Mathematical Biosciences and Engineering

MBE, 20(8): 14136−14157.
DOI: 10.3934/mbe.2023632
Received: 03 May 2023
Revised: 31 May 2023
Accepted: 11 June 2023
Published: 26 June 2023

http://www.aimspress.com/journal/MBE

Research article

Inferring drug-disease associations by a deep analysis on drug and disease networks

Lei Chen[1,*], Kaiyu Chen[1] and Bo Zhou[2]

[1] College of Information Engineering, Shanghai Maritime University, Shanghai 201306, China
[2] Shanghai University of Medicine & Health Sciences, Shanghai 201318, China

* **Correspondence:** Email: chen_lei1@163.com; Tel: +008621382828251; Fax: +00862138282800.

Abstract: Drugs, which treat various diseases, are essential for human health. However, developing new drugs is quite laborious, time-consuming, and expensive. Although investments into drug development have greatly increased over the years, the number of drug approvals each year remain quite low. Drug repositioning is deemed an effective means to accelerate the procedures of drug development because it can discover novel effects of existing drugs. Numerous computational methods have been proposed in drug repositioning, some of which were designed as binary classifiers that can predict drug-disease associations (DDAs). The negative sample selection was a common defect of this method. In this study, a novel reliable negative sample selection scheme, named RNSS, is presented, which can screen out reliable pairs of drugs and diseases with low probabilities of being actual DDAs. This scheme considered information from k-neighbors of one drug in a drug network, including their associations to diseases and the drug. Then, a scoring system was set up to evaluate pairs of drugs and diseases. To test the utility of the RNSS, three classic classification algorithms (random forest, bayes network and nearest neighbor algorithm) were employed to build classifiers using negative samples selected by the RNSS. The cross-validation results suggested that such classifiers provided a nearly perfect performance and were significantly superior to those using some traditional and previous negative sample selection schemes.

Keywords: drug-disease association; drug repositioning; negative sample selection; network embedding; binary classification; random forest

1. Introduction

Diseases are a core issue of human health, has and have existed since the emergence of human beings. A large number of people die from various diseases every year. The problem of how to treat different diseases is always a hot topic in medical science. Numerous efforts have been made over the years, especially in the past 100 years. Plenty of schemes have been designed to treat diseases. Among them, drug is considered to be one of the effective ways. However, it is not easy to develop a new drug, which always requires some rigorous and complex steps; drug development is a long procedure and is always very expensive. According to relevant reports, the average time for designing a new drug is about 10−15 years [1] and can cost up to 802 million dollars [2]. Although the investment on drug development has sharply increased in these years, the number of drug approvals each year remains quite low. Designing new techniques for accelerating the procedures of drug development remains quite urgent.

Drug repositioning is deemed as an alternative pipeline, which can promote drug development procedures. For most existing drugs, our cognizance is not very complete, as some latent effects have not been discovered. The purpose of drug repositioning is to discover the latent effects of existing drugs, thereby discovering the new diseases that these drugs can treat. Because numerous clinical tests have been conducted on existing drugs, the launch of these drugs for treating new diseases can be evidently accelerated. However, it is still laborious to find out and confirm the new effects of existing drugs. The use of computational methods is an effective alternative procedure, which has become quite popular [3−7].

In recent years, lots of computational methods have been designed for drug repositioning. Many of these methods focused on the prediction of drug-disease associations (DDAs). The validated DDAs were deeply analyzed and some special patterns were discovered, which can be used to identify latent DDAs. Previous studies have modeled such prediction problems as a recommender system [8−12]. These methods always set up one or more kernels of drugs, diseases, or drugs and diseases. Some complex fusion procedures were applied to these kernels, thereby scoring each pair of drugs and diseases. Network-based methods are another group for predicting DDAs. They always construct multiple networks, which contain not only drugs and diseases but also other related objects, such as long non-coding RNAs (lncRNAs), micro RNAs (miRNAs), target proteins, etc. Among these methods, some can be directly applied to networks to make predictions [13−19], whereas others can adopt networks to access features of drugs and diseases, and the downstream classification algorithms complete the prediction task [20−25]. Recently, deep learning algorithms have been used to construct methods for predicting DDAs, such as convolutional neural networks [26] and graph convolutional networks [27,28]. Several computational methods designed binary classifiers to predict DDAs. The validated DDAs were retrieved from public databases, which were termed as positive samples. However, the selection of negative samples was a problem. Some studies adopted random selection to pick up the same number of negative samples from unlabeled pairs of drugs and diseases [21,26,27]. As latent DDAs may be included, such selection may lead to an unstable decision boundary of the following constructed classifiers. This study gave a contribution in this regard.

In this study, a novel reliable negative selection scheme, named RNSS, was proposed, which can help us select reliable negative samples (i.e., pairs of drugs and diseases with low probabilities of being actual DDAs). Such a scheme employed the k-neighbors of a drug in a drug network and evaluated the relationships between one drug and one disease based on the relationships between k-neighbors and

the disease. Three classic classification algorithms were adopted to construct classifiers using negative samples generated by an RNSS: random forest (RF) [29], bayes network (BN), and the nearest neighbor algorithm (NNA) [30]. The results indicated that the classifiers with the RNSS provided a nearly perfect performance and were much better than those with traditional negative selection schemes, indicating that the RNSS can genuinely screen out reliable negative samples.

2. Materials and methods

2.1. Data source

The validated DDAs were directly accessed from a previous study [21]. These interactions were extracted from chemical-disease interactions collected in the Comparative Toxicogenomics Database (CTD) (http://ctdbase.org) [31−33]. In detail, the file "CTD_chemicals_diseases.csv.gz" in CTD was downloaded, from which the chemical-disease associations with "DirectEvidence" were extracted. Then, the chemical-disease associations without DrugBank IDs were discarded. Approximately 63,472 associations remained, which were deemed as positive samples in this study. Approximately 2,794 drugs (represented by DrugBank IDs) and 3,019 diseases (represented by MESH or OMIM identifiers) were involved in the positive samples.

Generally, negative samples were necessary to build the binary classification model. However, the selection of negative samples was a challenging problem as the unlabeled pairs of drugs and diseases may be latent DDAs. Some previous studies adopted random selection to construct negative samples. Here, we proposed a scheme to select reliable negative samples. Based on such a scheme, negative samples that were one, two, and three times as many positive samples were selected and combined with positive samples to constitute datasets.

2.2. Reliable negative sample selection

In this study, we tackled the prediction of DDAs by modeling a binary classification problem. From the CTD, the validated positive samples were obtained as mentioned in Section 2.1. However, there are no public databases that collect the negative samples for DDAs due to a lack of their application values [26]. Several previous studies adopted a random selection scheme to generate negative samples, which may induce an unstable decision boundary of the classifier [34]. Thus, it is necessary to design an efficient scheme to select reliable negative samples. These samples should have a low likelihood of being actual DDAs. This section introduced a novel scheme to select reliable negative samples, named RNSS. Its procedures are described below.

First, a drug network was constructed, which defined 2,794 drugs as nodes. The associations between drugs should be determined, thereby defining the edges within the network. It is known that the Simplified Molecular Input Line Entry System (SMILES) [35] format is the most widely used representations of drugs, from which the fingerprints of drugs can be extracted. Here, the RDKit (http://www.rdkit.org/) was adopted to extract the extended-connectivity fingerprint (ECFP) of each investigated drug. For drug p, its fingerprints constitute a set, denoted by $F(p)$. The Tanimoto coefficient was applied on the fingerprint sets of two drugs, p_1 and p_2, to measure their associations, formulated by

$$Q^f(p_1, p_2) = \frac{|F(p_1) \cap F(p_2)|}{|F(p_1) \cup F(p_2)|}. \quad (1)$$

Two nodes in the network were connected by an edge if and only if the association between their corresponding drugs was larger than zero. In addition, each edge e was assigned a weight, denoted by $w(e)$, which was the association between two drugs. Such a network is denoted by W_d.

It is known that drugs in similar structures are more likely to treat similar diseases. In view of this, the relationships between a drug, denoted by p_i, and a disease, denoted by q_j, can be measured by the relationships between similar drugs of p_i and q_j. In W_d, these drugs are the direct neighbors of p_i, (i.e., the 1-neighbors of p_i). Furthermore, the drugs with a distance two to p_i (2-neighbors of p_i) may also provide contributions, also for the k-neighbors of p_i ($k > 2$). In view of this, the k-neighbors of p_i was picked up from W_d, denoted by $N_k(p_i)$, which consisted of drugs with distance k to p_i. For each drug p in $N_1(p_i)$, the association between it and p_i (i.e., the weight on the edge connecting them) can be directly used to measure the relationship between p_i and q_j. However, it is problematic to utilize drugs in $N_k(p_i)$ ($k > 1$), as these drugs have no direct associations with p_i. To settle such a problem, the weight of a path must be defined. For a path P with length l, containing edges $e_1, e_2, \cdots, e_l$, its weight, denoted by $w(P)$, was defined as

$$w(P) = (\prod_{i=1}^{l} w(e_i))^{F_{decay}(P)}, \quad (2)$$

where $w(e_i)$ represents the weight of edge e_i, $F_{decay}(P)$ is a decay function, which can increase the influence of path length as the long path indicate weak association between two endpoints, computed by

$$F_{decay}(P) = \theta \cdot l, \quad (3)$$

where θ is a parameter, which was set to 2.26 as suggested in [36−40]. For each drug p in $N_k(p_i)$ ($k > 1$), its linkage to p_i can be measured by the weights of the paths connecting them. If multiple paths connecting them, the maximum path weight was selected. Based on the above definitions, the linkage between p_i and drug p in $N_k(p_i)$ can be synthesized as follows:

$$L(p_i, p) = \begin{cases} Q^f(p_i, p) & p \in N_1(p_i) \\ \max\{w(P_i) | i = 1,2,\cdots,m\} & p \in N_k(p_i)\ (k > 1) \end{cases}, \quad (4)$$

where $P_1, P_2, \cdots, P_m$ represent all paths connecting p_i and p with length k. In $N_k(p_i)$, some drugs can constitute DDAs with q_j, whereas others cannot. In view of this, we defined an indictor function as follows:

$$\Delta(p, q_j) = \begin{cases} 1 & \text{if } p \text{ and } q_j \text{ can constitute a DDA} \\ 0 & \text{otherwise} \end{cases}. \quad (5)$$

Then, the relationship between p_i and q_j can be measured by the following level score:

$$Score(p_i, q_j) = \sum_{k=1}^{t} \sum_{p \in N_k(p_i)} L(p_i, p) \cdot \Delta(p, q_j). \quad (6)$$

Because it is time-consuming to find all paths connecting two nodes with a long distance, a threshold t was employed in Eq (6). Such a setting was also reasonable because paths with long distances play few or even no contributions to measure the linkage between p_i and q_j. An example is shown in Figure 1, where the threshold t is set to 2.

It is clear that the high outcome of Eq (6) indicated the strong relationships between the drug and disease. On the contrary, it was almost impossible for the pair of drugs and diseases with a low score to be an actual DDA. These pairs can be high-quality negative samples and may be helpful to construct classifiers with a high performance. In theory, the unlabeled pairs of drugs and diseases with low level scores should be selected as negative samples. Such an operation can be conducted by setting a low threshold s to the level score (i.e., the unlabeled pairs of drugs and diseases with level scores no more than s were selected). These selected negative samples comprised a negative sample pool, denoted by $NS(s)$.

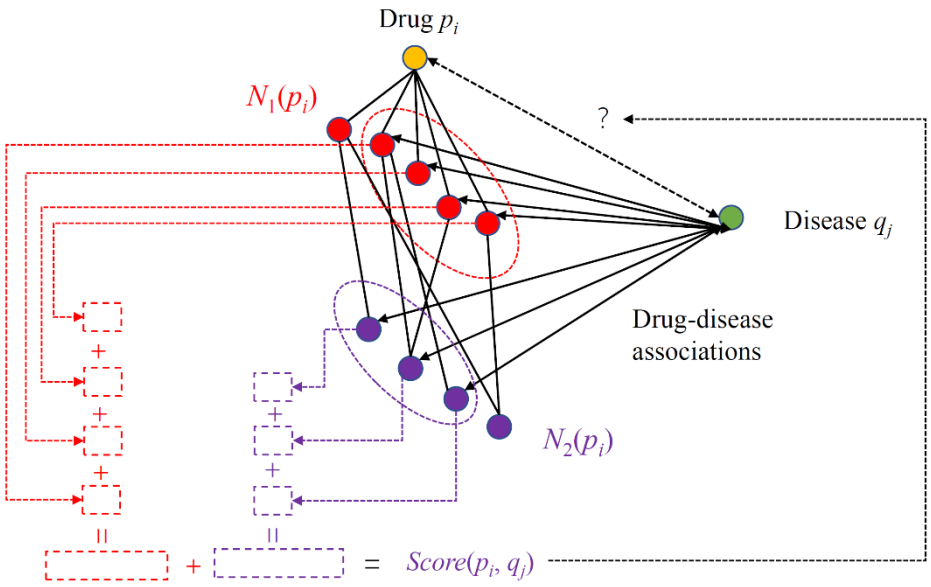

Figure 1. An example for showing the principle to calculate the level score of one drug and one disease using the direct neighbors and 2-neighbors of the drug.

2.3. Network construction and feature extraction

In traditional machine learning, it is very important to encode each sample with its essential properties. In this study, we directly adopted the drug and disease features reported in our previous study [21], which were derived from multiple networks. Networks are recently a popular research form as they can overview each object with all other objects as background. As the procedures of network construction and feature extraction have been described in detail in a previous study [21], we only gave a brief introduction on such procedures.

2.3.1. Drug network construction

Twelve drug networks were constructed, where eight contained only drugs, and the other four included other objects. For the former eight networks, they were constructed in terms of drug associations collected from two public databases: KEGG (https://www.genome.jp/kegg/) [41,42] and STITCH (http://stitch.embl.de, version 4.0) [43], and that defined by the Anatomical Therapeutic Chemical (ATC) codes of two drugs. As for the later four networks, they indicated the relationships

between drug and one of the following objects: proteins, pathways, side effects, and gene ontology (GO) terms. The target proteins and side effects of drugs were retrieved from DrugBank (https://go.drugbank.com/) [44] and SIDER (http://sideeffects.embl.de/) [45], respectively. In CTD, the related GO terms and pathways for chemicals were also collected, which were picked up for building drug networks.

2.3.2. Disease network construction

Three networks were built for diseases, involving basic information of diseases such as pathway, gene, and phenotype information. This information was also sourced from the CTD. According to the related pathways of two diseases, their association was measured by the Tanimoto coefficient of two pathway sets. The disease associations based on gene and phenotype information can be assessed in the same way. Then, three networks were built with these disease associations.

2.3.3. Feature extraction

The drug and disease networks mentioned above contained abundant information of drugs and diseases, respectively. Informative drug and disease features can be extracted from them. As multiple networks were constructed for drug and disease, a powerful network embedding algorithm, Mashup [46], was adopted. Its greatest merit is that it can process more than one network. Two stages are contained in this algorithm. In the first stage, the raw feature vector for each node in each network is extracted in terms of a random walk with restart [47,48]. The feature vectors for the same node that are derived from different networks are fused in the second stage. At the same time, the dimension is reduced. Mashup was applied to the twelve drug networks to generate drug features and the disease features were produced from three disease networks. Various dimensions, changing from 50 to 1000, were produced for drugs and diseases. The optimal dimension for drugs and diseases can be determined by a ten-fold cross-validation [49].

2.4. Classification algorithms

In this study, the RNSS was proposed to select reliable negative samples. Three classic classification algorithms were selected to construct models based on positively validated positive samples and selected reliable negative samples, thereby elaborating the utility of an RNSS. These classification algorithms included RF [29], BN and NNA [30], which were also adopted in the previous study [21]. These algorithms were designed using quite different ideas and principles. Their common results can provide a universal significance. For this investigation, if the classifiers with an RNSS were generally better than those without an RNSS or with other negative sample selection schemes for any of these three algorithms, it can prove that an RNSS is an efficient scheme to select high-quality negative samples. To quickly implement the above algorithms, corresponding tools (RandomForest, BayesNet and IBk) in Weka [50] were employed. Their default parameters were adopted because the purpose of this study was to test whether the employment of an RNSS can improve the performance of models rather than to build models with excellent performance.

2.5. Performance evaluation

In this study, a ten-fold cross-validation was adopted to evaluate the performance of all constructed classifiers [49]. Such a method divides samples into ten parts. Each part is singled out as the test set and the rest of the parts constitute the training set. The classifier based on the training set is applied to the test set. In an RNSS, the calculation of the level score is related to the positive samples in the training set. Thus, we first divided the positive samples into ten parts. When one part of the positive samples was singled out, which was put into the test set, we used the rest of the positive samples to compute the level scores of unlabeled samples and then selected negative samples. In this way, the information of the test samples was completely excluded when training the classifiers. It was a rigorous cross-validation.

For the binary classification, plenty of measurements have been designed to evaluate the performance of various models. The direct way to display the predicted results of one model is a confusion matrix, which contains four entries: true positive (TP), false negative (FN), false positive (FP), and true negative (TN). Several measurements can be calculated according to these entries. In this study, we selected sensitivity (SN), specificity (SP), accuracy (ACC), precision, F1-measure [51−57], and the Matthews correlation coefficient (MCC) [58], which can be computed by

$$SN = \frac{TP}{TP+FN} \tag{7}$$

$$SP = \frac{TN}{TN+FP} \tag{8}$$

$$\mathrm{ACC} = \frac{TP+TN}{TP+FP+TN+FN} \tag{9}$$

$$Precision = \frac{TP}{TP+FP} \tag{10}$$

$$\mathrm{F1-measure} = \frac{2\times Recall\times Precision}{Recall+Precision} \tag{11}$$

$$\mathrm{MCC} = \frac{TP\times TN-FP\times FN}{\sqrt{(TP+FP)\times(TP+FN)\times(TN+FP)\times(TN+FN)}} \tag{12}$$

Besides, the receiver operating characteristic (ROC) and precision-recall (PR) curves were used to comprehensively evaluate the models' performance. The ROC curve sets the SN as the Y-axis and the false positive rate (i.e., 1-SP) as the X-axis, which are obtained by setting different thresholds. The PR curve is defined in a similar way, which defines recall (i.e., SN) as the X-axis and precision as the Y-axis. The area under the ROC and PR curves are essential measurements to assess the models' performance, which were denoted by AUROC and AUPR, respectively.

3. Results and discussion

In this study, a computation method was proposed to identify DDAs. To enhance the performance of the method, a novel negative sample selection scheme, RNSS, was designed. The classifiers using samples selected by an RNSS were built and evaluated. The entire procedure is illustrated in Figure 2.

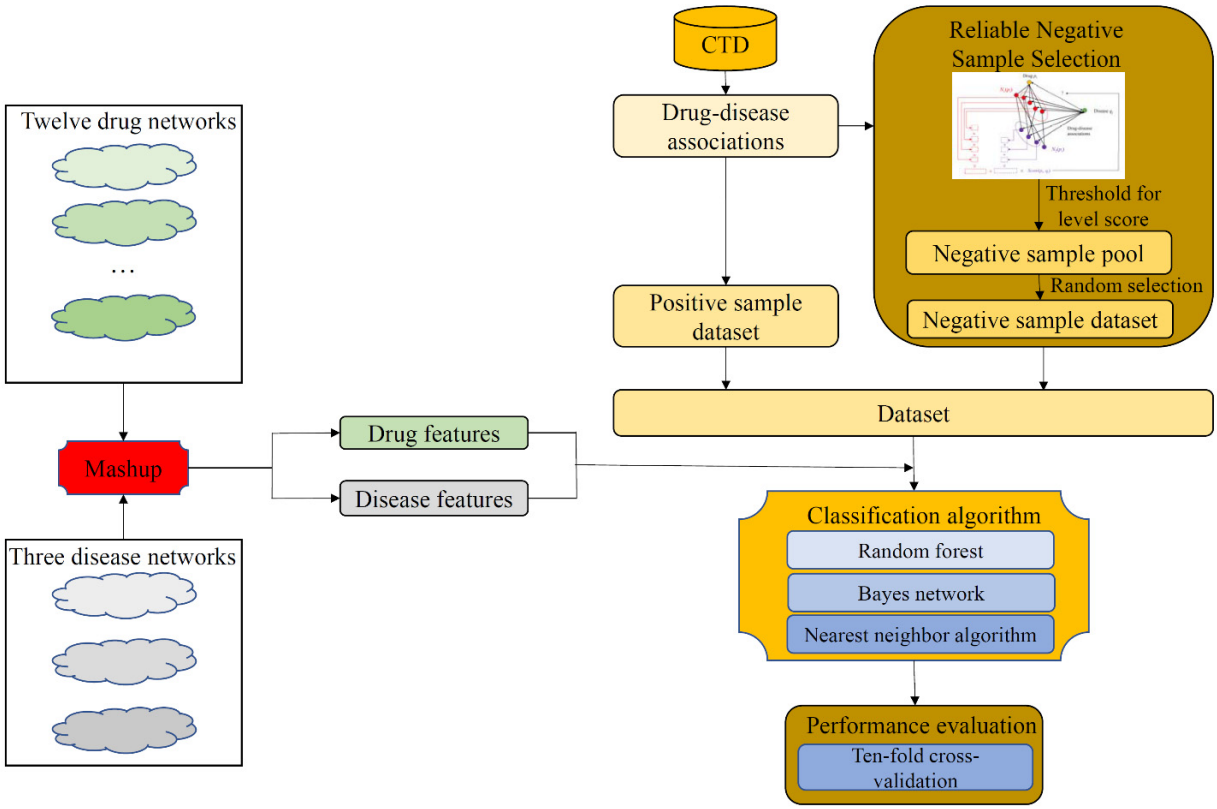

Figure 2. Entire procedures of this study. Validated drug-disease associations (DDAs) are retrieved from CTD and constitute the positive sample dataset. The reliable negative sample selection scheme is designed to screen out negative samples with high quality and comprise negative sample dataset. Each sample is represented by drug and disease features derived from multiple drug and disease networks via Mashup. Three classification algorithms are adopted to build the classifiers for evaluating the effectiveness of the negative sample selection scheme.

3.1. Performance of the classifiers with RNSS on balanced datasets

When assessing the level score of one drug and one disease, the threshold t (Eq (6)) determined which k-neighbors of the drug were considered. It is clear that neighbors with a long distance to the drug give few contributions. Here, we set such threshold t as two, that is, the direct neighbors and 2-neighbors of the drug were included to assess its associations to diseases. After obtaining the level scores of all unlabeled pairs using Eq (6) with $t = 2$, we set two thresholds (0.05 and 0.1) to construct two negative sample pools (i.e., $NS(0.05)$ and $NS(0.1)$). From each pool, the same number of negative samples to the positive samples were randomly selected and combined with positive samples to constitute a balanced dataset. Each sample was represented by drug and disease features derived from multiple drug and disease networks. The dimensions for drug and disease features were set to various values between 50 and 1,000. All possible dimension combinations were attempted. Three classification algorithms (RF, BN, and NNA) were applied to construct classifiers on balanced datasets. These classifiers were assessed by a ten-fold cross-validation. The best performance

(measured by MCC) of the RF, BN, and NNA classifiers using negative samples selected from two pools is listed in Table 1.

Table 1. Performance of three classifiers with RNSS built on balanced datasets.

Classification algorithm	Negative sample pool	Dimension		SN	SP	ACC	Precision	F1-measure	MCC
		Drug feature	Disease feature						
Random forest	*NS*(0.05)	1000	1000	1.000	0.979	0.989	0.979	0.989	0.979
	NS(0.1)	850	900	1.000	0.953	0.975	0.950	0.974	0.951
Bayes network	*NS*(0.05)	550	50	1.000	0.979	0.989	0.979	0.989	0.979
	NS(0.1)	550	50	1.000	0.952	0.975	0.950	0.974	0.950
Nearest neighbor algorithm	*NS*(0.05)	1000	50	0.978	0.974	0.976	0.974	0.976	0.951
	NS(0.1)	1000	50	0.956	0.954	0.955	0.954	0.955	0.910

For the balanced dataset with negative samples selected from *NS*(0.05), the RF classifier yielded an MCC of 0.979, which was very high. The other five measurements (SN, SP, ACC, precision, and F1-measure) were 1.000, 0.979, 0.989, 0.979, and 0.989, respectively. Such a performance suggested that the RF classifier can give a nearly perfect prediction. As for the BN and NNA classifiers, they also produced a high performance. The MCC values of these two classifiers were 0.979 and 0.951, respectively. Considering that BN and NNA were not very powerful classification algorithms, such a performance was extreme high for them. The ROC and PR curves of above three classifiers are illustrated in Figure 3. The AUROC values for three classifiers were 0.9962, 0.9893, and 0.9758, respectively, and AUPR values were 0.9969, 0.9893, and 0.9653, respectively. They were all very high, further suggesting the high performance of three classifiers. The above results implied that the negative samples selected from *NS*(0.05) were quite different from the positive samples, inducing easy classifications. This fact also proves the effectiveness of the RNSS.

For another pool *NS*(0.1), negative samples were also randomly selected and comprised the balanced dataset with positive samples. The MCC of the RF classifier on such a dataset was 0.951, which was lower than that yielded by the RF classifier using samples selected from *NS*(0.05). The SP, ACC, precision, and F1-measure all slightly decreased compared with those of the RF classifier using the samples selected from *NS*(0.05) (Table 1). The ROC and PR curves as alongside the AUROC and AUPR (Figure 3) of the RF classifier on such a balanced dataset were inferior to those yielded by the RF classifier using samples selected from *NS*(0.05). A similar phenomenon occurred for the BN and NNA classifiers. It was suggested that the level score (Eq (6)) can really indicate the quality of negative samples, that is, low level scores indicated the high quality of negative samples. It was a proper strategy to select negative samples with low level scores.

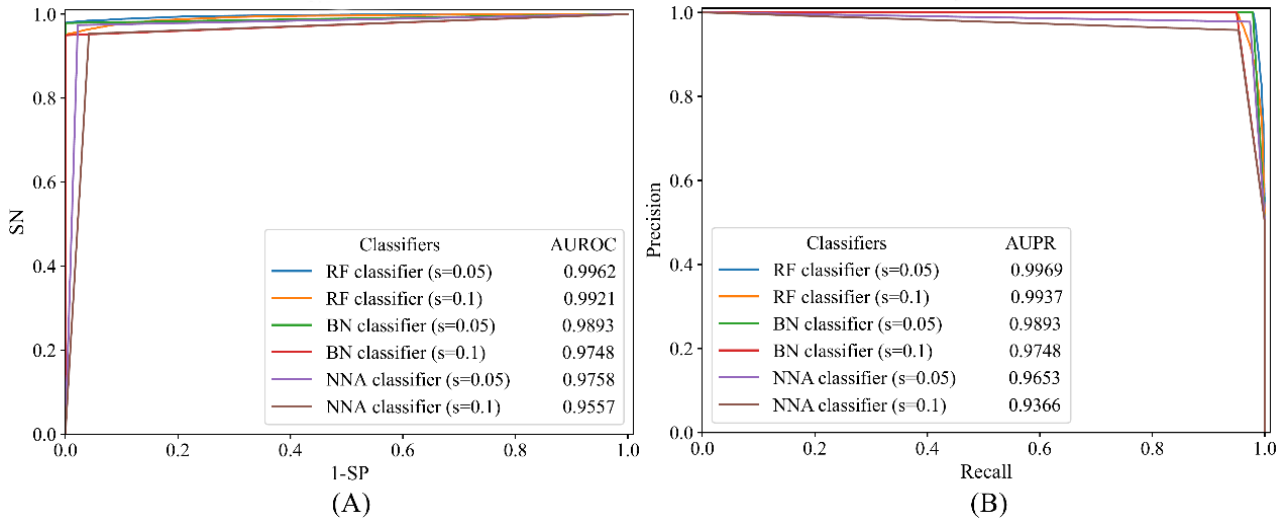

Figure 3. ROC and PR curves of three classifiers on balanced datasets. (A) ROC curves; (B) PR curves. The parameters *s* represents the threshold for constructing negative sample pool.

3.2. Performance of the classifiers with RNSS on imbalanced datasets

In Section 3.1, all classifiers were set up on balanced datasets. To give a further test, some classifiers based on imbalanced datasets were constructed and evaluated. We randomly selected negative samples from each negative sample pool that were twice or thrice as many as positive samples to comprise imbalanced datasets. On each imbalanced dataset, the classifiers with different combinations of drug and disease feature dimensions were set up and evaluated by a ten-fold cross-validation. The best performance for three classification algorithms is listed in Tables 2 and 3.

Of the imbalanced datasets, which contained negative samples twice as many as positive samples, the RF classifier generated MCC values of 0.984 and 0.962 for two negative sample pools (Table 2). The values for the BN classifier were 0.984 and 0.963 (Table 2). For the NNA classifier, it yielded MCC values of 0.962 and 0.925 (Table 2). It was amazing that these three classifiers on such imbalanced datasets provided even better performance than those on the balanced datasets, as described in Section 3.1. Generally, the performance of the classifiers on imbalanced dataset may decrease, especially on the minor class. However, the overall performance of the above classifiers actually increased. As for their performance on the minor class (positive samples), measured by SN, the decrease range was quite limited. The ROC and PR curves of the above classifiers are illustrated in Figure 4(A),(B). Based on AUROC and AUPR, the performance of the classifiers on imbalanced datasets did not clearly decrease, quite coincident with results assessed by other measurements.

Table 2. Performance of three classifiers with RNSS built on imbalanced datasets, where the negative samples are twice as many as positive samples.

Classification algorithm	Negative sample pool	Dimension		SN	SP	ACC	Precision	F1-measure	MCC
		Drug feature	Disease feature						
Random forest	*NS*(0.05)	1000	1000	1.000	0.989	0.993	0.979	0.989	0.984
	NS(0.1)	850	900	0.999	0.975	0.983	0.950	0.974	0.962
Bayes network	*NS*(0.05)	550	50	1.000	0.989	0.993	0.979	0.989	0.984
	NS(0.1)	550	50	1.000	0.975	0.983	0.950	0.974	0.963
Nearest neighbor algorithm	*NS*(0.05)	1000	50	0.978	0.985	0.983	0.971	0.974	0.962
	NS(0.1)	1000	50	0.953	0.973	0.967	0.946	0.950	0.925

Table 3. Performance of three classifiers with RNSS built on imbalanced datasets, where the negative samples are thrice as many as positive samples.

Classification algorithm	Negative sample pool	Dimension		SN	SP	ACC	Precision	F1-measure	MCC
		Drug feature	Disease feature						
Random forest	*NS*(0.05)	1000	1000	0.999	0.992	0.994	0.979	0.989	0.984
	NS(0.1)	850	900	0.999	0.984	0.987	0.950	0.974	0.966
Bayes network	*NS*(0.05)	550	50	1.000	0.992	0.994	0.979	0.989	0.985
	NS(0.1)	550	50	1.000	0.984	0.987	0.950	0.974	0.966
Nearest neighbor algorithm	*NS*(0.05)	1000	50	0.976	0.988	0.985	0.970	0.973	0.962
	NS(0.1)	1000	50	0.951	0.982	0.974	0.945	0.948	0.931

For other imbalanced datasets containing negative samples thrice as many as positive samples, the performance of the three classifiers on two pools is shown in Table 3. It can be observed that such a performance was quite similar to that in Tables 1 and 2. The same conclusions can be obtained from the ROC and PR curves (Figure 4(C),(D)) of these classifiers. These results indicate that the classifiers with an RNSS were not very sensitive to the imbalanced problem. As the negative sample pools constructed by the RNSS included negative samples with high quality, the increment on negative samples did not amplify the difficulties for learning an efficient classifier. Furthermore, the classifiers using negative samples selected from *NS*(0.05) were superior to those using negative samples selected from *NS*(0.1), suggesting that the level score was a good indicator to select negative samples with higher quality.

3.3. The effect of the distance limitation

In the RNSS, the distance limitation t was an important parameter, which directly influenced the calculation of the level score (see Eq (6)). It was interesting to investigate the effect of such a parameter. The above classifiers were all based on negative samples selected by the RNSS with $t = 2$. Here, we investigated the classifiers using negative samples selected by an RNSS with $t = 1$. In fact, an RNSS

with $t = 1$ was the same as a previous negative sample selection method reported in [59], named finding reliable negative samples (FIRE).

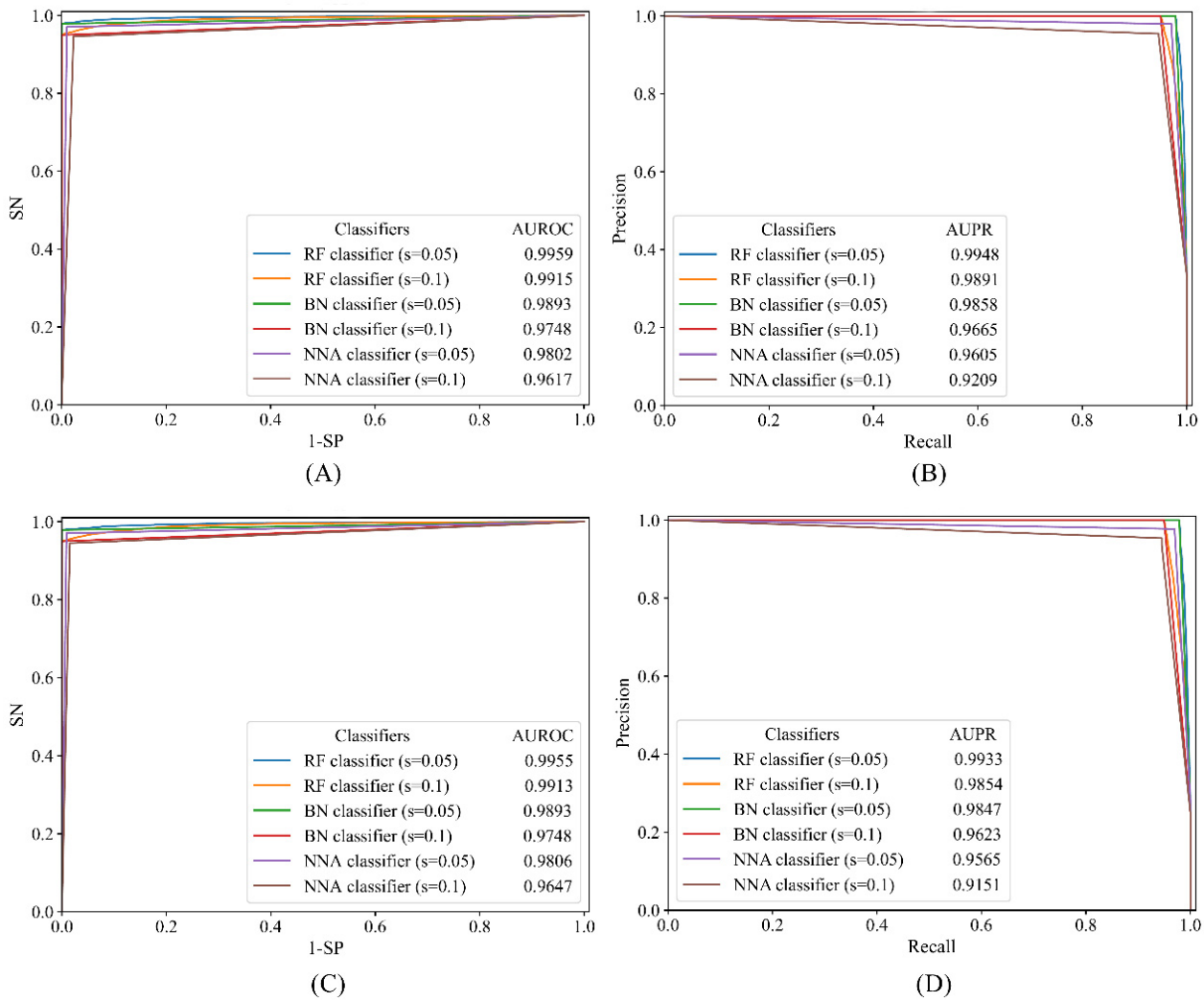

Figure 4. ROC and PR curves of three classifiers on imbalanced datasets. (A) ROC curves on imbalanced datasets, where negative samples are twice as many as positive samples; (B) PR curves on imbalanced datasets, where negative samples are twice as many as positive samples; (C) ROC curves on imbalanced datasets, where negative samples are thrice as many as positive samples; (D) PR curves on imbalanced datasets, where negative samples are thrice as many as positive samples. The parameters s represents the threshold for constructing negative sample pool.

When t was set to 1, numerous unlabeled samples were assigned a level score of 0. Thus, we added the threshold 0 for parameter s (i.e., three negative sample pools were considered), including $NS(0)$, $NS(0.05)$ and $NS(0.1)$. From each pool, we first randomly selected as many negative samples as positive samples to constitute balanced datasets. Three classifiers with different parameter combinations mentioned above were constructed on each balanced dataset and evaluated by a ten-fold cross-validation. The best performance for each classification algorithm was picked up and detailed

measurements listed in Section 2.6 are illustrated in Figure 5. For easy comparisons, the performance of classifiers with an RNSS ($t = 2$) is also listed in this figure. Given the same classification algorithm (RF, BN or NNA), classifiers with an RNSS ($t = 2$) were generally superior to those with an RNSS ($t = 1$) in terms of all measurements. The improvement for BN and NNA classifiers was very great, whereas that for the RF classifier was slightly enhanced. As the RF classifiers with an RNSS ($t = 1$) provided a relatively higher performance than BN and NNA classifiers with an RNSS ($t = 1$), it was difficult to achieve any improvements.

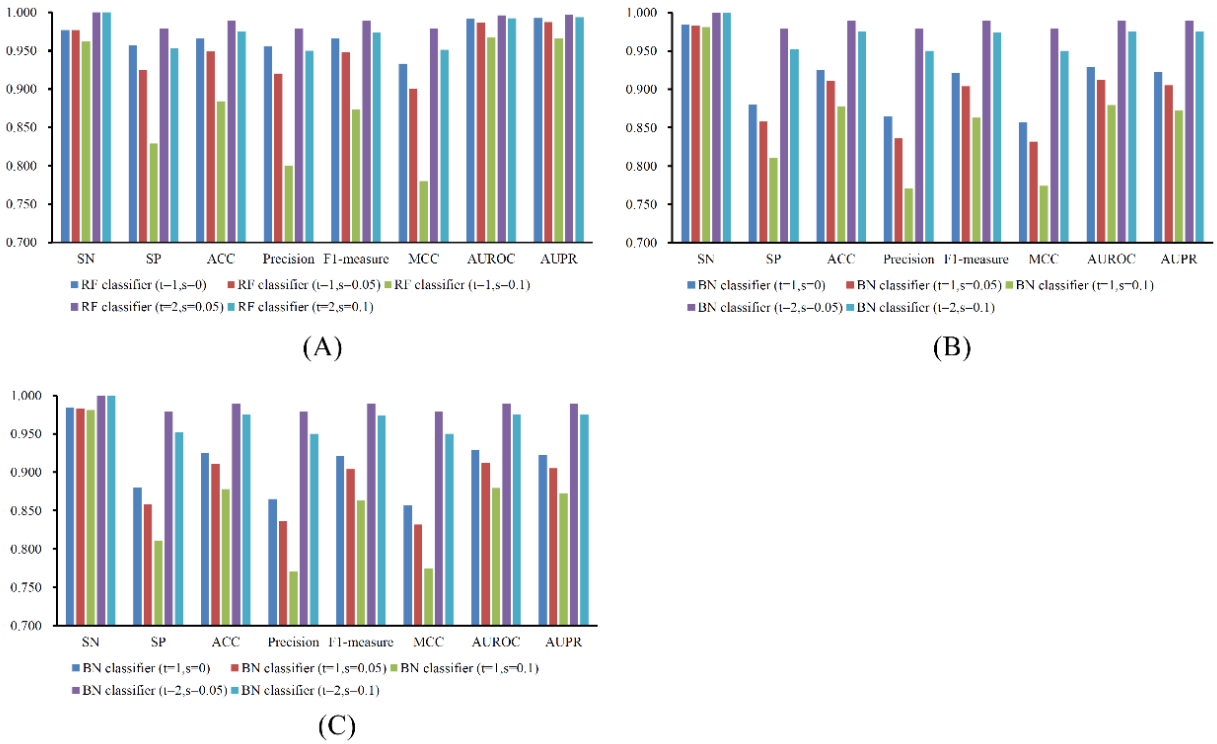

Figure 5. Comparison of three classifiers on balanced datasets containing negative samples selected by RNSS with different parameters. (A) Comparison of RF classifiers; (B) Comparison of BN classifiers; (C) Comparison of NNA classifiers. The parameter t represents the distance limitation for calculating level score and s indicates the threshold for constructing negative sample pool.

In addition, the imbalanced datasets were constructed by selecting two and three times as many negative samples as positive samples from each of three pools. Classifiers built on these datasets were also evaluated by a ten-fold cross-validation. The best performance for each classification algorithm is shown in Appendix Figures A1 and A2, from which we can conclude the same result (i.e., the classifiers with an RNSS ($t = 2$) were obviously better than those with an RNSS ($t = 1$)).

With the above arguments, despite the balanced or imbalanced datasets, classifiers with an RNSS ($t = 2$) yielded better performance, indicating that the employment of 2-neighbors of drugs can improve the selection of negative samples. As mentioned above, numerous unlabeled samples were assigned a level score of zero by an RNSS ($t = 1$). The drug in each of these samples did not have direct neighbors that were related to the disease in the same sample. However, such a drug may have some 2-neighbors

that were associated with the disease, which made the level score yielded by an RNSS (t = 2) larger than zero. The employment of 2-neighbors can help us further classify these unlabeled samples, thereby screening out negative samples with a higher quality.

3.4. Comparison with classifiers using randomly selected negative samples

In many studies, random selection of negative samples is a widely used scheme to construct binary classifiers [21,26,27,60,61]. Here, several classifiers were built using such scheme, which were compared with classifiers with an RNSS to elaborate the superiority of the RNSS. To give a full comparison, we constructed three datasets containing negative samples one, two, and three times as many as positive samples. Three classification algorithms (RF, BN and NNA) were adopted to build the classifiers. Feature dimensions of drug and disease were the same as classifiers with an RNSS (t = 2, s = 0.05), as listed in Tables 1−3. All classifiers were also assessed by a ten-fold cross-validation. The performance is listed in Table 4.

Table 4. Performance of three classifiers using randomly selected negative samples.

Classification algorithm	Ratio of positive and negative samples	SN	SP	ACC	Precision	F1-measure	MCC	AUROC	AUPR
Random forest	1:1	0.739	0.782	0.761	0.773	0.756	0.522	0.818	0.803
	1:2	0.570	0.884	0.779	0.711	0.632	0.483	0.818	0.690
	1:3	0.435	0.925	0.803	0.659	0.524	0.420	0.807	0.591
Bayes network	1:1	0.667	0.837	0.752	0.804	0.729	0.512	0.751	0.703
	1:2	0.673	0.836	0.782	0.672	0.672	0.509	0.751	0.558
	1:3	0.675	0.826	0.788	0.564	0.614	0.473	0.747	0.460
Nearest neighbor algorithm	1:1	0.695	0.685	0.690	0.688	0.691	0.380	0.671	0.616
	1:2	0.571	0.780	0.711	0.565	0.568	0.351	0.660	0.451
	1:3	0.497	0.826	0.744	0.487	0.492	0.321	0.638	0.346

For balanced dataset (i.e., the ratio of positive and negative samples was 1:1), the MCC values of RF, BN, and NNA classifiers were only 0.522, 0.512, and 0.380, respectively. Compared with the MCC values of RF, BN, and NNA classifiers with an RNSS (t = 2, s = 0.05), which were 0.979, 0.979, and 0.951 (Table 1), respectively, such a performance was much lower. The same results can be obtained in terms of other measurements. Thus, the classifiers with an RNSS were much stronger than those using randomly selected negative samples. For the imbalanced datasets (the ratio of positive and negative samples = 1:2 or 1:3), we can also conclude that classifiers using randomly selected negative samples was much inferior to those with an RNSS (see Tables 2−4). Above results indicated that the RNSS was effective to help us select negative samples with high quality, thereby improving the classifiers.

3.5. Comparison with classifiers using negative samples clustered by K-means

Besides random selection of negative samples, some studies adopted another scheme to select negative samples [62,63]. For unlabeled samples, K-means were adopted to cluster them and negative

samples were equally and randomly selected from each cluster. As stated in [64], the clustering effect was best when the unlabeled samples were clustered into 23 clusters. We also adopted such a setting (i.e., unlabeled pairs of drugs and diseases were clustered into 23 clusters). Unlabeled pairs selected from each cluster were combined to constitute the negative sample set, whose size was the same as the positive sample set. Three classification algorithms (RF, BN, and NNA) were adopted to build the classifiers. Feature dimensions of drug and disease were the same as classifiers with an RNSS (t = 2, s = 0.05), as listed in Tables 1. The evaluation results yielded by the ten-fold cross-validation are listed in Table 5. Compared with evaluation results of classifiers with an RNSS (Table 1), these classifiers were very poor. These results further confirmed the utility of the RNSS.

Table 5. Performance of three classifiers using negative samples clustered by K-means.

Classification algorithm	Ratio of positive and negative samples	SN	SP	ACC	Precision	F1-measure	MCC	AUROC	AUPR
Random forest	1:1	0.053	0.867	0.460	0.271	0.088	-0.142	0.600	0.510
Bayes network	1:1	0.662	0.809	0.736	0.776	0.714	0.477	0.734	0.684
Nearest neighbor algorithm	1:1	0.732	0.757	0.745	0.751	0.741	0.490	0.722	0.669

3.6. Comparison with previous drug-disease association prediction methods

To date, several DDA prediction methods have been proposed. Here, some of them were selected to compare with our method with the RNSS. Their performance is listed in Table 6. For easy comparisons, the performance of our method (RF classifier with s = 0.05 and t = 2) is also provided in this table. It can be observed that our method provided best performance for all eight measurements. This result indicated the superiority of our model and the RNSS.

Table 6. Performance of different drug-disease association prediction methods[$].

Model	SN	SP	ACC	Precision	F1-measure	MCC	AUROC	AUPR
Our model	1.000	0.979	0.989	0.979	0.989	0.979	0.996	0.997
RepCOOL [24]	0.930	-	-	0.530	0.670	-	0.670	-
RLFDDA [25]	0.897	-	0.901	0.904	0.900	-	0.964	-
Li et al.' method [26]	0.862	0.868	0.865	0.867	-	0.730	0.936	0.935
MGP-DDA [23]	0.842	-	0.867	0.886	0.863	-	0.930	0.944
Yang and Chen's method [21]	0.872	0.843	0.858	0.847	0.860	0.716	0.928	0.919

$: Measurements for all methods except our method were directly picked up from their corresponding literature; -: This measurement was not reported.

3.7. Similarities and differences to previous negative sample selection schemes

The selection of negative samples is a challenging problem in association prediction. Some schemes have been designed in recent years [59,65,66]. The proposed scheme, RNSS, is more similar to the methods in [59,65]. Thus, this section focused on the similarities and differences to the method in [66]. This method was called self-paced negative sampling strategy (SNSS).

SNSS employs a hardness function to indicate the likehood of one unlabeled sample to be an actual negative sample, similar to the level score in our scheme. This function relies on a multilayer perceptron (MLP) classifier, which is very different from the drug network in our scheme. It is defined as the differences of the probability score yielded by an MLP and the ground-truth label. It is hard to say which scoring system is better, as there does not exist any generally accepted dataset to validate whether the score is correct. On the other hand, the selection strategies of the two methods were quite different. In SNSS, it divides unlabeled samples into some categories according to the results of hardness function and selects negative samples from each category with different proportions. The selection scope was all unlabeled samples. Such a selection can fully train the classifier and increase the robustness. In our scheme, we select negative samples from a pool consisting of unlabeled samples with low level scores (i.e., the selection scope was a part of unlabeled samples). Such a selection can improve the performance of the classifier. The different intentions induce different selection strategies. These methods provide alternative ways to select negative samples when building the binary association prediction methods. It is also interesting to fuse the merits of these two methods for designing a new negative sample selection method.

4. Conclusions

In this study, we proposed a novel scheme to select high-quality negative drug-disease samples. To elaborate its utility, several classifiers were constructed with negative samples selected by such a scheme. The evaluation results suggested that these classifiers had an extremely strong ability to identify DDAs. Additionally, these classifiers were much better than those using some traditional and previous schemes, confirming the positive effects of the proposed scheme in selecting high-quality negative samples. It is hopeful that such a scheme can be applied to other related problems for building classifiers with a high performance.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. D. McHale, M. Penny, Chapter 19 - Genomics, New drug development, and precision medicines, in *Medical and Health Genomics*, Oxford: Academic Press, (2016), 247−259. https://doi.org/10.1016/B978-0-12-420196-5.00019-8
2. C. W. Lindsley, New statistics on the cost of new drug development and the trouble with CNS drugs, *ACS Chem. Neurosci.*, **5** (2014), 1142. https://doi.org/10.1021/cn500298z
3. M. R. Hurle, L. Yang, Q. Xie, D. K. Rajpal, P. Sanseau, P. Agarwal, Computational drug repositioning: from data to therapeutics, *Clin. Pharmacol. Ther.*, **93** (2013), 335−341. https://doi.org/10.1038/clpt.2013.1
4. J. Li, S. Zheng, B. Chen, A. J. Butte, S. J. Swamidass, Z. Lu, A survey of current trends in computational drug repositioning, *Briefings Bioinf.*, **17** (2016), 2−12. https://doi.org/10.1093/bib/bbv020
5. Q. Dai, C. Bao, Y. Hai, S. Ma, T. Zhou, C. Wang, et al., MTGIpick allows robust identification of genomic islands from a single genome, *Briefings Bioinf.*, **19** (2018), 361−373. https://doi.org/10.1093/bib/bbw118
6. R. Kong, X. Xu, X. Liu, P. He, M. Q. Zhang, Q. Dai, 2SigFinder: the combined use of small-scale and large-scale statistical testing for genomic island detection from a single genome, *BMC Bioinf.*, **21** (2020), 159. https://doi.org/10.1186/s12859-020-3501-2
7. D. Lai, L. Tan, X. Zuo, D. Liu, D. Jiao, G. Wan, et al., Prognostic ferroptosis-related lncRNA signatures associated with immunotherapy and chemotherapy responses in patients with stomach cancer, *Front. Genet.*, **12** (2022), 798612. https://doi.org/10.3389/fgene.2021.798612
8. F. Napolitano, Y. Zhao, V. M. Moreira, R. Tagliaferri, J. Kere, M. D'Amato, et al., Drug repositioning: a machine-learning approach through data integration, *J. Cheminf.*, **5** (2013), 30. https://doi.org/10.1186/1758-2946-5-30
9. Z. Cui, Y. L. Gao, J. X. Liu, J. Wang, J. Shang, L. Y. Dai, The computational prediction of drug-disease interactions using the dual-network $L_{2,1}$-CMF method, *BMC Bioinf.*, **20** (2019), 5. https://doi.org/10.1186/s12859-018-2575-6
10. Y. Wang, S. Chen, N. Deng, Y. Wang, Drug repositioning by kernel-based integration of molecular structure, molecular activity, and phenotype data, *PLoS One*, **8** (2013), e78518. https://doi.org/10.1371/journal.pone.0078518
11. L. Lu, H. Yu, DR2DI: a powerful computational tool for predicting novel drug-disease associations, *J. Comput.-Aided Mol. Des.*, **32** (2018), 633−642. https://doi.org/10.1007/s10822-018-0117-y
12. C. Q. Gao, Y. K. Zhou, X. H. Xin, H. Min, P. F. Du, DDA-SKF: predicting drug-disease associations using similarity kernel fusion, *Front. Pharmacol.*, **12** (2021), 784171. https://doi.org/10.3389/fphar.2021.784171
13. G. Wu, J. Liu, C. Wang, Predicting drug-disease interactions by semi-supervised graph cut algorithm and three-layer data integration, *BMC Med. Genomics*, **10** (2017), 79. https://doi.org/10.1186/s12920-017-0311-0
14. A. P. Chiang, A. J. Butte, Systematic evaluation of drug-disease relationships to identify leads for novel drug uses, *Clin. Pharmacol. Ther.*, **86** (2009), 507−510. https://doi.org/10.1038/clpt.2009.103

15. C. Wu, R. C. Gudivada, B. J. Aronow, A. G. Jegga, Computational drug repositioning through heterogeneous network clustering, *BMC Syst. Biol.*, **7** (2013), S6. https://doi.org/10.1186/1752-0509-7-S5-S6
16. H. Luo, J. Wang, M. Li, J. Luo, X. Peng, F. X. Wu, et al., Drug repositioning based on comprehensive similarity measures and Bi-Random walk algorithm, *Bioinformatics*, **32** (2016), 2664−2671. https://doi.org/10.1093/bioinformatics/btw228
17. W. Wang, S. Yang, X. Zhang, J. Li, Drug repositioning by integrating target information through a heterogeneous network model, *Bioinformatics*, **30** (2014), 2923−2930. https://doi.org/10.1093/bioinformatics/btu403
18. V. Martínez, C. Navarro, C. Cano, W. Fajardo, A. Blanco, DrugNet: network-based drug-disease prioritization by integrating heterogeneous data, *Artif. Intell. Med.*, **63** (2015), 41−49. https://doi.org/10.1016/j.artmed.2014.11.003
19. Y. F. Huang, H. Y. Yeh, V. W. Soo, Inferring drug-disease associations from integration of chemical, genomic and phenotype data using network propagation, *BMC Med. Genomics*, **6** (2013), S4. https://doi.org/10.1186/1755-8794-6-S3-S4
20. A. Gottlieb, G. Y. Stein, E. Ruppin, R. Sharan, PREDICT: a method for inferring novel drug indications with application to personalized medicine, *Mol. Syst. Biol.*, **7** (2011), 496. https://doi.org/10.1038/msb.2011.26
21. Y. Yang, L. Chen, Identification of drug-disease associations by using multiple drug and disease networks, *Curr. Bioinf.*, **17** (2022), 48−59. https://doi.org/10.2174/1574893616666210825115406
22. H. Jiang, Y. Huang, An effective drug-disease associations prediction model based on graphic representation learning over multi-biomolecular network, *BMC Bioinf.*, **23** (2022), 9. https://doi.org/10.1186/s12859-021-04553-2
23. T. Kawichai, A. Suratanee, K. Plaimas, Meta-path based gene ontology profiles for predicting drug-disease associations, *IEEE Access*, **9** (2021), 41809−41820. https://doi.org/10.1109/ACCESS.2021.3065280
24. G. Fahimian, J. Zahiri, S. S. Arab, R. H. Sajedi, RepCOOL: computational drug repositioning via integrating heterogeneous biological networks, *J. Transl. Med.*, **18** (2020), 375. https://doi.org/10.1186/s12967-020-02541-3
25. M. L. Zhang, B. W. Zhao, X. R. Su, Y. Z. He, Y. Yang, L. Hu, RLFDDA: a meta-path based graph representation learning model for drug-disease association prediction, *BMC Bioinf.*, **23** (2022), 516. https://doi.org/10.1186/s12859-022-05069-z
26. Z. Li, Q. Huang, X. Chen, Y. Wang, J. Li, Y. Xie, et al., Identification of drug-disease associations using information of molecular structures and clinical symptoms via deep convolutional neural network, *Front. Chem.*, **7** (2019), 924. https://doi.org/10.3389/fchem.2019.00924
27. Z. Wang, M. Zhou, C. Arnold, Toward heterogeneous information fusion: bipartite graph convolutional networks for in silico drug repurposing, *Bioinformatics*, **36** (2020), i525−i533. https://doi.org/10.1093/bioinformatics/btaa437
28. B. W. Zhao, Z. H. You, L. Wong, P. Zhang, H. Y. Li, L. Wang, MGRL: predicting drug-disease associations based on multi-graph representation learning, *Front. Genet.*, **12** (2021), 657182. https://doi.org/10.3389/fgene.2021.657182
29. L. Breiman, Random forests, *Mach. Learn.*, **45** (2001), 5−32. https://doi.org/10.1023/A:1010933404324

30. T. Cover; P. Hart, Nearest neighbor pattern classification, *IEEE Trans. Inf. Theory*, **13** (1967), 21−27. https://doi.org/10.1109/TIT.1967.1053964
31. A. P. Davis, C. J. Grondin, R. J. Johnson, D. Sciaky, J. Wiegers, T. C. Wiegers, et al., Comparative toxicogenomics database (CTD): update 2021, *Nucleic Acids Res.*, **49** (2021), D1138−D1143. https://doi.org/10.1093/nar/gkaa891
32. A. P. Davis, C. G. Murphy, R. Johnson, J. M. Lay, K. Lennon-Hopkins, C. Saraceni-Richards, et al., The comparative toxicogenomics database: update 2013, *Nucleic Acids Res.*, **41** (2013), D1104−D1114. https://doi.org/10.1093/nar/gks994
33. C. J. Mattingly, M. C. Rosenstein, G. T. Colby, J. N. Forrest, J. L. Boyer, The comparative toxicogenomics database (CTD): a resource for comparative toxicological studies, *J. Exp. Zool. Part A: Comp. Exp. Biol.*, **305** (2006). https://doi.org/10.1002/jez.a.307
34. E. Sansone, F. G. De Natale, Z. H. Zhou, Efficient training for positive unlabeled learning, *IEEE Trans. Pattern Anal. Mach. Intell.*, **41** (2018), 2584−2598. https://doi.org/10.1109/TPAMI.2018.2860995
35. D. Weininger, SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules, *J. Chem. Inf. Comput. Sci.*, **28** (1988), 31−36. https://doi.org/10.1021/ci00057a005
36. X. Xiao, W. Zhu, B. Liao, J. Xu, C. Gu, B. Ji, et al., BPLLDA: predicting lncRNA-disease associations based on simple paths with limited lengths in a heterogeneous network, *Front. Genet.*, **9** (2018), 411. https://doi.org/10.3389/fgene.2018.00411
37. W. Ba-alawi, O. Soufan, M. Essack, P. Kalnis, V. B. Bajic, DASPfind: new efficient method to predict drug-target interactions, *J. Cheminf.*, **8** (2016), 15. https://doi.org/10.1186/s13321-016-0128-4
38. Z. H. You, Z. A. Huang, Z. Zhu, G. Y. Yan, Z. W. Li, Z. Wen, et al., PBMDA: a novel and effective path-based computational model for miRNA-disease association prediction, *PLoS Comput. Biol.*, **13** (2017), e1005455. https://doi.org/10.1371/journal.pcbi.1005455
39. J. Gao, B. Hu, L. Chen, A path-based method for identification of protein phenotypic annotations, *Curr. Bioinf.*, **16** (2021), 1214−1222. https://doi.org/10.2174/1574893616666210531100035
40. M. Jiang, B. Zhou, L. Chen, Identification of drug side effects with a path-based method, *Math. Biosci. Eng.*, **19** (2022), 5754−5771. https://doi.org/10.3934/mbe.2022269
41. H. Ogata, S. Goto, K. Sato, W. Fujibuchi, H. Bono, M. Kanehisa, KEGG: Kyoto encyclopedia of genes and genomes, *Nucleic Acids Res.*, **27** (1999), 29−34. https://doi.org/10.1093/nar/27.1.29
42. M. Kanehisa, M. Furumichi, Y. Sato, M. Ishiguro-Watanabe, M. Tanabe, KEGG: integrating viruses and cellular organisms. *Nucleic Acids Res.*, **49** (2021), D545−D551. https://doi.org/10.1093/nar/gkaa970
43. M. Kuhn, D. Szklarczyk, S. Pletscher-Frankild, T. H. Blicher, C. von Mering, L. J. Jensen, et al., STITCH 4: integration of protein–chemical interactions with user data, *Nucleic Acids Res.*, **42** (2014), D401−D407. https://doi.org/10.1093/nar/gkt1207
44. D. S. Wishart, Y. D. Feunang, A. C. Guo, E. J. Lo, A. Marcu, J. R. Grant, et al., DrugBank 5.0: a major update to the DrugBank database for 2018, *Nucleic Acids Res.*, **46** (2018), D1074−D1082. https://doi.org/10.1093/nar/gkx1037
45. M. Kuhn, M. Campillos, I. Letunic, L. J. Jensen, P. Bork, A side effect resource to capture phenotypic effects of drugs, *Mol. Syst. Biol.*, **6** (2010), 343. https://doi.org/10.1038/msb.2009.98

46. H. Cho, B. Berger, J. Peng, Compact integration of multi-network topology for functional analysis of genes, *Cell Syst.*, **3** (2016), 540−548. https://doi.org/10.1016/j.cels.2016.10.017
47. H. Tong, C. Faloutsos, J. Pan, Fast random walk with restart and its applications, in *Sixth International Conference on Data Mining (ICDM'06)*, (2006), 613−622. https://doi.org/10.1109/ICDM.2006.70
48. S. Kohler, S. Bauer, D. Horn, P. N. Robinson, Walking the interactome for prioritization of candidate disease genes, *AJHG*, **82** (2008), 949−958. https://doi.org/10.1016/j.ajhg.2008.02.013
49. R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, in *Proceedings of the 14th international joint conference on Artificial intelligence*, **2** (1995), 1137−1145.
50. E. Frank, M. Hall, L. Trigg, G. Holmes, I. H. Witten, Data mining in bioinformatics using Weka, *Bioinformatics*, **20** (2004), 2479−2481. https://doi.org/10.1093/bioinformatics/bth261
51. D. Powers, Evaluation: from precision, recall and f-measure to roc., informedness, markedness aand correlation, *J. Mach. Learn. Technol.*, **2** (2011), 37−63.
52. F. Huang, M. Fu, J. Li, L. Chen, K. Feng, T. Huang, et al., Analysis and prediction of protein stability based on interaction network, gene ontology, and KEGG pathway enrichment scores, *Biochim. Biophys. Acta, Proteins Proteomics*, **1871** (2023), 140889. https://doi.org/10.1016/j.bbapap.2023.140889
53. F. Huang, Q. Ma, J. Ren, J. Li, F. Wang, T. Huang, et al., Identification of smoking associated transcriptome aberration in blood with machine learning methods, *Biomed Res. Int.*, **2023** (2023), 5333361. https://doi.org/10.1155/2023/5333361
54. M. Onesime, Z. Yang, Q. Dai, Genomic island prediction via Chi-Square test and random forest algorithm, *Comput. Math. Methods Med.*, **2021** (2021), 9969751. https://doi.org/10.1155/2021/9969751
55. H. Wang, L. Chen, PMPTCE-HNEA: predicting metabolic pathway types of chemicals and enzymes with a heterogeneous network embedding algorithm, *Curr. Bioinf.*, **2023** (2023). https://doi.org/10.2174/1574893618666230224121633
56. C. Wu, L. Chen, A model with deep analysis on a large drug network for drug classification, *Math. Biosci. Eng.*, **20** (2023), 383−401. https://doi.org/10.3934/mbe.2023018
57. J. Ren, Y. Zhang, W. Guo, K. Feng, Y. Yuan, T. Huang, et al., Identification of genes associated with the impairment of olfactory and gustatory functions in COVID-19 via machine-learning methods, *Life*, **13** (2023), 798. https://doi.org/10.3390/life13030798
58. B. Matthews, Comparison of the predicted and observed secondary structure of T4 phage lysozyme, *Biochim. Biophys. Acta, Protein Struct.*, **405** (1975), 442−451. https://doi.org/10.1016/0005-2795(75)90109-9
59. Z. Cheng, K. Huang, Y. Wang, H. Liu, J. Guan, S. Zhou, Selecting high-quality negative samples for effectively predicting protein-RNA interactions, *BMC Syst. Biol.*, **11** (2017), 9. https://doi.org/10.1186/s12918-017-0390-8
60. X. Zhao, L. Chen, J. Lu, A similarity-based method for prediction of drug side effects with heterogeneous information, *Math. Biosci.*, **306** (2018), 136−144. https://doi.org/10.1016/j.mbs.2018.09.010
61. Y. Jia, R. Zhao, L. Chen, Similarity-based machine learning model for predicting the metabolic pathways of compounds, *IEEE Access*, **8** (2020), 130687−130696. https://doi.org/10.1109/ACCESS.2020.3009439

62. S. Zhou, S. Wang, Q. Wu, R. Azim, W. Li, Predicting potential miRNA-disease associations by combining gradient boosting decision tree with logistic regression, *Comput. Biol. Chem.*, **85** (2020), 107200. https://doi.org/10.1016/j.compbiolchem.2020.107200
63. Y. Zhao, X. Chen, J. Yin, Adaptive boosting-based computational model for predicting potential miRNA-disease associations, *Bioinformatics*, **35** (2019), 4730−4738. https://doi.org/10.1093/bioinformatics/btz297
64. F. Rayhan, S. Ahmed, S. Shatabda, D. M. Farid, Z. Mousavian, A. Dehzangi, et al., iDTI-ESBoost: identification of drug target interaction using evolutionary and structural features with boosting, *Sci. Rep.*, **7** (2017), 17731. https://doi.org/10.1038/s41598-017-18025-2
65. H. Liang, L. Chen, X. Zhao, X. Zhang, Prediction of drug side effects with a refined negative sample selection strategy, *Comput. Math. Methods Med.*, **2020** (2020), 1573543. https://doi.org/10.1155/2020/1573543
66. Z. Tian, Y. Yu, H. Fang, W. Xie, M. Guo, Predicting microbe–drug associations with structure-enhanced contrastive learning and self-paced negative sampling strategy, *Briefings Bioinf.*, **24** (2023), bbac634. https://doi.org/10.1093/bib/bbac634

Appendix

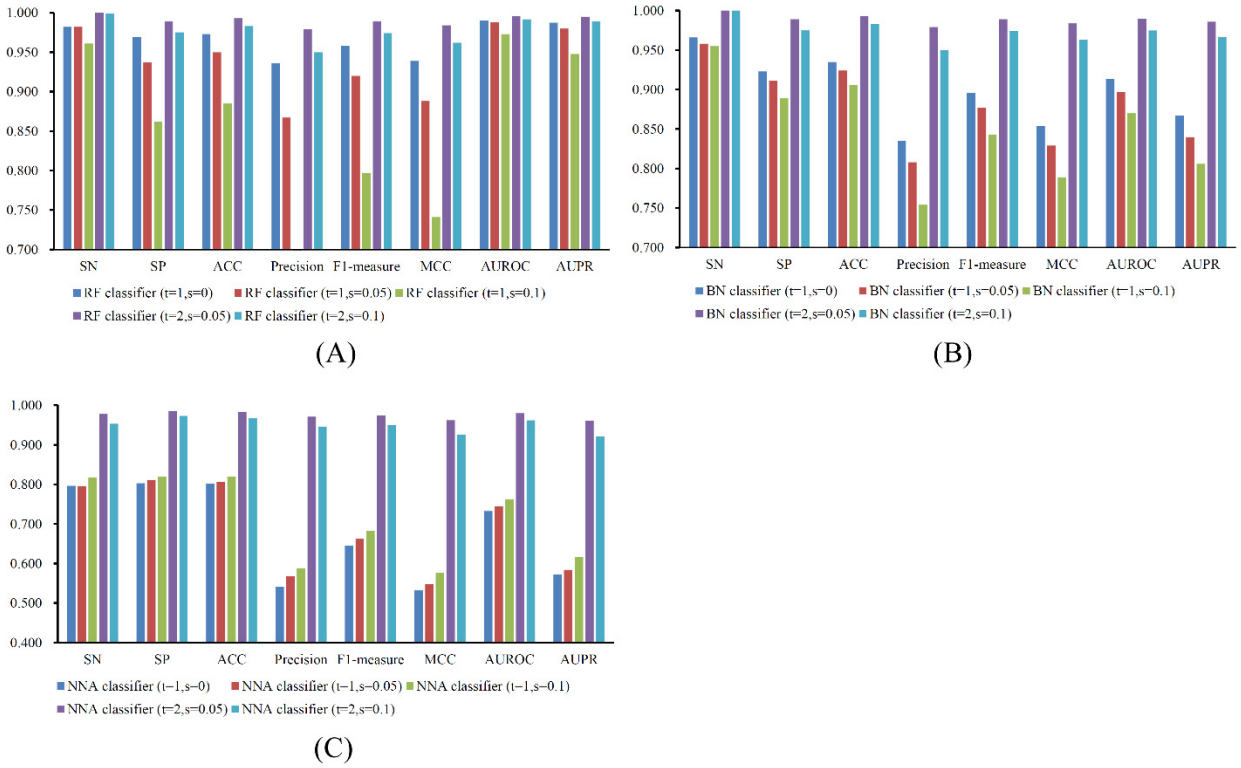

Figure A1. Comparison of three classifiers on imbalanced datasets containing negative samples selected by RNSS with different parameters, which are twice as many as positive samples. (A) Comparison of RF classifiers; (B) Comparison of BN classifiers; (C) Comparison of NNA classifiers. The parameter t represents the distance limitation for calculating level score and s indicates the threshold for constructing negative sample pool.

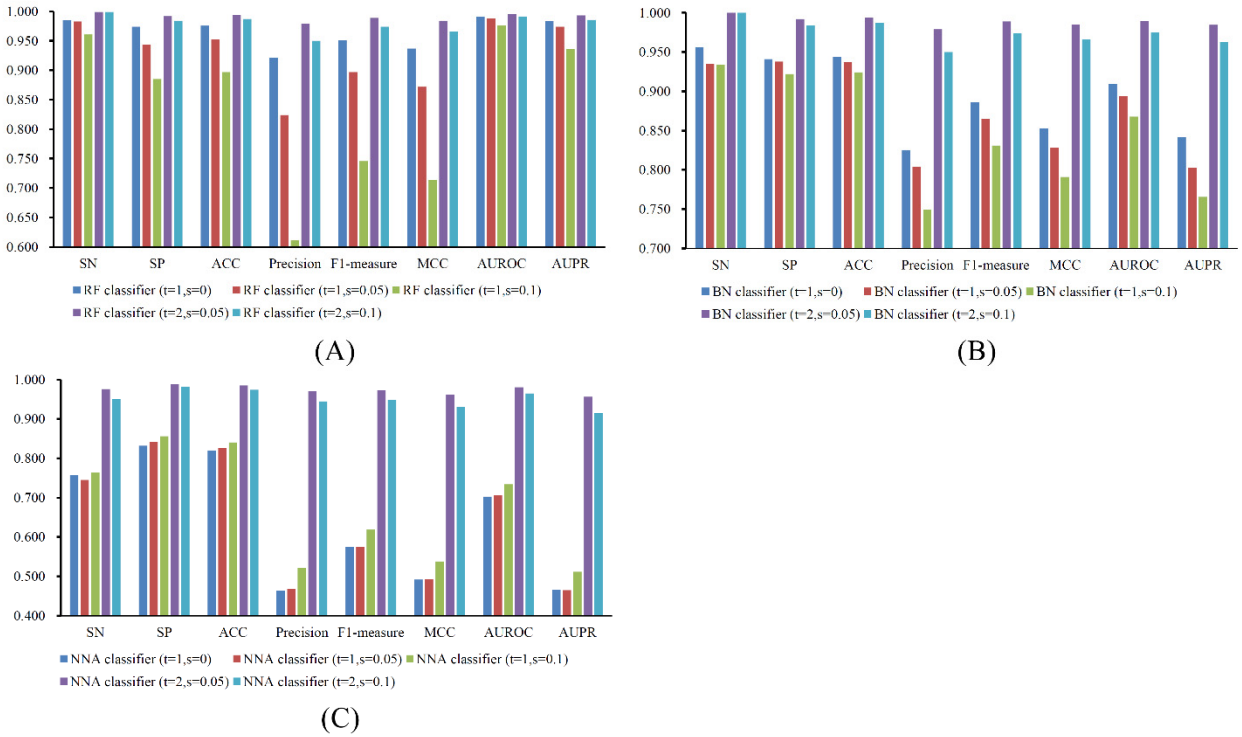

Figure A2. Comparison of three classifiers on imbalanced datasets containing negative samples selected by RNSS with different parameters, which are thrice as many as positive samples. (A) Comparison of RF classifiers; (B) Comparison of BN classifiers; (C) Comparison of NNA classifiers. The parameter t represents the distance limitation for calculating level score and s indicates the threshold for constructing negative sample pool.