A Drug-Target Interaction Prediction Based on NV-DNN Learning

Xiaodan Wang[1], Jihong Wang[2], and Jianhui Wang[1]
[1]School of Pharmaceutical Chemistry and Chemical Engineering, Guangdong Pharmaceutical University, Zhongshan, Guangdong, China
[2]School of Data and Computer Science, Sun Yat-Sen University, Guangzhou, Guangdong, China
Email: redblue04@163.com, wangxiaodan@gdpu.edu.cn, 549323028@qq.com

***Abstract*—Drug Target Interactions (DTIs) prediction research is one of the key links in drug development, and is of great significance to the fields of new drug development and drug relocation. In recent years, network representation learning technology has developed rapidly. Network representation learning is also known as Graph Embedding, which can be used in applications such as node classification, link prediction, and community discovery. The key challenge in this field is to effectively vectorize complex homogeneous or heterogeneous networks, and how to fully reflect the network structure, node relationships and connection information. Deep learning has made great achievements in the fields of speech recognition, computer vision, and natural language processing. There are also many research results in the field of DTIs prediction, but the combination of the two is relatively small, and it is worthy of in-depth study. This paper proposes DTIs prediction based on NV-DNN method. The drug-drug and target-target relationships are used to form the Jaccard matrix. The Node2Vec method is used to learn the feature vector representation, and then input it into the DNN network for deep learning to improve classification and prediction capabilities. Experiments show that AUC 0.89 is better than other common methods that only use the network structure information of drugs and targets, and do not use the attribute information. Due to the simple input of the model, it can predict drugs and targets with unknown attributes, so it has a wide range of adaptability.**

***Index Terms*—drug-target interaction prediction, network representation learning, Node2Vec, DNN**

I. Introduction

Drug R&D is a systematic project, facing the challenges of high cost, long cycle and low success rate [1], [2]. In bioinformatics research, pharmaceutical research is one of the most widely used and valuable fields of application. Drug research is expensive and a long process. It is reported that only one compound in every 10,000 new chemical entities can eventually become a drug, and the entire process takes more than ten years and costs more than US $800 million [3].

In recent years, with the continuous development of molecular biology technology, high-throughput sequencing technology and nuclear magnetic resonance technology of testing methods [4], many protein targets related to disease occurrence have been discovered and can be designed to design drug molecular structures. At the same time, along with big data, cloud computing, machine learning, deep learning and other intelligent computer technologies have been widely used. As an aid to drug design, Computer-Aided Drug Design (CADD) and drug screening have become increasingly important, and have penetrated into all aspects of new drug development [5].

According to the theory of applying computer technology to assist the design of drugs, the small molecules of drug compounds mainly include enzymes, nuclear receptors, G-protein coupled receptors and ion channels. Channel and other biological macromolecules interact, and then use computer-aided drug design, and hope to use computer simulation, calculation and related prediction techniques to study the relationship between drug compound molecules and target proteins [6], from the perspective of informatics, based on systems biology and machine learning related methods, integrate various data sources, predict candidate drug-target interactions, and screen out drug-target pairs that meet the requirements, thereby improving efficiency, reducing costs, and reducing drug development. Risk of failure in the process. Therefore, computer-assisted drug screening to improve the screening success rate has very important research significance.

In recent years, network representation learning technology has developed rapidly, and network representation learning is also called graph embedding (Graph Embedding). They range from bioinformatics drugs, target interactions, and interest recommendations in social networks. Network representation can be used for applications such as node classification, link prediction, and community discovery. The key challenge in this field is to effectively vectorize complex homogeneous or heterogeneous networks, which can fully reflect the network structure, node relationships, and connection information. The vectors represented by the network can be used as input for machine learning. Most of them use heuristics to extract network features and perform structured coding [7], such as the artificial PPIN network constructed by Li [8] and others. Network representation learning can be directly supervised or unsupervised, and can also be combined with other algorithms to serve as the feature representation tool of the latter. The generated

Manuscript received June 11, 2021; revised August 23, 2021.

doi: 10.18178/ijpmbs.10.4.135-141

topological feature vector is used as the input of subsequent machine learning algorithm, so as to achieve node classification and link prediction tasks [9]-[12]. As one of the most commonly used deep learning methods, Deep Neural Networks (DNN) [13] has strong generalization. In many scenarios, it can get better learning ability than tree model. Therefore, DNN is used as a classifier for learning.

II. Materials and Methods

A. DataSets

The data source of drug-target network prediction research comes from public databases. Generally, a heterogeneous network is constructed with drugs, targets, diseases, and side effects. Drug information is extracted from the DrugBank [14] database, target protein information is extracted from the HPRD [15] database, disease information is extracted from the comparative toxicology genomics [16] database, and drug side effects information is extracted from the SIDER [17] database. Here, in order to study the predictive ability of the simple ID-type feature network, we only use the drug-drug relationship matrix and the target-target relationship matrix to model, and the label is represented by whether there is a drug-target interaction. The number of samples obtained after the final processing and screening is 708 drugs, 1512 targets, and 1923 Drug-Target Interactions (DTI), so the number of positive samples used for training is 1923 pairs. As shown in Table I.

TABLE I. Details of the Number of Samples in the Experimental Data Set

	Drug	Target	DTI
Quantity	708	1512	1923

B. Data Preprocessing

In this paper, after constructing each network extracted from the database into an adjacency matrix. In this paper, only the network nodes that need to obtain drugs and targets are expressed, and the remaining node types are only used for intermediate relationship inheritance. Therefore, in order to reduce the node types in the network to be suitable for the migration process of each network in the next stage, for each network adjacency matrix takes drugs and target nodes as a set, calculates the Jaccard similarity coefficient between each node in the network, and then generates the corresponding Jaccard similarity network.

Separately study the similarity network of drugs and targets to obtain independent drug network features and target network features, and the data input when training a binary classification model is DTI feature. Therefore, after obtaining the network characteristics of the drug target, we need to splice and construct the drug target pair characteristics according to the needs of the sample. The DTI feature is mainly composed of the drug feature and the target feature. The last dimension represents the label of the DTI pair. If the DTI pair exists, that is, the current drug will act on the target, the label takes the dimension 1 and the current sample is positive sample. Otherwise, the value of the label is 0, indicating that the sample is a negative sample. Therefore, we can convert the problem of predicting the relationship between drug targets to the binary classification problem of judging whether the current drug will have a relationship with the target.

During the experimental sampling, we selected the existing 1923 drug target relationship pairs with interaction relationship as positive samples, and randomly combined equal amounts of drug target pairs with no interaction relationship found as negative samples. Make the ratio of positive and negative samples in the training data set reach 1: 1 to improve the accuracy of model training.

C. Node2Vec

Aditya [18] *et al.* Proposed Node2Vec in 2016, an algorithmic framework for learning the continuous feature representation of each node in a continuous network. Node2Vec can learn a mapping and map the nodes in the network to the low-dimensional feature vector space, while maximizing the preservation of the neighbor relationship between the nodes in the original network.

As Perozzi's [12] DeepWalk obtains the network representation of each node in the graph through special modeling. The idea of Node2Vec algorithm is very close to DeepWalk, but it is a targeted improvement of DeepWalk. It defines a very flexible concept for the neighbor relationship in the network, and then searches for the neighbor relationship in the network more efficiently, A semi-supervised random walk strategy with bias is proposed, that is, the probability of each node going down to a different node is different. The node sequence obtained after random walk is consistent with DeepWalk, and still uses Skip-Gram algorithm to train.

If the problem of searching for the neighbor relationship of a node in the network is regarded as a graph local search problem, then there are two common search strategies: depth first search (DFS) and breadth first search (BFS). as shown in Fig. 1.

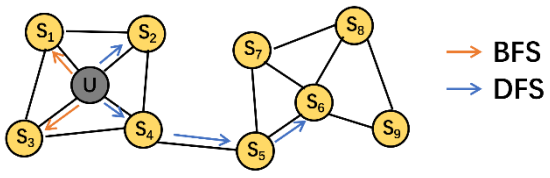

Figure 1. BFS and DFS search strategy.

In the prediction task of network nodes, two types of equivalence usually occur: homogeneity (Homophily) equality and structural (Structural) equality [19]. In the homogeneity hypothesis [20], [21], the nodes are highly interconnected, and all belong to a similar network group or their groups should be embedded very closely together. In the structural equality hypothesis [22], if two nodes do not belong to the same network group, but they have similar network structure roles in the network, then they should also be embedded together.

Through observation, it is found that the two strategies of BFS and DFS play a very crucial role in generating the node representation and reflecting the above two types of equality. When the neighbor nodes obtained by BFS sampling are embedded, it is related to structural equality. In DFS, the sampled nodes accurately reflect the relationship between the leaders from a macro perspective, which is necessary to infer the network group based on homogeneity. Therefore, Node2Vec has designed a more flexible, random walk strategy with offset between BFS and DFS that is very stable.

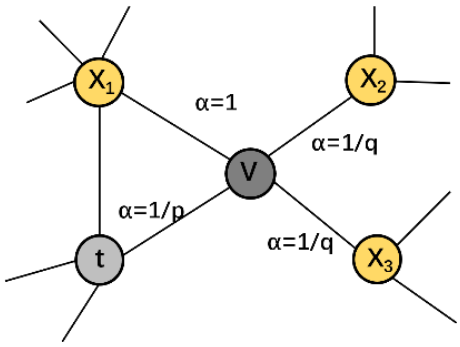

Figure 2. Andom walk strategy with bias.

Random walk with offset is a second-order strategy guided by two parameters p and q, as shown in Fig. 2: Suppose the process of a random walk is from point t to point v, and then from v Start the next step of the walk. The transition probability π_{vx} between the node x adjacent to the point v depends on the shortest path distance d_{tx} between the point x and the point t in the previous step. That is, $\pi_{vx} = a_{pq}(t,x) * w_{vx}$, and

$$\alpha_{pq}(t,x) = \begin{cases} \frac{1}{p} & if\ d_{tx} = 0 \\ 1 & if\ d_{tx} = 1 \\ \frac{1}{q} & if\ d_{tx} = 2 \end{cases}$$

where w_{vx} is the weight of the edge (v,x) in the network, d_{tx} represents the shortest path distance between the point x and the point t in the previous step, that is, the value of d_{tx} can only be one of $\{0,1,2\}$. p and q control the speed of visiting and leaving neighbor nodes during the walk. p is the return parameter, which controls the possibility of directly returning to the previous node, and q is the access parameter, it can distinguish between internal nodes and external nodes when traveling, if $q > 1$, the closer the walk is to BFS, the closer to t the greater the probability of accessing the node, if $q < 1$, the walk is closer to DFS, and the probability of visiting nodes further away from point t is small.

D. Deep Neural Network

Deep Neural Network (DNN) is also called Multi-Layer Perceptron (MLP), which can be understood as a neural network with many hidden layers. DNN greatly improves the recognition rate by increasing the number of hidden layers on the basis of ordinary neural networks. Zhixiang Wang et al. Used the DNN model to predict potential adverse drug reactions, and achieved good results [23]. In this experiment, a deep learning framework tensorflow [24] is used to construct a deep neural network.

Tensorflow is a second-generation intelligent learning system created by the Google Brain team. It is an open-source library that uses data flow graphs for numerical calculation and large-scale machine learning. Tensorflow can be implemented by Python code. Tensorflow can train and run neural networks, which are mainly used in Deep Neural Networks (DNN), Convolutional Neural Networks (CNN) [25], Recurrent Neural Networks (RNN) [26] and other deep learning fields.

E. Node2Vec-Deep Neural Network (NV-DNN) Model

A flowchart of the NV-DNN model is illustrated in Fig. 3, and the steps are follows:

Retrieve the interactions of the drug and target. Two groups of interaction pairs with drugs and targets were respectively constructed. The Jaccard similarity network was generated by calculate Jaccard's similarity coefficient, and then the drug and target eigenvector representation was obtained by using Node2Vec. The vector feature and the drug-target relationship are spliced to construct equilibrium samples. Utilize the DNN algorithm to construct a prediction model based on the training set by 10-fold cross-validation.

F. Comparison with Other Methods

The principle of K-Nearest Neighbor (KNN, K-Nearest Neighbor) is to achieve the classification effect by measuring the distance between different feature values and identifying the distance between different feature values. When importing samples of unknown categories into the model, if the samples are most similar to the existing K in the feature space (most feature spaces are closest), most samples belong to a certain category. In the KNN algorithm, the selected adjacent samples are all correctly classified objects [27]. When adding a new sample, the new sample will be classified according to its principle. After classification, the position of the sample and its feature vector are recorded as the criteria for classifying subsequent samples. When there are multiple known feature samples near the new sample, KNN will use the principle of minority majority in the classification decision to classify the new sample using the category of more samples in the same range nearby.

Like the neural network algorithm, the decision tree is an algorithm that can solve both classification and regression problems. The model structure is a tree. The decision tree classifies the sample nodes of the same class as leaves by learning the training set. Otherwise, the decision tree selects the feature vector with the optimal splitting ability of different types of samples as the decision node according to the current decision node attributes. For different values, the training set samples are divided into several subsets, each value forms a branch, and several values form several branches. For the obtained subset, the previous steps are repeated, and the nodes recursively form a decision tree on each divided sample to form the basic structure of the model.

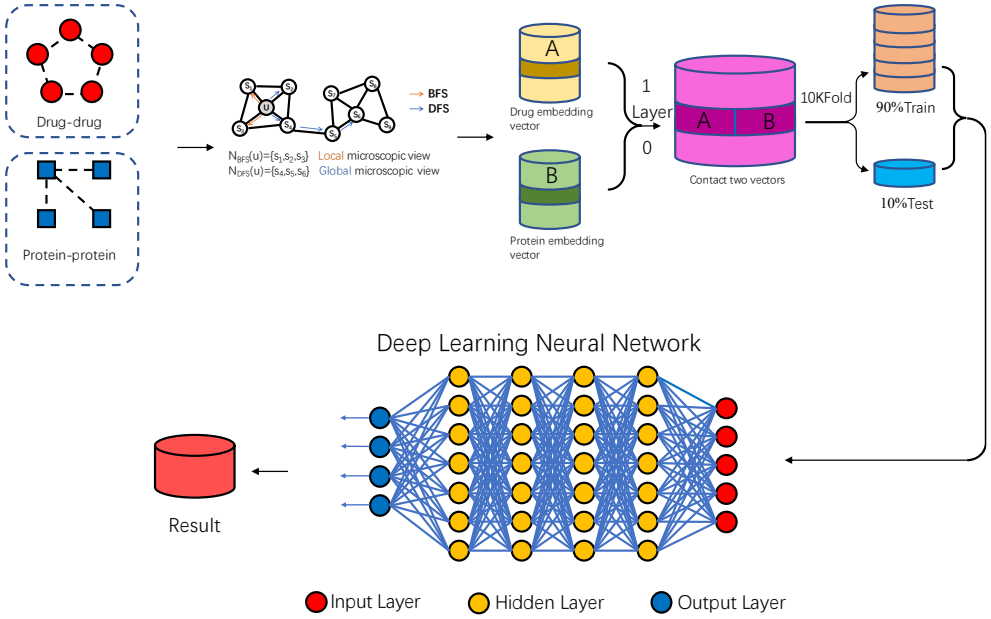

Figure 3. Flowchart of the NV-DNN model.

The forest structure in the Random Forest (RF) is actually composed of multiple decision trees. The main principle is to select multiple samples and features of the same size by replacing samples with the boostrapping method, and generate multiple decision trees to form a random forest. RF uses cart decision tree as weak learner to generate multiple trees randomly, and the final prediction is based on voting of all trees [28].

III. Results

A. Parameter Settings

Learning rate: The learning rate indicates the magnitude of each parameter update in the neural network. The learning rate is large and the learning speed of the model is fast, but the loss value is easy to oscillate around the minimum value and cannot converge to the minimum Value; the learning rate is low, the learning speed of the model is slow, and the convergence speed is relatively slow. The model can learn better, but it is easy to cause over-fitting problems. Therefore, choosing an appropriate learning rate is very important for the training effect of the model. By comparing the results obtained by using different learning rates for the same data and keeping other parameters the same, as shown in Table II, the learning rate of 0.001 is the best in this experiment.

TABLE II. Comparison of Results of Different Learning Rates

Learning rate	AUC	PRE
0.01	0.84	0.82
0.005	0.85	0.86
0.001	0.89	0.87
0.0001	0.86	0.86

Activation function: The activation function can perform non-linear transformation on each node in the neural network. The commonly used activation functions are mainly sigmoid, relu and tanh. In general, the calculation speed of the relu function is much faster than the two activation functions of sigmoid and tanh. In the process, the gradient disappears, so the relu function is currently the most common activation function used in artificial neural networks. However, when the input value is negative, some neurons cannot easily be activated. In this experiment, after comparison and verification, as shown in Table III, the results of the sigmoid function running in this experiment are in three kinds of activation The effect in the function is the best.

TABLE III. Comparison of Results of Different Activation Functions

Activation function	AUC	PRE
sigmoid	0.89	0.87
tanh	0.84	0.81
relu	0.85	0.84

Optimizer: The optimizer in the neural network is similar to the catalyst in the chemical reaction. When the neural network structure reaches a certain degree of complexity and the amount of data is relatively large, the time spent training the network model will greatly increase, in order to be able to train the model at a faster speed and does not affect the effect of the loss value gradient drop when training the model, you need to use an optimizer to optimize the training of the neural network. Common neural network optimizers in Tensorflow mainly include GradientDescent optimizer, Adagrad optimizer, Adam optimizer, and RMSProp optimizer. Through the comparison of the results of several optimizers, as shown in Table IV, it is found that Adam and RMSProp have similar effects. Found that Adam's effect is generally better than RMSProp, so I chose Adam as the optimizer in the model.

TABLE IV. EFFECT COMPARISON OF DIFFERENT OPTIMIZERS

optimizers	AUC	PRE
Adam	0.89	0.87
RMSProp	0.88	0.86
GradientDescent	0.80	0.79
Adagrad	0.78	0.79

Overfitting: The problem of overfitting is that the machine learning model learns too many features of the training set during feature learning on the samples of the training set, causing the model to predict the test set When there is a big difference between the predicted effect and the real value. In order to solve the problem of overfitting in neural networks, it mainly includes increasing the number of samples, dropout, and regularization processing. Since the data taken in the experiment is equal to the number of positive and negative samples, it is not easy to increase the sample size, so the two methods of dropout and regularization are used.

- Dropout: The use of dropout to solve the overfitting problem in neural networks was first proposed by Hinton *et al.* In 2012 [29]. Dropout processing means that during the training of the neural network model, certain neurons in the hidden layer are randomly hidden in the propagation at the set ratio for the time being. They will not participate in the work. The weights will not be updated but will be retained. It reappears during the propagation, and continues to participate in the propagation training of the neural network to update the weights, while the other neurons are hidden according to the set ratio. The weights will not be updated but will still be retained. The use of dropout increases the generalization ability of the deep neural network model. Its use can make the deep neural network not dependent on some local features, thereby reducing the impact of overfitting.
- Regularization: An important reason for the formation of overfitting is that the function considers every point too much when the model is fitted, thereby forming a fitting function with large fluctuations. By using the regularization method, the fluctuation of the function can be restricted, the complexity and instability of the model can be reduced, and the situation of overfitting can be reduced to a certain extent. Commonly used regularizations include L1 regularization and L2 regularization. In general, L2 regularization can make the model convergence more stable and faster, and there is no problem of matrix invertibility when using the equation to minimize the loss function. Therefore, in this experiment, L2 regularization was selected as the regularization increase term of the loss function to use.

The comparison of the running results of the code in the experiment shows that after using dropout and regularization to place the model overfitting, the model's effect has been effectively improved, and the AUC value has been increased from about 0.84 to about 0.89. The model uses 10-fold cross-experience, the specific results are shown in Table V:

TABLE V. NV-DNN TEST RESULTS

ID	AUC	PRE
01	0.884	0.853
02	0.902	0.872
03	0.893	0.881
04	0.878	0.79
05	0.882	0.850
06	0.904	0.888
07	0.898	0.863
08	0.886	0.871
09	0.891	0.874
10	0.886	0.861

By calculating the average AUC value of the ten times results is about 0.890, the average Precision value is about 0.866, indicating that the deep neural network is used as a drug target prediction with high confidence. This result is compared with some commonly used machine learning classification models to verify the effect of the method.

B. Model Comparison with Other Classifiers

The method comparison needs to use the same data for testing. Here all use the same parameters for Node2Vec representation learning, and then compare the learning capabilities of several different classifiers. Here, the commonly used KNN, DT, RF methods are compared with DNN. The prediction results are shown in Table VI:

TABLE VI. NV-DNN TEST RESULTS

MODEL	AUC	PRE
NV-KNN	0.796	0.712
NV-DT	0.712	0.684
NV-RF	0.881	0.816
NV-DNN	0.890	0.866

From the comparison of the effects of the various models, it can be seen that the AUC and AUPR of NV-DNN have achieved the best results, followed by NV-RF, and the effects of NV-KNN and NV-DT models are relatively poor. It shows that NV-DNN has the strongest ability to predict drug-target interactions. The fusion model of network representation learning and deep learning can improve the learning ability of the model.

IV. DISCUSSION

In this paper, the Node2Vec method is used to perform feature representation learning on the topological relationship between drugs and target networks. After splicing into drug-target pairs, several typical classification algorithms are used to perform classification learning and observe their respective effects.

The K-Nearest Neighbor (KNN) algorithm is relatively simple and easy to understand compared to other models,

and it is also relatively powerful in classification learning. However, in training and learning, KNN needs to traverse all data for each classification, which is not friendly for the prediction of drug target interactions that often perform big data processing tasks. At the same time, the KNN algorithm has poor fault tolerance for the training set. Since KNN is classified by calculating the distance of neighbor nodes and the labels of neighbor nodes, this leads to if one or two of the neighbor nodes are wrong. Will result in inaccurate predictions. In this experiment, the KNN with the optimal parameters was used. The AUC value in drug target prediction research failed to reach more than 0.8. When predicting the characteristics of some drug targets with different positive and negative sample ratios, if you want to choose a model for prediction research, you need to the model was further improved.

Decision Tree algorithm (DT) will consider all feature vectors as leaf nodes when constructing the tree structure, so in the calculation, you only need to split along the root of the tree to determine the leaf is the only classification, which makes Learn to predict fast. Due to the large number of feature vectors, the decision tree has many split rules, and the model learns too many features, which leads to a decrease in model accuracy and overfitting. In this experiment, the result obtained by the decision tree model is the lowest of several machine learning algorithm models, with an AUC of only 0.71, which is lower in confidence for drug target prediction.

Based on the decision tree, multiple random decision trees are randomly generated to classify the data, and then a Random Forest algorithm (RF) is generated. RF is a typical example of superposition of multiple weak classifiers to generate a strong classifier. It has the advantage of fast training speed of the decision tree algorithm. At the same time, because of its own two randomness, don't worry too much about overfitting. In the face of unbalanced data sets, it can also achieve the effect of balancing errors, and it can still maintain a certain accuracy even on data with missing features. In this experiment, the AUC value predicted using RF reached 0.88, which is the second DNN classification method.

In this paper, the deep learning algorithm selects DNN. From the experimental results, based on the same network representation of the learned feature vectors, DNN has achieved the best results, indicating that the prediction of DTIs based on the NV-DNN model is advanced.

However, from the perspective of forecasting indicators, there should be more room for optimization. Deep learning usually requires large-scale data to learn fully, and the drug-target pairs with interactions studied in this article are only 1923 pairs. They are regarded as positive samples. In order to ensure the consistency of the number of positive and negative samples, each data is randomly extracted There are only 1923 pairs of drug-target pairs with no interaction, so the total number of samples in each data is 3846, and the samples are too few to meet the training needs of machine learning. Therefore, in the follow-up research, it is necessary to try to increase the size of the data sample appropriately to improve the learning ability of the model. In the future, we can continue to study more advanced network representation learning algorithms, such as line, gcn etc., through advanced algorithms to extract more hidden information in the network; the current research is based on the formation of a node structure network, which can be based on a structured network, enrich the content information of the nodes, realize the fusion of the structure and content of the network, play a complementary role and enhance the expression ability of the network; in the deep learning classification algorithm, the latest attention mechanism can be introduced to improve through the attention deep learning model prediction ability.

V. Conclusion

Network representation learning is a very effective feature representation learning method that can effectively vectorize complex homogeneous or heterogeneous networks, and fully reflect the network structure, node relationships and connection information. Deep learning has been widely used in many fields, and has also achieved many results in the field of bioinformatics. The NV-DNN-based method proposed in this paper is used to form a Jaccard matrix through drug-drug and target-target relationships. Then use the Node2Vec method to get the feature vector representation, and finally input it into the DNN network for deep learning. Thereby, a classification model is obtained. Experiments have proved that this method has strong classification and prediction ability. Due to the simple input of the model, no attributes such as drugs and targets are needed, and it has good generalization ability.

Conflict of interest

The authors declare no conflict of interest.

Author Contributions

Xiaodan Wang was responsible for the creative, design model and guidance of the thesis. Jianhui Wang completed the experiment and Jihong Wang completed the final writing of the thesis.

Acknowledgment

This work was supported by the special fund No. 51661001 of Special funds of key disciplines construction from Guangdong and Zhongshan cooperating.

References

[1] M. Shea, L. Ostermann, R. Hohman, *et al.*, "Regulatory watch: Impact of breakthrough therapy designation on cancer drug development," *Nature Reviews Drug Discovery*, vol. 15, no. 13, p. 152, 2016.

[2] J. Settleman and R. L. Cohen, "Communication in drug development: 'Translating' scientific discovery," *Cell*, vol. 164, no. 6, pp. 1101-1104, 2016.

[3] S. Hutchins, T. Torphy, and C. Muller, "Open partnering of integrated drug discovery:continuing evolution of the pharmaceutical mode," *Drug Discovery Today*, vol. 16, no. 7-8, pp. 281-283, 2011.

[4] E. Freyhult, M. G. Guastafsson, and H. Strömbergsson, "A machine learning approach to explain drug selectivity to soluble and membrane protein targets," *Molecular Infomatics*, vol. 34, no. 1, pp. 1101-1104, 2015.
[5] S. Chen, Y. Zhu, D. Zhang, *et al*., "Feature extraction approaches based on matrix pattern: MatPCA and MatFLDA," *Pattern Recognition Letters*, vol. 26, no. 8, pp. 1157-1167, 2005.
[6] L. Wu, Y. Shen, M. Li, *et al.*, "Network output controllability-based mehtod for drug target identification," *IEEE Transactions on Nanobioscience*, vol. 14, no. 2, p. 184, 2015.
[7] W. L. Hamilton, R. Ying, and J. Leskovec. (Sep. 2017). Representation learning on graphs: Methods and applications. *IEEE Computer Society Technical Committee*. [Online]. Available: https://arxiv.org/abs/1709.05584
[8] Z. C. Li, W. Q. Zhong, Z. Q. Liu, *et al.*, "Large-scale identification of potential drugtargets based on thetopological features of human protein-protein interaction network," *Analytica Chimica Acta*, vol. 871, pp. 18-27, 2015.
[9] C. Knox, V. Law, T. Jewison, *et al.*, "DrugBank 3.0: A comprehensive resource for 'Omics' research on drugs," *Nucleic Acids Research*, vol. 39, pp. D1035-D1041, 2011.
[10] W. Ba-Alawi, O. Soufan, M. Essack, *et al.*, "DASPfind: New efficient method to predict drug-target interactions," *Journal of Cheminformatics*, vol. 8, no. 1, pp. 1-9, 2016.
[11] D. Zhang, J. Yin, X. Zhu, *et al.*, "Network representation learning: A survey," *IEEE Transactions on Big Data*, vol. 6, no. 1, pp. 3-28, 2020.
[12] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," in *Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 701-710.
[13] M. Wen, Z. Zhang, S. Niu, *et al.*, "Deep-learning-based drug-target interaction prediction," *Journal of Proteome Research*, vol. 16, no. 4, pp. 1401-1409, 2017.
[14] C. S. Wang, P. J. Lin, C. L. Cheng, S. H. Tai, Y. H. K. Yang, and J. H. Chiang, "Detecting potential adverse drug reactions using a deep neural network model," *Med. Internet Res.*, vol. 21, no. 2, article e11016, 2019.
[15] T. S. K. Prasad, R. Goel, K. Kandasamy, *et al.*, "Human protein reference database—2009 update," *Nucleic Acids Research*, vol. 37, pp. 767-772, 2009.
[16] A. P. Davis, B. L. King, S. Mockus, *et al.*, "The comparative toxicogenomics database: Update 2013," *Nucleic Acids Research*, vol. 41, pp. D1104-D1114, 2013.
[17] M. Kuhn, M. Campillos, I. Letunic, *et al.*, "A side effect resource to capture phenotypic effects of drugs," *Molecular Systems Biology*, vol. 6, no. 1, p. 343, 2010.
[18] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proc. the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 855-864.
[19] P. D. Hoff, A. E. Raftery, and M. S. Handcock, "Latent space approaches to social network analysis," *Journal of the American Statistical Association*, vol. 97, no. 460, pp. 1090-1098, 2002.
[20] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3, pp. 75-174, 2009.
[21] J. Yang and J. Leskovec, "Overlapping communities explain core–periphery organization of networks," *Proceedings of the IEEE*, vol. 102, no. 12, pp. 1892-1902, 2014.
[22] K. Henderson, B. Gallagher, T. Eliassi-Rad, *et al.*, "RolX: Structural role extraction & mining in large graphs," in *Proc. the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2012, pp. 1231-1239.
[23] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *Proc. the 26th International Conference on Neural Information Processing Systems*, 2013, pp. 2553-2561.
[24] M. Abadi, *et al.*, "Tensorflow: A system for large-scale machine learning," in *Proc. the 12th USENIX Symposium on Operating Systems Design and Implementation*, Savannah, 2016, pp. 265-283.
[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84-90, 2017.
[26] J. Chung, *et al.*, "Empirical evaluation of gated recurrent neural networks on sequence modeling," presented in NIPS 2014 Deep Learning and Representation Learning Workshop, 2014.
[27] D. Cao, L. Zhang, G. Tan, *et al*., "Computational prediction of drug target interactions using chemical, biological, and network features," *Molecular Informatics*, vol. 33, no. 10, pp. 669-681, 2014.
[28] J. P. Mei, C. K. Kwoh, P. Yang, *et al.*, "Drug-target interaction prediction by learning from local information and neighbors," *Bioinformatics*, vol. 29, no. 2, pp. 238-245, 2013.
[29] G. E. Hinton, N. Srivastava, A. Krizhevsky, *et al.* (2012). Improving neural networks by preventing co-adaptation of feature detectors. *Computer Science*. [Online]. Available: https://arxiv.org/abs/1207.0580

Xiaodan Wang is Lecturer at School of Medicine and Chemical Engineering, Guangdong Pharmaceutical University. The main research contents include the influence of polymer materials on its form and drug effect in the process of drug delivery, including the synthesis and application of polymer prodrugs, as well as the research of related medicinal polymer materials; The machine learning methods and big data were used to predict the drug-target relationship, protein-protein relationship prediction, etc.