve the quality and availability of KG. Moreover, we can provide interpretability for the results predicted by the model on such KG.

## Funding

The work was supported in part by National Natural Science Foundation of China [61872309, 61972138], in part by the Fundamental Research Funds for the Central Universities [531118010355] and in part by Hunan Provincial Natural Science Foundation of China [2020JJ4215].

*Conflict of Interest*: The authors declare that there is no conflict of interest.

## Data availability

All data used in the study are from public resources. DrugBank is available at https://go.drugbank.com/releases/5-1-8/downloads/all-full-database; TWOSIDES is available at http://tatonettilab.org/resources/nsides/; DRKG is available at https://github.com/gnn4dr/DRKG/. Data processing scripts are also provided in MUFFIN GitHub code repository, which is https://github.com/xzenglab/MUFFIN.

## Reference

Belleau,F. *et al.* (2008) Bio2RDF: towards a mashup to build bioinformatics knowledge systems. *J. Biomed. Inform.*, **41**, 706–716.

Bordes,A. *et al.* (2013) Translating embeddings for modeling multi-relational data. *Adv. Neural Inf. Process. Syst.*, 2787–2795.

Cheng,F. and Zhao,Z. (2014) Machine learning-based prediction of drug--drug interactions by integrating drug phenotypic, therapeutic, chemical, and genomic properties. *J. Am. Med. Inform. Assoc.*, **21**, e2e278–e2e286. (

Deac,A. *et al.* (2019) Drug–drug adverse effect prediction with graph co-attention.

Giacomini,K.M. *et al.* (2007) When good drugs go bad. *Nature*, **446**, 975–977.

Gilmer,J. *et al.* (2017) Neural message passing for quantum chemistry. *International Conference on Machine Learning*, pp. 1263–1272.

Gottlieb,A. *et al.* (2012) INDI: a computational framework for inferring drug interactions and their associated recommendations. *Nat. Rev. Drug Discov.*, **8**, 592.

Hu,W. *et al.* (2019) Strategies for pre-training graph neural networks.

Huang,K. *et al.* (2020) CASTER: predicting Drug Interactions with Chemical Substructure Representation. *Proc. AAAI Conf. Artif. Intell.*, **34**, 702–709.

Ioannidis,V.N. *et al.* (2020) Drkg-drug repurposing knowledge graph for covid-19. https://github.com/gnn4dr/DRKG/.

Kanehisa,M. and Goto,SJNar. (2000) KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Res.*, **28**, 27–30.

Karim,M.R. *et al.* (2019) Drug–drug interaction prediction based on knowledge graph embeddings and convolutional-LSTM network. In *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*. Niagara Falls, NY, pp. 113–123.

Krizhevsky,A. *et al.* (2017) Imagenet classification with deep convolutional. *Neural Netw.*, **60**, 84–90.

Li,P. *et al.* (2015) Large-scale exploration and analysis of drug combinations. *Bioinformatics*, **31**, 2007–2016.

Lin,X. *et al.* (2020) KGNN: Knowledge graph neural network for drug–drug interaction prediction. In *IJCAI*. Yokohama, Japan.

Ma,T. *et al.* (2018) Drug similarity integration through attentive multi-view graph auto-encoders.

Mikolov,T. *et al.* (2013) Efficient estimation of word representations in vector space.

Mohamed,S.K. *et al.* (2020) Discovering protein drug targets using knowledge graph embeddings. *Bioinformatics*, **36**, 603–610.

Perozzi,B. *et al.* (2014) Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York City, NY, USA. pp. 701–710.

Plumpton,C.O. *et al.* (2016) A systematic review of economic evaluations of pharmacogenetic testing for prevention of adverse drug reactions. *Pharmacoeconomics*, **34**, 771–793.

Rogers,D. and Hahn,M.J.J. (2010) Extended-connectivity fingerprints, *J. Chem. Inf. Model.*, **50**, 742–754.

Ryu,J.Y. *et al.* (2018) Deep learning improves prediction of drug–drug and drug–food interactions. *Proc. Natl. Acad. Sci. USA*, **115**, E4304–E4311.

Takeda,T. *et al.* (2017) Predicting drug–drug interactions through drug structural similarities and interaction networks incorporating pharmacokinetics and pharmacodynamics knowledge, *J. Cheminform.*, **9**, 1–9.

Tang,J. *et al.* (2015) Line: large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web*. Florence, Italy. pp. 1067–1077.

Tatonetti,N.P. *et al.* (2012a) A novel signal detection algorithm for identifying hidden drug–drug interactions in adverse event reports. *J Am Med Inform Assoc.*, **19**, 79–85.

Tatonetti,N.P. *et al.* (2012b) Data-driven prediction of drug effects and interactions. *Sci. Transl. Med.*, **4**, 125ra131–125ra131.

Toropov,A.A. *et al.* (2005) Simplified molecular input line entry system (SMILES) as an alternative for constructing quantitative structure-property relationships (QSPR). *Indian J. Chem.*, **44**, 1545–1552.

Trouillon,T. *et al.* (2016) Complex embeddings for simple link prediction. In: *International Conference on Machine Learning (ICML)*. New York City, NY, USA.

Vilar,S. *et al.* (2012) Drug–drug interaction through molecular structure similarity analysis. *J. Am. Med. Informatics Assoc.*, **19**, 1066–1074.

Vilar,S. *et al.* (2014) Similarity-based modeling in large-scale prediction of drug–drug interactions. *Nat. Protoc.*, **9**, 2147–2163.

Wang,Q. *et al.* (2019) Drug–drug interactions of amiodarone and quinidine on the pharmacokinetics of eliglustat in rats. *Drug Des. Dev. Ther.*, **13**, 4207–4213.

Whitebread,S. *et al.* (2005) Keynote review: in vitro safety pharmacology profiling: an essential tool for successful drug development. *Drug Disc. Today*, **10**, 1421–1433.

Wishart,D.S. *et al.* (2018) DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic Acids Res.*, **46**, D1074–D1082.

Zeng,X. *et al.* (2019) deepDR: a network-based deep learning approach to in silico drug repositioning. *Bioinformatics*, **35**, 5191–5198.

Zhang,W. *et al.* (2017) Predicting potential drug-drug interactions by integrating chemical, biological, phenotypic and network data. *BMC Bioinformatics*, **18**, 18.

Zhou,Y. *et al.* (2020a) Network-based drug repurposing for novel coronavirus 2019-nCoV/SARS-CoV-2. *Cell Discov.*, **6**, 1–18.

Zhou,Y. *et al.* (2020b) Artificial intelligence in COVID-19 drug repurposing. *The Lancet. Digital health*. **2**, e667–e676.

Zitnik,M. *et al.* (2018) Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics*, **34**, i457–i466.