MDF-SA-DDI: Predicting drug-drug interaction events based on multi-source drug fusion, multi-source feature fusion and transformer self-attention mechanism

Article *in* Briefings in Bioinformatics · October 2021

DOI: 10.1093/bib/bbab421

Briefings in Bioinformatics, 00(00), 2021,1–13

https://doi.org/10.1093/bib/bbab421
Problem Solving Protocol

MDF-SA-DDI: predicting drug–drug interaction events based on multi-source drug fusion, multi-source feature fusion and transformer self-attention mechanism

Shenggeng Lin[†], Yanjing Wang[†], Lingfeng Zhang[†], Yanyi Chu, Yatong Liu, Yitian Fang, Mingming Jiang, Qiankun Wang, Bowen Zhao, Yi Xiong and Dong-Qing Wei

Corresponding authors: Yi Xiong, State Key Laboratory of Microbial Metabolism, Shanghai-Islamabad-Belgrade Joint Innovation Center on Antibacterial Resistances, Joint International Research Laboratory of Metabolic & Developmental Sciences and School of Life Sciences and Biotechnology, Shanghai Jiao Tong University, Shanghai 200030, P.R. China. Tel.: +86-21-34204573. E-mail: xiongyi@sjtu.edu.cn; Dong-Qing Wei, Peng Cheng Laboratory, Vanke Cloud City Phase I Building 8, Xili Street, Nanshan District, Shenzhen, Guangdong 518055, P.R. China. E-mail: dqwei@sjtu.edu.cn
[†]These authors contributed equally to this work.

Abstract

One of the main problems with the joint use of multiple drugs is that it may cause adverse drug interactions and side effects that damage the body. Therefore, it is important to predict potential drug interactions. However, most of the available prediction methods can only predict whether two drugs interact or not, whereas few methods can predict interaction events between two drugs. Accurately predicting interaction events of two drugs is more useful for researchers to study the mechanism of the interaction of two drugs. In the present study, we propose a novel method, MDF-SA-DDI, which predicts

Shenggeng Lin is a master student at the School of Life Sciences and Biotechnology, Shanghai Jiao Tong University. He works on drug discovery through deep learning methods.
Yanjing Wang is a postdoctoral scholar at the School of Life Sciences and Biotechnology, Shanghai Jiao Tong University. Her research interests focus on drug discovery through machine learning methods, molecular dynamics simulations and cancer genomics.
Lingfeng Zhang is a Master's student of the School of Electrical Engineering and Computer Science, University of Ottawa, Canada. His research interests include deep learning, computer vision and data science.
Yanyi Chu is a Ph.D. candidate at the School of Life Sciences and Biotechnology, Shanghai Jiao Tong University. She works on drug discovery through machine learning methods.
Yatong Liu is a master student at the School of Life Sciences and Biotechnology, Shanghai Jiao Tong University. She works on disease prediction though multi-label machine learning methods.
Yitian Fang is a PhD student at the School of Life Sciences and Biotechnology, Shanghai Jiao Tong University. She works on drug discovery through machine learning methods and molecular simulation.
Mingming Jiang is a master student at the School of Life Sciences and Biotechnology, Shanghai Jiao Tong University. He works on machine learning and natural language processing in Bioinformatics.
Qiankun Wang is a Ph.D. candidate at the School of Life Sciences and Biotechnology, Shanghai Jiao Tong University. He has expertise in computer-aided drug design and machine learning.
Bowen Zhao is a master candidate at the School of Life Sciences and Biotechnology, Shanghai Jiao Tong University. His research interests include molecular representation learning and machine learning.
Yi Xiong is an associate professor at the School of Life Sciences and Biotechnology, Shanghai Jiao Tong University. His main research interests focus on machine learning and bioinformatics.
Dong-Qing Wei is a full professor at the School of Life Sciences and Biotechnology, Shanghai Jiao Tong University. He made many groundbreaking contributions to the development of bioinformatics techniques and their interdisciplinary applications to systems of ever-increasing complexity.
Submitted: 4 August 2021; **Received (in revised form):** 1 September 2021

drug–drug interaction (DDI) events based on multi-source drug fusion, multi-source feature fusion and transformer self-attention mechanism. MDF-SA-DDI is mainly composed of two parts: multi-source drug fusion and multi-source feature fusion. First, we combine two drugs in four different ways and input the combined drug feature representation into four different drug fusion networks (Siamese network, convolutional neural network and two auto-encoders) to obtain the latent feature vectors of the drug pairs, in which the two auto-encoders have the same structure, and their main difference is the number of neurons in the input layer of the two auto-encoders. Then, we use transformer blocks that include self-attention mechanism to perform latent feature fusion. We conducted experiments on three different tasks with two datasets. On the small dataset, the area under the precision–recall-curve (AUPR) and F1 scores of our method on task 1 reached 0.9737 and 0.8878, respectively, which were better than the state-of-the-art method. On the large dataset, the AUPR and F1 scores of our method on task 1 reached 0.9773 and 0.9117, respectively. In task 2 and task 3 of two datasets, our method also achieved the same or better performance as the state-of-the-art method. More importantly, the case studies on five DDI events are conducted and achieved satisfactory performance. The source codes and data are available at https://github.com/ShenggengLin/MDF-SA-DDI.

Key words: drug–drug interaction; multi-source drug fusion; multi-source feature fusion; self-attention mechanism

Introduction

Most human diseases are caused by complex biological processes and are resistant to the activity of any single drug [1]. Combination drug therapy is becoming a promising approach because it can improve drug efficacy and reduce drug resistance [2]. However, some drugs may interact with other drugs when they are taken together, and unexpected drug–drug interactions (DDIs) happened, which may lead to adverse drug events [3]. In order to avoid such events, it is highly desirable to identify potential DDIs. In addition, antibody-dependent enhancements caused by critical DDIs have led to the withdrawal of drugs from the market. Therefore, accurate prediction of DDIs is important for safer and improved prescription to patients [2].

Vitro experiments and clinical trials can be performed to identify DDIs, but systematic combinatorial screening of potential DDI remains challenging and expensive [1]. The computational methods are widely developed for prediction of DDIs due to their advantages such as high efficiency. These methods are roughly classified into six categories: (1) machine learning-based; (2) deep learning-based; (3) matrix factorization-based; (4) network diffusion-based; (5) ensemble learning-based methods and (6) literature-based or text mining methods.

In the last decade, a large number of machine learning-based models have been developed for prediction of DDIs [4–10]. For example, Kastrin *et al.* [7] considered topological and semantic features similarities, and used five classifiers to predict drug–drug interactions. Qian *et al.* [5] used feature similarity and feature selection methods to build a gradient boosting classifier to speed up the process and achieve robust prediction performance. Gottlieb *et al.* [8] calculated seven types of similarities and combined the two best similarities of each drug pair to generate a feature. Cami *et al.* [9] used standard multivariate methods to combine multiple predictors to make DDI predictions. The authors constructed a DDI network and obtained multiple covariates from the network to construct a logistic regression model and generalize linear mixed models. Cheng *et al.* [10] extracted features from simplified molecular-input line-entry system data and side effect similarities of drug pairs, and applied support vector machines (SVMs) to predict DDIs.

Deep learning-based models mainly include deep neural networks (DNN)-based models, graph embedding-based models and knowledge graph embedding-based models [1, 11–23]. Rohani *et al.* [23] calculated multiple drug similarities and Gaussian interaction curves of drug pairs, and applied the method to select the most informative and less redundant similarities as features. Then, the feature vectors of drug pairs were taken as input of the neural network for prediction. Zitnick *et al.* [1] proposed a method for predicting the side effects of drugs. This method regards DDI prediction as a multi-relational link prediction problem on a multi-modal graph, including the relationship between drugs, proteins and side effects. The authors used a graph convolutional network (GCN) as an encoder to generate embeddings for nodes on the graph, and a tensor decomposition model as a decoder to predict DDI. In addition, this work extended GCN to graphs with multiple node types and multiple edge types.

The DDI prediction task can be represented as a matrix completion task, which aims to predict unobserved interactions. Typical matrix factorization methods include non-negative matrix factorization, singular value decomposition and so on. Besides, some methods develop novel matrix factorization models based on manifold learning algorithms, artificial neural networks and so on [3, 24–31]. Zhu *et al.* [24] designed a dependent network to model the drug dependency and propose an attribute supervised learning model probabilistic dependent matrix tri-factorization (PDMTF) for ADDI prediction. Yu *et al.* [30] developed a novel method called DDINMF, which is based on the semi-nonnegative matrix factorization. Zhang *et al.* [31] proposed a manifold regularized matrix factorization method for DDI prediction.

The network diffusion-based methods can infer novel DDIs through a constructed network [3, 32–40]. Zhang *et al.* [37] constructed a network based on the structural and side effect similarities of drugs, and applied a label propagation algorithm to identify DDIs [37]. Park *et al.* [38] applied the random walk with restart on the protein–protein network to predict DDIs. Sridhar *et al.* [39] proposed a probabilistic approach to infer DDIs from the network, which is constructed based on multiple drug–drug similarities and known interactions [39]. Nyamabo *et al.* [40] proposed a deep learning framework, which performs the DDI prediction task between two drugs by identifying pairwise interactions between their respective substructures.

The ensemble learning-based methods reasonably combine several models to achieve better performance than individual models [3, 41–43]. Cheng *et al.* [10] proposed a heterogeneous network-assisted inference framework to assist with the prediction of DDIs, which applied five predictive models: naive Bayes, decision tree, k-nearest neighbor, logistic regression and SVM. Deepika and Geetha [42] adopted a semi-supervised learning framework with network representation learning and meta-learning from four drug datasets to predict DDIs. Chen *et al*. [43] proposed a multi-scale feature fusion deep learning model named MUFFIN, which can learn the drug representation from the drug structure and the knowledge graph with rich bio-medical information.

The literature-based or text mining methods can integrate multiple information sources, use domain knowledge and clinical evidence, and improve the computational model with more external knowledge, thereby improving the prediction performance and interpretability [6, 44–49]. Liu *et al*. [44] proposed a machine learning framework to extract useful features from the FDA adverse event reports and then identify potential high-priority DDIs using an autoencoder-based semi-supervised learning algorithm. Asada *et al*. [45] proposed a novel method to effectively utilize external drug database information as well as information from large-scale plain text for DDI extraction. They focused on drug description and molecular structure information as the drug database information. Zhao *et al*. [49] presented a syntax convolutional neural network (CNN) based DDI extraction method and obtained better performance than other state-of-the-art methods.

Generally, most state-of-the-art works mentioned above only predict whether there exists a DDI or not between a pair of drugs, which is formulated as a binary classification task. However, these methods do not provide sufficient details on DDIs in terms of pharmacological effects, which could suggest potential causal mechanisms on the DDI occurrence of drug pairs [50]. In this study, DDIs are subsequently represented by various events, and predicting these events can be regarded as a multi-class classification task. Predicting DDI-associated events is a meaningful but challenging task, and has received increasing interests recently [3, 43]. Ryu *et al*. [50] proposed a computational framework DeepDDI that uses names of drug–drug pairs and their structural information as inputs to accurately generate 86 important DDI types as outputs that are human-readable sentences. Lee *et al*. [2] used auto-encoders and a deep feed-forward network that are trained by using the structural similarity profiles, gene ontology term similarity profiles, and target gene similarity profiles of known drug pairs to predict pharmacological effects of DDIs. Deng *et al*. [3] proposed a multimodal deep learning framework named DDIMDL that combines diverse drug features for predicting DDI-associated events.

In addition to the models mentioned above, some machine learning methods that can be used for multi-classification tasks, such as k-nearest neighbor (KNN) [51], naive Bayes (NB) [52], SVM [51] and logistic regression (LR) [53], which can be used to predict DDI events. KNN is a simple and effective classifier. NB has a solid theoretical foundation for multi-classification. SVM generally perform well in small sample datasets, and LR has good interpretability. Some ensemble learning models can also be used to predict DDI events, such as Random Forest (RF) [51], Gradient Boosting Decision Tree (GBDT) [54] and eXtreme Gradient Boosting(XGBoost) [55]. These ensemble learning classifier combines multiple weak decision trees to obtain better and more comprehensive strong classifier. Some of models mentioned above are used as baseline methods in the paper.

It should be emphasized that there are multi-classification and multi-label classification in machine learning [56]. Multi-classification means that there is only one label to be predicted, but the value of the label may have multiple situations, that is, there are k possible categories for each sample ($k \geq 3$). But in a multi-label classification task, a sample may have multiple labels. Each label may have two or more categories. Common multi-label classification algorithms include ML-KNN [57], Rank-SVM [58], NLSP [59] and so on. Predicting DDI-associated events is a multi-class classification task.

Although methods mentioned above achieve satisfactory results, they still have some limitations. Firstly, most of these models based on deep learning techniques concatenate two drug vectors together to predict DDIs or DDI events without trying other ways to fuse the information of drug pairs, such as summation or latent feature fusion via Siamese network. Secondly, most of methods have performed well in predicting unobserved interactions between known drugs. However, they are hard to predict unobserved interactions between new drugs.

To overcome these limitations, we propose a novel method, MDF-SA-DDI, which predicts DDI events based on multi-source drug fusion, multi-source feature fusion and transformer self-attention mechanism. MDF-SA-DDI (Figure 1) is mainly composed of two parts: multi-source drug fusion and multi-source feature fusion, specifically, multi-source features are fused via transformer self-attention mechanism. First, we combine two drugs in four different ways and input the combined drug feature representation into four different drug fusion networks to obtain the latent feature vectors of the drug pairs. Specifically, we present a drug as a k-dimensional vector. (i) The first combination method is to input features of two drugs into a Siamese network (the first drug fusion network, SN), which is two auto-encoders that can share parameters. Two latent vectors of the Siamese neural network are used as new features of a drug pair. (ii) The second combination method is to combine two drug features into a 2*k-dimensional feature vector, and then input the 2*k-dimensional feature vector into the CNN (the second drug fusion network, CN). The output of CNN is used as the latent vector of the drug pair. (iii) The third combination method is to concatenate the feature vectors of two drugs to obtain a 1*2 k-dimensional feature vector, and then input the 1*2 k-dimensional feature vector to a larger single auto-encoder (the third drug fusion network, AE1). The latent vector of the auto-encoder is used as the new feature of the drug pair. (iv) The fourth combination method is to element-wise add the feature vectors of two drugs to obtain a 1*k-dimensional feature vector, and then input the 1*k-dimensional feature vector to the single auto-encoder (the fourth drug fusion network, AE2). The latent vector of the auto-encoder is used as the new feature of the drug pair. Then, we use several transformer blocks which include self-attention mechanism to perform feature fusion, which is the fusion of new features from different drug pairs. Finally, the obtained fusion features are fed into the fully connected layer to predict drug interaction events.

The experimental results show that our proposed MDF-SA-DDI method achieves better performance than several classic machine learning algorithms and state-of-the-art methods on all three tasks in two different datasets in the DDI events prediction problem. In addition, this study also proved the effectiveness of multi-source drug fusion. More importantly, we also conducted case studies, which prove the effectiveness of our method in practice.

Materials and Methods

Datasets

In this study, we use two datasets. The first data set (Dataset1) is the public dataset that Deng *et al*. [3] collected from DrugBank. Dataset1 contains 572 drugs with 74 528 pairwise DDIs, which are associated with 65 types of events. Each drug has four types of features: chemical substructures, targets, pathways

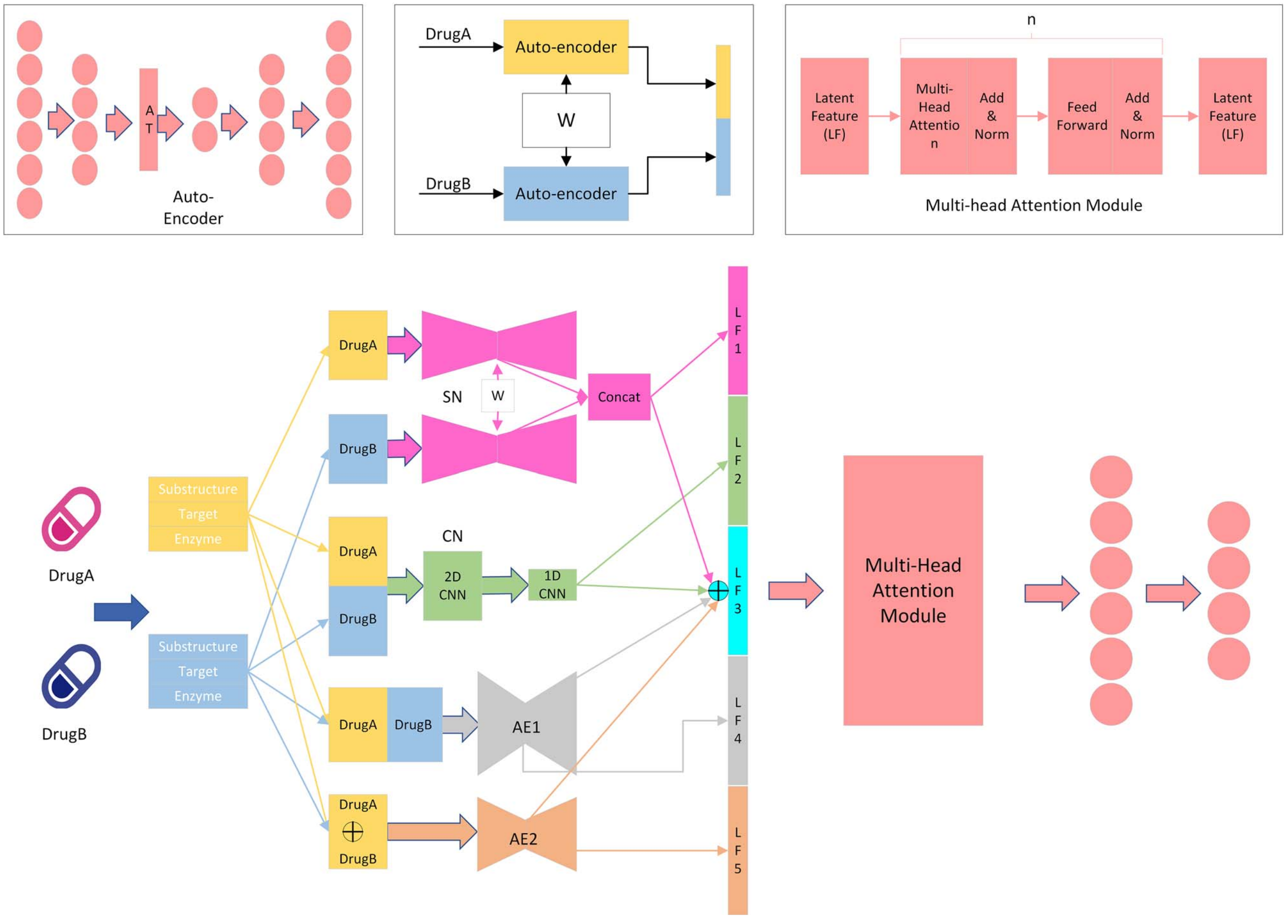

Figure 1. The workflow of the proposed MDF-SA-DDI method.

Table 1. Summary of the Dataset1' information and Dataset2's information

Dataset	Drug number	DDI number	DDI events number
Dataset1	572	74,528	65
Dataset2	1258	323,539	100

and enzymes. According to Deng *et al.*'s [3] experiment, the combination of substructures, targets and enzymes performs best in all combinations of features. Therefore, we construct the second dataset by collecting DDIs and drugs with three features: chemical substructures, targets and enzymes from DrugBank. We remove the rare events and select the events which have more than fifty DDIs. Thus, we obtain 1258 drugs with 323 539 pairwise DDIs, which are associated with 100 types of events (Dataset2). The dataset collection and processing methods have been described in detail in previous studies [3]. The detailed information of the two datasets is shown in Table 1.

The number of drugs in Dataset2 is more than twice the number of drugs in Dataset1. The number of DDI events in Dataset2 is more than four times that in Dataset1. And the types of DDI events in Dataset2 are also richer than those in Dataset1. Therefore, Dataset2 contains more DDI event information than Dataset1. More data in Dataset2 can help the model train better, but more DDI event types in Dataset2 also make it more difficult for the model to fit the data. In general, Dataset1 can measure the performance of the model on a small dataset, and Dataset2 can measure the performance of the model on a large and diverse dataset.

Drug feature extraction

Feature extraction and representation are essential for model construction. According to the previous study [3], the combination of chemical substructures, targets, and enzymes can achieve the best performance. Therefore, the MDF-SA-DDI model in the following experiments is constructed with three features: substructures, targets and enzymes. Each feature corresponds to a set of descriptors, so one drug can be represented by a binary feature vector, and its value (1 or 0) indicates the presence or absence of the corresponding element. However, these feature vectors have high dimensionality, most of which are 0. High dimensionality input may cost too much computational resources and may induce the phenomenon of the curse of dimensionality, which can lead to extremely inferior performance for some models. Therefore, based on the assumption that similar drugs may interact with the same drugs, we do not use bit vectors as input, but use the Jaccard similarity calculated from bit vector. Jaccard similarity is calculated by the following equation.

$$\text{Jaccard}\,(A, B) = \frac{|\,A \cap B\,|}{|\,A \cup B\,|} = \frac{|\,A \cap B\,|}{|\,A\,| + |\,B\,| - |\,A \cap B\,|}$$

where *A* and *B* are original bit vectors of two drugs; $|\ A \cap B\ |$ is the intersection of *A* and *B*; $|\ A \cup B\ |$ is the union. Based on the drug–drug jaccard similarity, in Dataset1, each drug feature is represented as a corresponding 572-dimensional vector. Therefore, each drug with three features is represented by a 3*572-dimentional vector. In the same way, each drug is represented as a corresponding 3*1258-dimensional vector in Dataset2.

Multi-source drug fusion

In the multi-source drug fusion module, we use CNNs, auto-encoders with self-attention mechanism and Siamese network to perform drug fusion. The multi-source drug fusion can provide deep learning models with diverse information from different views to accurately predict DDI events, compared with only one way to fuse the information of one drug pair. Next, we will describe them in detail.

Convolutional neural networks

CNN has achieved satisfactory performance in the field of computer vision since it focuses on local information. In addition, parameter sharing of CNN can save a lot of computational resources compared with fully connected layers, and its nature of translation invariance can focus on extracting location-insensitive information of descriptors. In our model, each drug is represented as a 1**k*-dimensional vector. We combine two drug vectors into a 2**k*-dimensional matrix and input it into the CNN. The size of the convolution kernel is 2**p*. Therefore, the CNN will output a row vector as the latent vector of the drug pair. Then input this latent vector into the 1-dimensional CNN to get the final latent vector of the drug pair.

Auto-encoders with self-attention mechanism

Auto-encoder is an unsupervised neural network model, which includes two parts: encoder and decoder. It can learn the hidden features from the input data. Since the input includes three feature similarities, and the dimensionality of the input is high, it is necessary to reduce the dimensionality of input features. Auto-encoder is a decent method to reduce the dimensionality and fuse three features together. In addition, with the help of auto-encoder outputs, an auxiliary loss can be used as a kind of deep learning network regularization method to improve the final performance.

The self-attention mechanism is a variant of the attention mechanism, which reduces the dependence on external information and is better at capturing the internal correlation of data or features. What's more, the self-attention mechanism can pay more attention to important features. Therefore, we add a self-attention layer before the output layer of the encoder in the auto-encoder.

In our model, we concatenate two drugs into a 1*2 *k*-dimensional vector and input it into an auto-encoder with a self-attention mechanism (named AE1). In addition, the 1**k*-dimensional vector obtained by element-wise adding two drug vectors is fed into another auto-encoder with a self-attention mechanism (named AE2). The latent vectors of two auto-encoders are used as the latent vectors of the drug pair.

Siamese network

The Siamese network [60] reduces the parameters of the neural network by sharing weights. It is often used to measure the similarity of two inputs. The applications of the Siamese network in object tracking and semantic similarity analysis have achieved satisfactory performance. In our model, we use another two auto-encoders with self-attention mechanism as sub-networks of the Siamese network. Two drugs are fed into the Siamese network respectively, and two latent vectors of the Siamese network are used as the latent vectors of the drug pair. Because the parameters of the two sub-networks are shared, so latent features can contain information of drug pairs. Siamese auto-encoder network can extract information from a single drug rather than a drug pair, and this network extracts drug-level features, instead of drug-pair-level features. In addition, this network can reduce the difference between two different drug input orders. In another word, the input with Drug A first and Drug B last is same as the input with Drug B first and Drug A last in this siamese network. However, the input with Drug A first and the input with Drug B first perform different in other components, such as a single auto-encoder and CNN.

Latent feature fusion

We use the encoder structure of the transformer [61] to perform latent feature fusion. Through four different drug fusion methods and network structures, we can obtain four different latent vectors of drug pairs. Then, we element-wise add these four latent vectors to get the fifth hidden vector. Finally, we concatenate these five hidden vectors as the new feature of the drug pair, and input this new feature into the encoder structure of the transformer (Figure 1). The transformer's encoder mainly includes self-attention layer, layer normalization, residual connections and feed-forward layer.

Multi-head attention mechanism

After obtaining the latent feature vectors of different drug combinations, we use the multi-head self-attention mechanism to perform feature fusion. Self-attention mechanism is a suitable method for features fusion [62]. Since some extracted features are redundant or less important, Transformer blocks with self-attention mechanism can help the network select important features [62], and give these important features with high weights for DDI events prediction.

Different latent feature vectors have inconsistent contributions to the prediction of drug interaction events. Therefore, we concatenate different latent vectors together and input them to the multi-head attention module, which is also called the transformer block, and use the self-attention mechanism to learn the weight distribution of different features. In another word, self-attention mechanism can help us to recognize which features are important for prediction. The multi-head attention is calculated by following formulas.

$$X_{\mathrm{MH\_attn}} = \mathrm{Concat}\left(\mathrm{head}_1, \mathrm{head}_2, \ldots, \mathrm{head}_{\mathrm{m}}\right) W^{o}$$

$$\mathrm{Head}_i = \mathrm{softmax}\left(\frac{Q_i \times K_i^T}{\sqrt{d_k}}\right) V_{i.}$$

$$\mathrm{Q_i} = \mathrm{X} \times \mathrm{W_i^Q}$$

$$K_\mathrm{i} = \mathrm{X} \times \mathrm{W_i^K}$$

$$\mathrm{V_i} = \mathrm{X} \times \mathrm{W_i^V}$$

$$X = \mathrm{Concat}\left(\mathrm{LF1}, \mathrm{LF2}, \mathrm{LF3}, \mathrm{LF4}, \mathrm{LF5}\right)$$

where LF1, LF2, LF3, LF4, LF5 are the latent feature vectors of different drug combinations, and X is the latent vector obtained by concatenating LF1, LF2, LF3, LF4 and LF5. $\mathrm{W_i^Q} \in \mathrm{R}^{\mathrm{d_{in}} \times \mathrm{d_Q}}$, $\mathrm{W_i^K} \in \mathrm{R}^{\mathrm{d_{in}} \times \mathrm{d_K}}$, $\mathrm{W_i^V} \in \mathrm{R}^{\mathrm{d_{in}} \times \mathrm{d_V}}$ are the parameter matrices. Q_i, K_i and V_i are the Q (Query), K (Key) and V (Value) matrices derived from the linear transformation of X, respectively.

Residual connections & layer normalization

Residual connection [63] can partially solve the problem of gradient disappearance, which can help the neural network design deeper. This is done by adding the output of the current layer and the output of the previous layer.

Layer normalization [64] was used in two occasions: after the self-attention layer and after the feed-forward network layer, with the goal to ameliorate the 'covariate-shift' problem by re-standardizing the computed vector representations. It can also accelerate the convergence of neural network parameters.

Data augmentation

Data augmentation is to help networks learn more representations from the original data without substantially increasing the data size, because it improves the quantity of the original data. In deep learning theory, training neural network with more data is likely to achieve better performance. Mixup [65] is a data augmentation algorithm, which obtains new training data by mixing different samples of data. It can be used to improve the generalization ability and robustness of the model. The mixup algorithm is calculated by following formulas:

$$\lambda = \mathrm{Beta}\left(\alpha, \beta\right)$$

$$\mathrm{Mixed\_batch}_x = \lambda \times \mathrm{batch}_{x1} + (1 - \lambda) \times \mathrm{batch}_{x2}$$

$$\mathrm{Mixed\_batch}_y = \lambda \times \mathrm{batch}_{y1} + (1 - \lambda) \times \mathrm{batch}_{y2}$$

where batch_{x1} is a set of batch samples, batch_{y1} is a set of labels corresponding to the batch samples. Shuffle the same batch samples to get batch_{x2}, and batch_{y2} is a set of labels corresponding to batch_{x2}. λ is the mixing coefficient calculated from the beta distribution with the hyper-parameters α and β. $\mathrm{Mixed\_batch}_x$ is a set of batch samples after mixup, and $\mathrm{mixed\_batch}_y$ is a set of labels corresponding to the $\mathrm{mixed\_batch}_x$.

Loss function

DDI events prediction is a multi-class classification problem, and the sample size of each class is not balanced. Therefore, we choose focal loss (FL) [66] as our classification loss function. FL can solve problems of imbalance in sample size of each category and difficulty of imbalanced classification. However, due to the instability of FL during training, the neural network needs to be trained by more epochs or even does not converge. Therefore, we still choose the cross-entropy (CE) loss function as our classification loss function in the first half training steps, and choose FL as our classification loss function in the second half training steps.

In our model, AE1, AE2 and Siamese network are all auto-encoders, so we choose the mean squared error (MSE) loss function as the loss function of the auto-encoder, which is the auxiliary loss for the classification loss. In order to make the model pay more attention to classification loss, we multiply the classification cross entropy loss and focal loss by a corresponding classification loss weight (clw). Therefore, the total loss function of the model is as follows:

$$\mathrm{Loss} = \begin{cases} \mathrm{clw} \times \mathrm{l_{CE}}\left(\mathrm{y}, \mathrm{y}^{\sim}\right) + \mathrm{l_{MSE}}\left(\mathrm{x_1}, \mathrm{x_1^{\sim}}\right) + \mathrm{l_{MSE}}\left(\mathrm{x_2}, \mathrm{x_2^{\sim}}\right) \\ \quad + \mathrm{l_{MSE}}\left(\mathrm{x_3}, \mathrm{x_3^{\sim}}\right) \text{ if epoch} < \text{epoch\_num//2} \\ \mathrm{clw} \times \mathrm{l_{FL}}\left(\mathrm{y}, \mathrm{y}^{\sim}\right) + \mathrm{l_{MSE}}\left(\mathrm{x_1}, \mathrm{x_1^{\sim}}\right) + \mathrm{l_{MSE}}\left(\mathrm{x_2}, \mathrm{x_2^{\sim}}\right) \\ \quad + \mathrm{l_{MSE}}\left(\mathrm{x_3}, \mathrm{x_3^{\sim}}\right) \text{ if epoch} \geq \text{epoch\_num//2} \end{cases}$$

where y is the class label of the sample, and $\mathrm{y}^{\sim}$ is the predicted value of the sample. x_1 is the input of AE1, and $x_1^{\sim}$ is the output of the AE1 decoder. x_2 is the input of AE2, and $x_2^{\sim}$ is the output of the AE2 decoder. x_3 is the input of the Siamese network, and $x_3^{\sim}$ is the output of the Siamese network decoder.

Results and Discussion

Experimental settings and tasks

This study evaluated the DDI events prediction problem through three experimental settings: (i) prediction of unobserved interaction events between known drugs (Task 1); (ii) prediction of interaction events between known drugs and new drugs (Task 2) and (iii) prediction of interaction events between new drugs (Task 3). New drug representations in the corresponding task are missing in the training set, but exist in the test set.

For task 1, we apply 5-fold cross-validation (5-CV) to DDIs and split all DDIs into five subsets. We train models based on DDIs in the training set, and then make predictions for DDIs in the test set. For task 2 and task 3, we apply 5-CV to drugs instead of DDI pairs. We randomly split drugs into five subsets, and used four of them as training drugs and the remaining as test drugs. For task 2, prediction models are constructed on the DDIs between two training drugs, and then make predictions for DDI events between one training drug and one test drug. For task 3, prediction models are constructed on the DDIs between two training drugs, and then make predictions for DDI events between two test drugs [3].

Our task is the multi-class classification work. For evaluation, accuracy (ACC), area under the precision–recall-curve (AUPR), area under the ROC curve (AUC), F1 score, precision and recall are adopted as evaluation metrics [3]. In highly imbalanced data, AUPR and F1 score metrics are more objective for model evaluation. Therefore, in the following discussion, we will pay more attention to these two metrics.

Hyper-parameter setting

The hyper-parameters could influence the performances of MDF-SA-DDI. Therefore, we discuss six hyper-parameters on task 1 of Dataset1: batch size (BS), classification loss weight (CLW), hidden layer dimension of auto-encoders (HLD), self-attention module layers (SML), learning rate (LRA) and training epochs (TE). We use Gaussian error linear unit [67] activation function and Radam optimizer [68]. The dropout layer and batch normalization layer [69] are used between the fully connected layers. The metric scores under different configurations are shown in Figure 2.

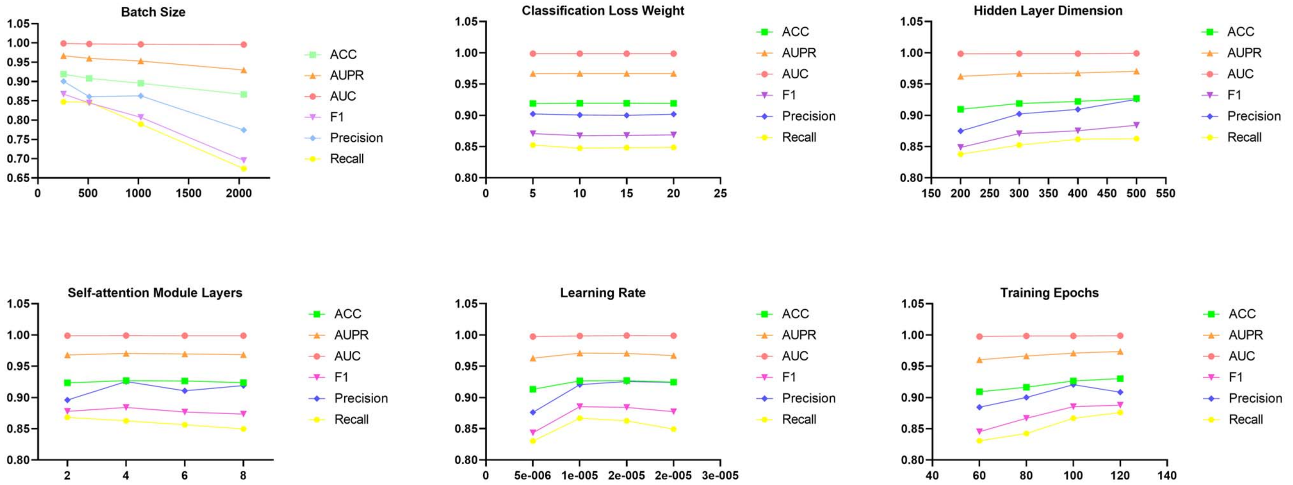

Figure 2. The metric scores under different hyper-parameters.

According to Figure 2, the performance of the model does not change drastically as the hyper-parameters change. This also illustrates the stability of our model. In the end, we chose 256 for batch size, 5 for classification loss weight, 500 for hidden layer dimension of auto-encoders, 4 for the number of self-attention module layers, 1e-5 for learning rate and 120 for training epoch.

For different tasks on different datasets, we fine-tuned the hyper-parameters. In task 2 and task 3 of Dataset1, we found that the loss of the test set can quickly converge but it is unstable after convergence during training. We speculate that the learning rate may be too large, so we reduced the learning rate in task 2 and task 3. In addition, task 2 and task 3 are more difficult than task 1. Therefore, in order to enhance the generalization ability of the model, we have also increased hidden layer dimension of auto-encoders. The size of Dataset2 is more than four times larger than that of Dataset1. Therefore, in order to speed up the training and make the training more stable, we increased the batch size in Dataset2. A larger batch size often requires a larger learning rate, so we also increase the learning rate. In order to make the model fit the larger dataset better, we increased hidden layer dimension of auto-encoders and training epochs. On task 1 of Dataset1, we roughly determined the range of each hyper-parameter. All hyper-parameter adjustments are based on the hyper-parameters on task 1 of Dataset1. The specific hyper-parameter settings are given in Supplementary Table S1.

Multi-source drug fusion improves DDI event prediction

In this section, we evaluated the influence of different drug fusions on the DDI event prediction. To compare the performance of only one kind of drug fusion and multiple versions of drug fusion, we build several MDF-SA-DDI models and adopt the metric scores of the models to assess the prediction power of corresponding drug fusion or multi-source drug fusion. The results of all prediction models are shown in Table 2.

Among all drug fusions, the element-wise summation of feature vectors of two drugs (AE2) seems to be the most informative and achieves an AUPR of 0.9476. The model which is the concatenating two drugs into a 1*2 k-dimensional vector (AE1) produces an AUPR of 0.9347, and the model which inputs features of two drugs into the Siamese network (SN) produces an AUPR of 0.9327. Combining two drug features into a 2*k-dimensional feature vector (CN) leads to the model with an AUPR of 0.2104. The AUPR in this drug combination method is low, and this is probably because 2*k-dimensional feature vectors are only suitable for 2D CNN to extract features. The combination of several different drug fusions provides the significant improvement compared with only one single version of drug fusions. The combination of CN and SN produces the best AUPR (0.9712) among all combinations of two versions of drug fusion. The combination of CN and AE2 produces the best F1 score (0.8687) among all combinations of two versions of drug fusion. The combination of CN, SN and AE1 produces the best AUPR (0.9722) among all combinations of three versions of drug fusion. The combination of CN, AE1 and AE2 produces the best F1 score (0.8814) among all combinations of three versions of drug fusion. The combination of CN, SN, AE1 and AE2 performs the best on all evaluation metrics in all combinations of drug fusions.

As mentioned in Multi-source drug fusion section, multi-source drug fusion can provide deep learning models with diverse information from different views to accurately predict DDI events, compared with only one way to fuse the information of one drug pair. The autoencoder can learn the characteristics of drug pairs in different combinations (element-wise addition and concatenation). Siamese network can extract information from a single drug rather than a drug pair, and this network extracts drug-level features, instead of drug-pair-level features. CNNs have powerful capabilities in local feature extraction. This is why multi-source drug fusion can improve the prediction of DDI events.

The effect of mixup data augmentation

In order to verify whether the mixup data augmentation algorithm works in our model, we verified the effectiveness of the mixup data augmentation algorithm in task 1, task 2 and task 3 of Dataset1. The specific approach for verification is to use the mixup data augmentation algorithm and not to use the mixup data augmentation algorithm on three different tasks. The effectiveness of mixup is determined by comparing the metric scores. The results of all prediction models are shown in Table 3

In all three tasks, the F1 score of the model using mixup is higher than that of the model without using mixup. The F1 score of the model using mixup in task 1 is 0.0305 higher than that without using mixup. The F1 score of the model using mixup in

Table 2. The performance of MDF-SA-DDI with different drug fusions

	ACC	AUPR	AUC	F1	Precision	Recall
CN	0.2630	0.2104	0.9346	0.0086	0.0080	0.0154
SN	0.8712	0.9327	0.9971	0.7138	0.7844	0.6863
AE1	0.8806	0.9347	0.9965	0.7705	0.7931	0.7747
AE2	0.8930	0.9476	0.9972	0.7808	0.8070	0.7803
CN + SN	0.9217	0.9712	0.9988	0.8489	0.8799	0.8332
CN + AE1	0.9161	0.9662	0.9987	0.8621	0.8849	0.8503
CN + AE2	0.9198	0.9661	0.9987	0.8687	0.9089	0.8482
SN + AE1	0.9162	0.9639	0.9983	0.8611	0.8818	0.8562
SN + AE2	0.9187	0.9636	0.9982	0.8657	0.8737	0.8659
AE1 + AE2	0.9138	0.9613	0.9982	0.8554	0.8757	0.8515
CN + SN + AE1	0.9263	0.9722	0.9988	0.8705	0.9008	0.8492
CN + SN + AE2	0.9241	0.9690	0.9987	0.8758	0.9052	0.8619
CN + AE1 + AE2	0.9242	0.9697	0.9989	0.8814	0.9033	0.8693
SN + AE1 + AE2	0.9248	0.9685	0.9986	0.8777	0.9042	0.8675
CN + SN + AE1 + AE2	0.9301	0.9737	0.9989	0.8878	0.9085	0.8760

Table 3. The performance of MDF-SA-DDI with/without mixup data augmentation

	ACC	AUPR	AUC	F1	Precision	Recall
Task1_with_mixup	0.9301	0.9737	0.9989	0.8878	0.9085	0.8760
Task1_without_mixup	0.9315	0.9737	0.9982	0.8573	0.8907	0.8393
Task2_with_mixup	0.6633	0.6776	0.9497	0.5584	0.6547	0.5078
Task2_without_mixup	0.6507	0.6473	0.9367	0.5124	0.5140	0.5270
Task3_with_mixup	0.4338	0.3873	0.8630	0.2329	0.2715	0.2226
Task3_without_mixup	0.4239	0.3249	0.8373	0.2071	0.1970	0.2329

task 2 is 0.0460 higher than that without using mixup. The F1 score of the model using mixup in task 3 is 0.0258 higher than that without using mixup. The AUPR of the model using mixup in task 2 and task 3 is higher than that of the model without using mixup. In task 1, the AUPR of the model using mixup is equal to that of the model without using mixup.

In general, by using the mixup data enhancement algorithm, the performance of the model has been improved on task 1, task 2 and task 3. The performance of the model on task 2 and task 3 is improved more obviously. This may be due to the distinct training/testing data splitting methods in task 2 and task 3. The test sets of task 2 and task 3 contain drugs that have not appeared in the training sets, so the sample distribution of the test sets and the distribution of the training sets have a greater difference than that of task 1. The mixup data enhancement algorithm can improve the generalization ability of the model by mixing data samples between different categories. This may be the reason why the mixup data enhancement algorithm performs better on task 2 and task 3.

Comparison with other methods

Dataset1

We compared MDF-SA-DDI with the state-of-the-art event prediction methods DDIMDL [3], DeepDDI [50] and Lee *et al.*'s methods [2]. We also considered several popular classification methods, namely fully connected DNN, RF, KNN and LR. We compare MDF-SA-DDI with these models to demonstrate the advantages of our model. Therefore, models based on RF, KNN, LR and DNN are used as baseline methods.

Task 1 is important for DDI prediction. We evaluate performances of all prediction methods for task 1, which predicts DDI-associated events between known drugs. The evaluation scores of all prediction models are shown in Table 4, and the results demonstrate that MDF-SA-DDI produces better performances than other methods in terms of all metrics. Furthermore, we investigate the performances of MDF-SA-DDI for each event and calculate the metric scores for events independently by using predicted scores and real labels. The AUPR scores and AUC scores of all prediction models for each event are shown in Figure 3. The original AUPR scores and AUC scores are listed in Supplementary Tables S2 and S3. It is likely that the events with higher frequency can gain better performances. Among 65 events, MDF-SA-DDI achieved the highest AUPR scores in 60 events and the highest AUPR scores in 52 events, far better than the other seven methods. In general, Figure 3 demonstrates that MDF-SA-DDI produces greater AUPR scores and AUC scores than other methods in most types of events. To further analyze the performances of prediction models, we use Figure 4 to display AUPR scores and AUC scores of different methods for 65 types of events. These boxplots clearly show that MDF-SA-DDI produces better performances for these events than the competing methods.

Moreover, we evaluate the performances of prediction models for task 2 and task 3. Here, we mainly compare our model with DDIMDL, Lee *et al*' method, DeepDDI and DNN, because the results for task 1 show that DDIMDL, Lee *et al*' method, DeepDDI and DNN are more competitive. The performances of all prediction methods are shown in Table 4. It could be concluded that without prior knowledge about the new drugs, the performances of all models for task 2 and task 3 decrease, especially for task 3. The experimental results also demonstrate that MDF-SA-DDI outperforms all other state-of-the-art methods for task 2 and task 3 except for AUC of ROC, which corroborates the

Table 4. The performance of different methods on Dataset1

		ACC	AUPR	AUC	F1	Precision	Recall
Task1	MDF-SA-DDI	**0.9301**	**0.9737**	**0.9989**	**0.8878**	**0.9085**	**0.8760**
	DDIMDL	0.8852	0.9208	0.9976	0.7585	0.8471	0.7182
	Lee *et al.*'s methods	0.9094	0.9562	0.9961	0.8391	0.8509	0.8339
	DeepDDI	0.8371	0.8899	0.9961	0.6848	0.7275	0.6611
	DNN	0.8797	0.9134	0.9963	0.7223	0.8047	0.7027
	RF	0.7775	0.8349	0.9956	0.5936	0.7893	0.5161
	KNN	0.7214	0.7716	0.9813	0.4831	0.7174	0.4081
	LR	0.7920	0.8400	0.9960	0.5948	0.7437	0.5236
Task2	MDF-SA-DDI	**0.6633**	**0.6776**	0.9497	**0.5584**	**0.6547**	**0.5078**
	DDIMDL	0.6415	0.6558	**0.9799**	0.4460	0.5607	0.4319
	Lee *et al.*'s methods	0.6405	0.6244	0.9247	0.5039	0.5388	0.4891
	DeepDDI	0.5774	0.5594	0.9575	0.3416	0.3630	0.3890
	DNN	0.6239	0.6361	0.9796	0.2997	0.4237	0.2840
Task3	MDF-SA-DDI	**0.4338**	**0.3873**	0.8630	**0.2329**	**0.2715**	**0.2226**
	DDIMDL	0.4075	0.3635	0.9512	0.1590	0.2408	0.1452
	Lee *et al.*'s methods	0.4097	0.3184	0.8302	0.2022	0.2216	0.2027
	DeepDDI	0.3602	0.2781	0.9059	0.1373	0.1586	0.1450
	DNN	0.4087	0.3776	**0.9550**	0.1152	0.1836	0.1093

The performance of different methods on Dataset1. The best results are highlighted in boldface.

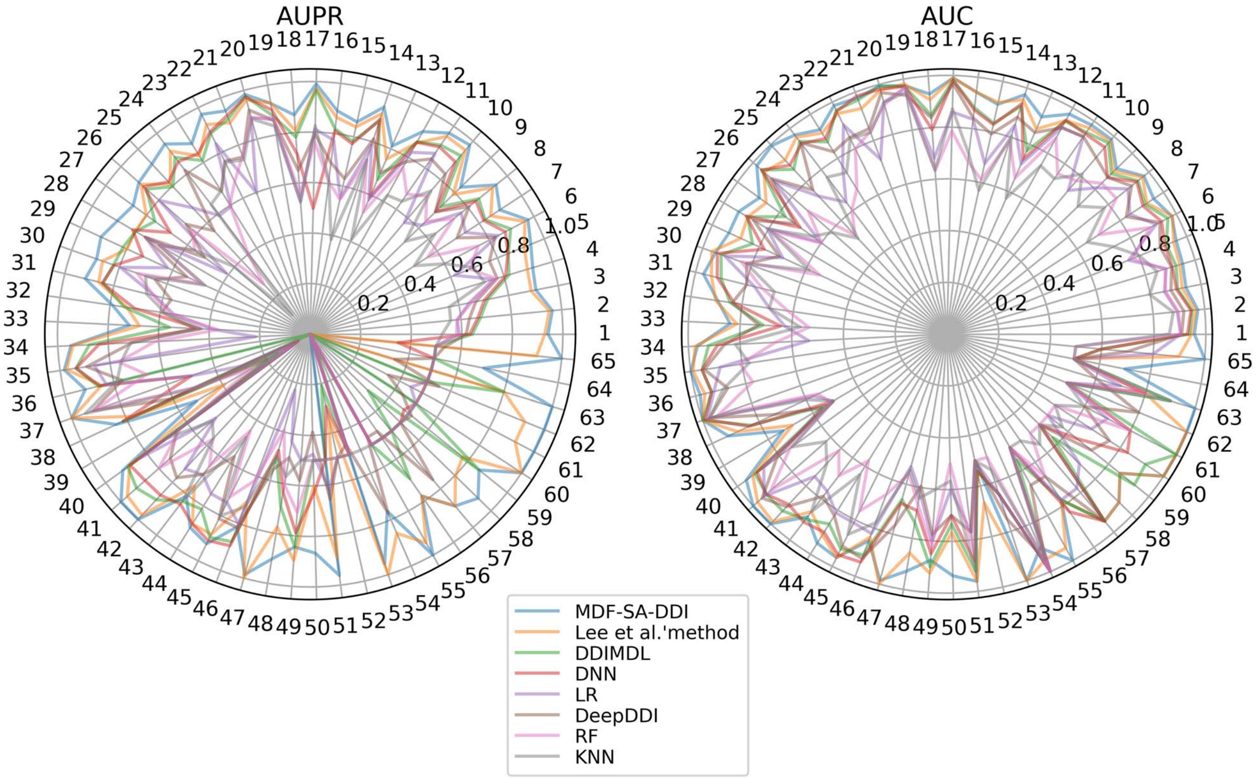

Figure 3. The AUPR scores and AUC scores of all prediction models for each event.

effectiveness of our model again. In the multi-class classification problem with imbalanced samples, the value of AUC of ROC cannot objectively evaluate the model.

The studies show that deep learning and multi-source drug fusions are effective for the DDI event prediction, and our multi-source drug fusions in the deep learning framework outperforms the traditional classifiers and deep network structures in previous studies.

Dataset2

We also used larger data set (Dataset2) to verify the effectiveness of our proposed method. We compared our method with the state-of-the-art methods and classic machine learning methods on three different tasks. The experimental results show that our method can achieve the same or better performance than the most advanced methods and machine learning methods.

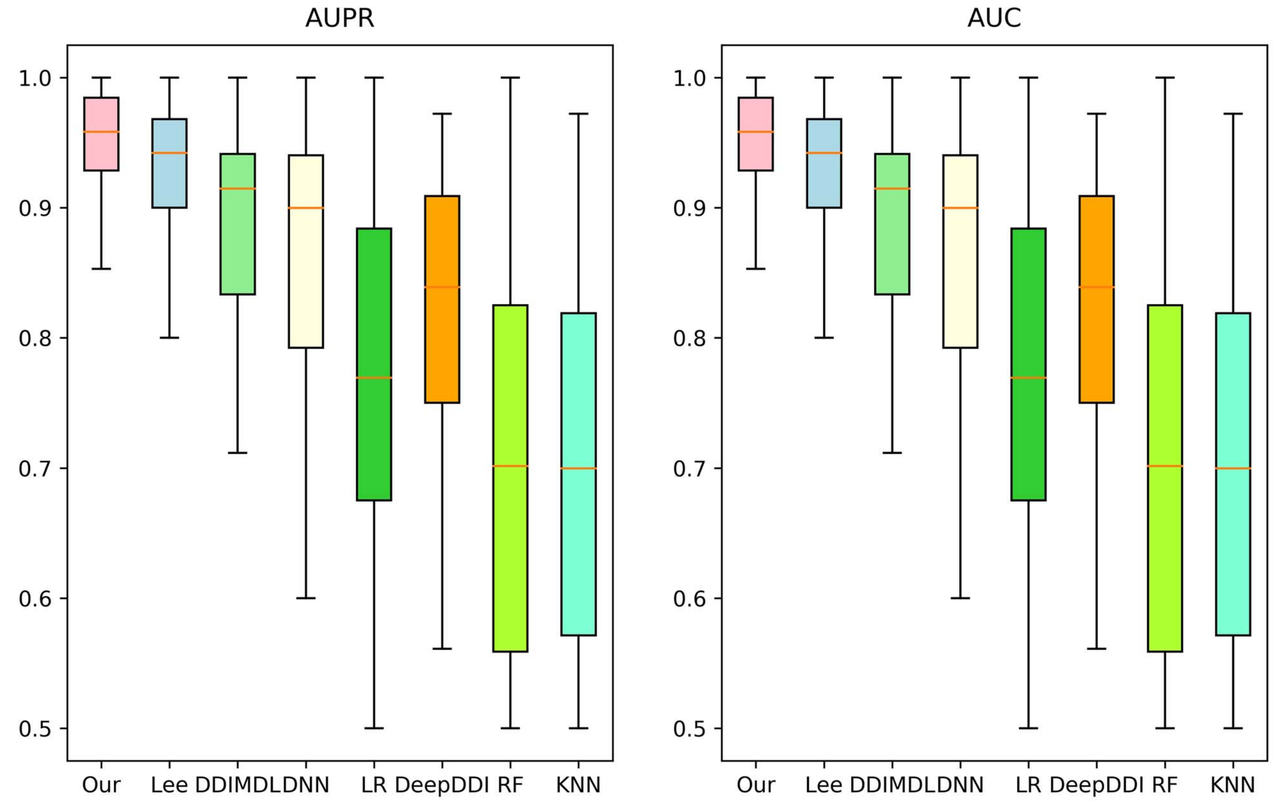

Figure 4. Boxplots displaying the AUPR and AUC of compared methods for each event.

The evaluation scores of all prediction methods are shown in Table 5. In Task 1, the AUPR of MDA-SA-DDI is 0.9773, which is 0.0018 worse than the highest AUPR, the F1 score of MDA-SA-DDI is 0.9117, which is 0.0064 worse than the highest F1 score. The AUC and precision of MDA-SA-DDI achieved the best performance, 0.9996 and 0.9381, respectively. As we all know, the hyper-parameters of the deep learning model can affect the performance of the model, and most of the hyperparameters of MDA-SA-DDI are obtained from the Dataset1, so it may affect the performance of MDA-SA-DDI on this larger data set. This may be the reason why MDA-SA-DDI is slightly worse than the state-of-the-art methods in some tasks. DDIMDL and Lee *et al.*'s method achieve good performance in Dataset2. The similarity between the two methods is that they both separate different types of features, and then send different types of features to the model for training, instead of combining different types of feature channels. Another interesting result is that the performance of DNN in Task1 is not very good, but the performance in Task 2 and Task 3 is very good. This may be because Task 2 and Task 3 are prone to over-fitting during the training process, so it is possible that simple models are not prone to over-fitting, and DNN performs very well. In general, our model performs well on Dataset2, which also proves that our model is suitable for different types and sizes of data sets.

Since there are more data on Dataset2, MDF-SA-DDI has achieved better performance on all three tasks of Dataset2, especially in task 2 and task 3. In task 2, the F1 score of MDF-SA-DDI on Dataset2 is 0.0335 higher than that of Dataset1. In task 3, the AUPR and f1 scores of MDF-SA-DDI on Dataset2 are 0.4450 and 0.0608 higher than those of Dataset1, respectively. The performance of other deep learning-models has also been improved in Dataset1. This may indicate that the hard task 2 and task 3 may be solved by increasing the data set for deep learning-based models. However, with the increasing size of the training data and DDI event types, the performance of the machine learning-based model on Dataset2 is worse than Dataset1. That may be because the fitting ability of the machine learning-based models is not enough to fit such a large amount of data and so many DDI event types.

Case study

In this section, we conduct case studies to validate the usefulness of MDF-SA-DDI in practice.

We use all DDIs and their events in our dataset which were originally obtained from DrugBank to train the prediction model, and then make predictions for other drug–drug pairs. We pay attention to five events with the highest frequencies and check up the top 20 predictions related to each event. We used the Interactions Checker tool provided by drugs.com to validate these predictions.

Thirty-five DDI events can be confirmed among 100 events, and they are shown in Supplementary Table S4. For example, the interaction between Etodolac and Dienogest is predicted to cause the event #0, and means Etodolac may decrease the excretion rate of Dienogest which could result in a higher serum level. The interaction between Etodolac and Lercanidipine is predicted to cause the event #1, and means Dopamine may decrease the antihypertensive activities of Lercanidipine. More evidence about confirmed DDI events is provided in Supplementary Table S4.

In addition, we also found that a certain drug may be closely related to a certain DDI event. For example, 10 of the top 20

Table 5. The performance of different methods on Dataset2

		ACC	AUPR	AUC	F1	Precision	Recall
Task1	MDF-SA-DDI	0.9291	0.9773	0.9996	0.9117	0.9381	0.8910
	DDIMDL	0.9229	0.9637	0.9993	0.9105	0.9212	0.9039
	Lee *et al.*'s methods	0.9370	0.9791	0.9991	0.9181	0.9226	0.9153
	DeepDDI	0.7211	0.7724	0.9914	0.6854	0.6654	0.7183
	DNN	0.7908	0.8539	0.9949	0.7649	0.7560	0.8046
	RF	0.6956	0.7567	0.9892	0.5760	0.6694	0.5426
	KNN	0.5797	0.5964	0.8998	0.3805	0.4758	0.3347
	LR	0.5229	0.5288	0.9805	0.2373	0.3128	0.2185
Task2	MDF-SA-DDI	0.6664	0.6820	0.9862	0.5919	0.6526	0.5518
	DDIMDL	0.6720	0.7086	0.9885	0.5817	0.6680	0.5295
	Lee *et al.*'s methods	0.6917	0.7119	0.9687	0.5934	0.6144	0.5848
	DeepDDI	0.5883	0.5851	0.9746	0.4709	0.5250	0.4361
	DNN	0.6687	0.6838	0.9818	0.6164	0.7279	0.5479
Task3	MDF-SA-DDI	0.4794	0.4450	0.9686	0.2937	0.3667	0.2659
	DDIMDL	0.4699	0.4386	0.9685	0.3032	0.3773	0.2729
	Lee *et al.*'s methods	0.4867	0.4349	0.9093	0.3082	0.3355	0.3066
	DeepDDI	0.3611	0.2820	0.9264	0.1868	0.2301	0.1711
	DNN	0.4570	0.4129	0.9565	0.2997	0.4345	0.2508

predictions related to event #2 (the serum concentration increase) are related to Ceritinib. A total of 16 of the top 20 predictions related to event #4 (the therapeutic efficacy decrease) are related to Naltrexone.

Conclusion

We proposed a DDI event prediction model based on multi-source drug fusion, multi-source feature fusion and transformer self-attention mechanism, and proved the effectiveness and robustness of our model. In addition, we also proved the effectiveness of the mixup data augmentation strategy. Experimental results have proved that our proposed model is better than the state-of-the-art models. The case studies were also performed to identify the new DDI events which are not included in our dataset.

Key Points

- This study proposed a novel DDI event prediction method, MDF-SA-DDI: predicting drug–drug interaction events based on multi-source drug fusion, multi-source feature fusion and transformer self-attention mechanism.
- This study proved the effectiveness of multi-modal drug fusion and mixup data augmentation algorithm.
- The MDF-SA-DDI method has achieved better performance than classic machine learning algorithms and the state-of-the-art DDI prediction methods on all three tasks and two datasets.
- The case studies for five DDI events were conducted, which confirmed the effectiveness of the proposed method.

Supplementary data

Supplementary data are available online at *Briefings in Bioinformatics*.

Code and data availability

The source codes and data are available at https://github.com/ShenggengLin/MDF-SA-DDI.

Funding

This work was supported by the grants from the National Science Foundation of China (grant numbers 62172274, 32070662, 61832019, 32030063]; the Key Research Area Grant (2016YFA0501703) of the Ministry of Science and Technology of China; the Science and Technology Commission of Shanghai Municipality (grant number 19430750600), as well as SJTU JiRLMDS Joint Research Fund and Joint Research Funds for Medical and Engineering and Scientific Research at Shanghai Jiao Tong University (ZH2018QNA41, YG2019GD01, YG2019ZDA12, YG2021ZD02). The computations were partially performed at the Pengcheng Lab and the Center for High-Performance Computing, Shanghai Jiao Tong University.

References

1. Zitnik M, Agrawal M, Leskovec J. Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics* 2018;**34**:i457–66.
2. Lee G, Park C, Ahn J. Novel deep learning model for more accurate prediction of drug-drug interaction effects. *BMC Bioinformatics* 2019;**20**:415.
3. Deng Y, Xu X, Qiu Y, *et al.* A multimodal deep learning framework for predicting drug–drug interaction events. *Bioinformatics* 2020;**36**:4316–22.
4. Yan C, Duan G, Pan Y, *et al.* DDIGIP: predicting drug-drug interactions based on Gaussian interaction profile kernels. *BMC Bioinformatics* 2019;**20**:538.
5. Qian S, Liang S, Yu H. Leveraging genetic interactions for adverse drug-drug interaction prediction. *PLoS Comput Biol* 2019;**15**:e1007068.
6. Liu S, Tang B, Chen Q, *et al.* Drug-drug interaction extraction via convolutional neural networks. *Comput Math Methods Med* 2016;**2016**:6918381.

7. Kastrin A, Ferk P, Leskosek B. Predicting potential drug-drug interactions on topological and semantic similarity features using statistical learning. *PLoS One* 2018;**13**:e196865.
8. Gottlieb A, Stein GY, Oron Y, *et al.* INDI: a computational framework for inferring drug interactions and their associated recommendations. *Mol Syst Biol* 2012;**8**:592.
9. Cami A, Manzi S, Arnold A, *et al.* Pharmacointeraction network models predict unknown drug-drug interactions. *PLoS One* 2013;**8**:e61468.
10. Cheng F, Zhao Z. Machine learning-based prediction of drug–drug interactions by integrating drug phenotypic, therapeutic, chemical, and genomic properties. *J Am Med Inform Assoc* 2014;**21**:e278–86.
11. Yu Y, Huang K, Zhang C, *et al.* SumGNN: multi-typed drug interaction prediction via efficient knowledge graph summarization. *Bioinformatics* 2021. doi: 10.1093/bioinformatics/btab207.
12. Fatehifar M, Karshenas H. Drug-drug interaction extraction using a position and similarity fusion-based attention mechanism. *J Biomed Inform* 2021;**115**:103707.
13. Zhang Y, Qiu Y, Cui Y, *et al.* Predicting drug-drug interactions using multi-modal deep auto-encoders based network embedding and positive-unlabeled learning. *Methods* 2020;**179**:37–46.
14. Kumar SP, Kumar SP, Sharma P, *et al.* Efficient prediction of drug-drug interaction using deep learning models. *IET Syst Biol* 2020;**14**:211–6.
15. Feng YH, Zhang SW, Shi JY. DPDDI: a deep predictor for drug-drug interactions. *BMC Bioinformatics* 2020;**21**: 419.
16. Dai Y, Guo C, Guo W, *et al.* Drug-drug interaction prediction with Wasserstein adversarial autoencoder-based knowledge graph embeddings. *Brief Bioinform* 2020:**22**.
17. Zheng Y, Peng H, Zhang X, *et al.* DDI-PULearn: a positive-unlabeled learning method for large-scale prediction of drug-drug interactions. *BMC Bioinformatics* 2019;**20**: 661.
18. Rohani N, Eslahchi C. Drug-drug interaction predicting by neural network using integrated similarity. *Sci Rep* 2019;**9**:13645.
19. Celebi R, Uyar H, Yasar E, *et al.* Evaluation of knowledge graph embedding approaches for drug-drug interaction prediction in realistic settings. *BMC Bioinformatics* 2019;**20**:726.
20. Zhou D, Miao L, He Y. Position-aware deep multi-task learning for drug-drug interaction extraction. *Artif Intell Med* 2018;**87**:1–8.
21. Zhang Y, Zheng W, Lin H, *et al.* Drug-drug interaction extraction via hierarchical RNNs on sequence and shortest dependency paths. *Bioinformatics* 2018;**34**:828–35.
22. Zheng W, Lin H, Luo L, *et al.* An attention-based effective neural model for drug-drug interactions extraction. *BMC Bioinformatics* 2017;**18**:445.
23. Rohani N, Eslahchi C. Drug-drug interaction predicting by neural network using integrated similarity. *Sci Rep* 2019;**9**:13645.
24. Zhu J, Liu Y, Zhang Y, *et al.* An attribute supervised probabilistic dependent matrix tri-factorization model for the prediction of adverse drug-drug interaction. *IEEE J Biomed Health Inform* 2020;**25**:2820–2832.
25. Zhang W, Jing K, Huang F, *et al.* SFLLN: A sparse feature learning ensemble method with linear neighborhood regularization for predicting drug–drug interactions. *Inform Sci* 2019;**497**:189–201.
26. Shi JY, Mao KT, Yu H, *et al.* Detecting drug communities and predicting comprehensive drug-drug interactions via balance regularized semi-nonnegative matrix factorization. *J Chem* 2019;**11**:28.
27. Yu H, Mao KT, Shi JY, *et al.* Predicting and understanding comprehensive drug-drug interactions via semi-nonnegative matrix factorization. *BMC Syst Biol* 2018;**12**:14.
28. Shi JY, Huang H, Li JX, *et al.* TMFUF: a triple matrix factorization-based unified framework for predicting comprehensive drug-drug interactions of new drugs. *BMC Bioinformatics* 2018;**19**:411.
29. Vilar S, Uriarte E, Santana L, *et al.* Detection of drug-drug interactions by modeling interaction profile fingerprints. *PLoS One* 2013;**8**:e58321.
30. Yu H, Mao KT, Shi JY, *et al.* Predicting and understanding comprehensive drug-drug interactions via semi-nonnegative matrix factorization. *BMC Syst Biol* 2018;**12**: 14.
31. Zhang W, Chen Y, Li D, *et al.* Manifold regularized matrix factorization for drug-drug interaction prediction. *J Biomed Inform* 2018;**88**:90–7.
32. Takeda T, Hao M, Cheng T, *et al.* Predicting drug-drug interactions through drug structural similarities and interaction networks incorporating pharmacokinetics and pharmacodynamics knowledge. *J Chem* 2017;**9**:16.
33. Sridhar D, Fakhraei S, Getoor L. A probabilistic approach for collective similarity-based drug-drug interaction prediction. *Bioinformatics* 2016;**32**:3175–82.
34. Huang J, Niu C, Green CD, *et al.* Systematic prediction of pharmacodynamic drug-drug interactions through protein-protein-interaction network. *PLoS Comput Biol* 2013;**9**:e1002998.
35. Guimera R, Sales-Pardo M. A network inference method for large-scale unsupervised identification of novel drug-drug interactions. *PLoS Comput Biol* 2013;**9**:e1003374.
36. Takarabe M, Shigemizu D, Kotera M, *et al.* Network-based analysis and characterization of adverse drug-drug interactions. *J Chem Inf Model* 2011;**51**:2977–85.
37. Zhang P, Wang F, Hu J, *et al.* Label propagation prediction of drug-drug interactions based on clinical side effects. *Sci Rep* 2015;**5**:12339.
38. Park K, Kim D, Ha S, *et al.* Predicting pharmacodynamic drug-drug interactions through signaling propagation interference on protein-protein interaction networks. *PLoS One* 2015;**10**:e140816.
39. Sridhar D, Fakhraei S, Getoor L. A probabilistic approach for collective similarity-based drug-drug interaction prediction. *Bioinformatics* 2016;**32**:3175–82.
40. Nyamabo AK, Yu H, Shi JY. SSI-DDI: substructure-substructure interactions for drug-drug interaction prediction. *Brief Bioinform* 2021. doi: 10.1093/bib/bbab133.
41. Cheng F, Zhao Z. Machine learning-based prediction of drug-drug interactions by integrating drug phenotypic, therapeutic, chemical, and genomic properties. *J Am Med Inform Assoc* 2014;**21**:e278–86.
42. Deepika SS, Geetha TV. A meta-learning framework using representation learning to predict drug-drug interaction. *J Biomed Inform* 2018;**84**:136–47.
43. Chen Y, Ma T, Yang X, *et al.* MUFFIN: multi-scale feature fusion for drug-drug interaction prediction. *Bioinformatics* 2021. doi: 10.1093/bioinformatics/btab169.
44. Liu N, Chen CB, Kumara S. Semi-supervised learning algorithm for identifying high-priority drug-drug interactions

through adverse event reports. *IEEE J Biomed Health Inform* 2020;**24**:57–68.

45. Asada M, Miwa M, Sasaki Y. Using drug descriptions and molecular structures for drug-drug interaction extraction from literature. *Bioinformatics* 2020.
46. Shen Y, Yuan K, Yang M, *et al.* KMR: knowledge-oriented medicine representation learning for drug-drug interaction and similarity computation. *J Chem* 2019;**11**:22.
47. Vilar S, Friedman C, Hripcsak G. Detection of drug-drug interactions through data mining studies using clinical sources, scientific literature and social media. *Brief Bioinform* 2018;**19**:863–77.
48. Zhang W, Chen Y, Liu F, *et al.* Predicting potential drug-drug interactions by integrating chemical, biological, phenotypic and network data. *BMC Bioinformatics* 2017;**18**:18.
49. Zhao Z, Yang Z, Luo L, *et al.* Drug drug interaction extraction from biomedical literature using syntax convolutional neural network. *Bioinformatics* 2016;**32**:3444–53.
50. Ryu JY, Kim HU, Lee SY. Deep learning improves prediction of drug–drug and drug–food interactions. *Proc Natl Acad Sci* 2018;**115**:E4304–11.
51. S JS, S QA, A MZ (2020), Random Forest vs. SVM vs. KNN in classifying Smartphone and Smartwatch sensor data using CRISP-DM, *2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI)*, pp. 1–4.
52. Y J, S Y, Y Z (2011), A novel Naive Bayes model: Packaged Hidden Naive Bayes, *2011 6th IEEE Joint International Information Technology and Artificial Intelligence Conference*, pp. 484–7.
53. J JD (1990), 'Logistic regression and the Boltzmann machine', *1990 IJCNN International Joint Conference on Neural Networks*, pp. 199–204.
54. Z W, B H, R K *et al.* (2018), Efficient Gradient Boosted Decision Tree Training on GPUs, *2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pp. 234–43.
55. N F, G K, S K *et al.* (2019), Self-trained eXtreme Gradient Boosting Trees, *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*, pp. 1–6.
56. Lin Y, Zhang W, Cao H, *et al.* Classifying breast cancer subtypes using deep neural networks based on multi-omics data. *Genes (Basel)* 2020;**11**. doi: 10.3390/genes11080888
57. Li GZ, Yan SX, You M, *et al.* Intelligent ZHENG classification of hypertension depending on ML-kNN and information fusion. *Evid Based Complement Alternat Med* 2012;**2012**:837245.
58. Wu G, Zheng R, Tian Y, *et al.* Joint ranking SVM and binary relevance with robust low-rank learning for multi-label classification. *Neural Netw* 2020;**122**:24–39.
59. Wang X, Wang Y, Xu Z, *et al.* ATC-NLSP: prediction of the classes of anatomical therapeutic chemicals using a network-based label space partition method. *Front Pharmacol* 2019;**10**:971.
60. Wu H, Pan X, Yang Y, *et al.* Recognizing binding sites of poorly characterized RNA-binding proteins on circular RNAs using attention Siamese network. *Brief Bioinform* 2021. doi: 10.1093/bib/bbab279.
61. Vaswani A, Shazeer N, Parmar N *et al.* (2017), 'Attention is all you need', *Conference and Workshop on Neural Information Processing Systems*.
62. Guo S, Wang Y, Yuan H, *et al.* TAERT: triple-attentional explainable recommendation with temporal convolutional network. *Inform Sci* 2021;**567**:185–200.
63. Kaiming H, Xiangyu Z, Shaoqing R *et al.* Deep residual learning for image recognition, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 2016:770–8.
64. Ba JL, Kiros JR, Hinton GE. Layer normalization [arXiv]. *arXiv* 2016;**14**.
65. Zhang H, Cisse M, Dauphin YN, *et al.* Mixup: beyond empirical risk minimization *[ArXiv]*. 2017.
66. Lin T, Goyal P, Girshick R, *et al.* Focal loss for dense object detection. *IEEE Trans Pattern Anal Mach Intell* 2020;**42**:318–27.
67. Hendrycks D, Gimpel K. Gaussian error linear units (GELUs) *[ArXiv]*. 2020.
68. Liu L, Jiang H, He P, *et al.* On the variance of the adaptive learning rate and beyond *[ArXiv]*. 2019.
69. Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift *[ArXiv]*. 2015.