Supplementary Materials

1. Performance evaluation for NASMDR

To evaluate the performance of the model, we conducted a five-fold cross-validation experiment on the ncDR dataset. When using the final descriptors as features, the current results were obtained as shown in the Fig.S1 and Table S1. As can be seen, NASMDR achieved an average accuracy of 0.8832 in the five experiments, while the AUC reached an average of 0.9468. NASMDR is an effective method for determining the presence of potential drug resistance in miRNAs in terms of precision, recall, and F1 score that measure the overall performance of the model.

Table S1. The five-fold cross-validation results performed by NASMDR on ncDR dataset.

Testing set	Accuracy	Precision	Recall	F1-score
1	0.8752	0.8776	0.8720	0.8748
2	0.8814	0.8742	0.8908	0.8824
3	0.8719	0.8754	0.8671	0.8712
4	0.8814	0.8876	0.8736	0.8806
5	0.9059	0.9145	0.8957	0.9050
Average	0.8832±0.0119	0.8859±0.0151	0.8798±0.0113	0.8828±0.0118

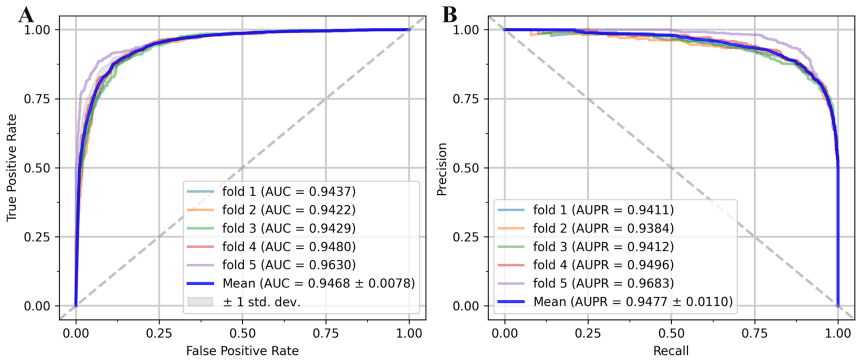

Fig S1. Performance evaluation for GCMDR. **A.** ROC curves performed by NASMDR on ncDR dataset. **B.** PR curves performed by NASMDR on ncDR dataset.

2. Comparison with miRNA and drug representation based on attribute character

with the development of technology and the accumulation of experience, some new drug representation techniques have appeared and been applied. According to their principles, these methods can be classified into three types: molecular fingerprint-based, neural network-based and collaborative-based. Molecular fingerprint-based method is achieved by searching for all substructures of a given step in all compounds, using one-hot encoding for the representation of compounds; Neural network-based approaches are also based on the structural information of the drug, and the two most common ideas are the use of natural language processing techniques (treating substructures as words and transforming them into word vectors) and the use of graph neural network techniques (using atoms as nodes to characterize the whole molecule). These approaches introduce association between substructures compared to molecular fingerprint-based approaches; The collaborative-based approach laterally characterizes the drug through relevant association information. These methods tend to rely heavily on existing experimental data and are hard to work if there are compounds whose properties have not been explored.

Therefore, three molecular fingerprint-based algorithms and two neural network-based algorithms with large differences in computational ideas were selected in this study. The details are as follows:

Extended Connectivity Fingerprint (ECFP) is a molecular fingerprint that can represent the internal structure of a compound, which often used as a baseline model/benchmark to compare with the effect of new methods in machine learning in recent years (Rogers and Hahn, 2010). The search algorithm (e.g. Morgan Algorithm) searches all compounds for all substructures of a given step, and the hash value of each substructure is obtained and the corresponding fingerprint is formed by hashing (Figueras, 1993). Here, the corresponding drug features are obtained using the calculation tool provided by RDKit (GetMorganFingerprintAsBitVect), which has three main parameters: the maximum diameter of the ECFPs, the fingerprint length and the identifier frequency (Landrum, 2013). The parameters used here are 3, 2048 and the default value, respectively.

Topological Atom Pairs Fingerprints (TPAPF) and Topological Atom Torsions Fingerprints (TPATF) are topological pharmacophore fingerprints (Carhart, et al., 1985; Kearsley, et al., 1996). TPAPF defines substructures called atom pairs and calculates drug characterization based on the atomic environment of all atom pairs in the topological representation of the chemical structure and the shortest path spacing between them. TPATF used topological torsion to characterize drugs using physicochemical

atom type descriptors. Here, we used the computational tools provided by MayaChemTools to implement the characterization of both molecular fingerprints (Sud, 2016).

Word2vec. Word2vec, as an input to the neural probabilistic language model, is actually a by-product of the neural probabilistic model. It is an intermediate result for learning the continuous bag-of-words model (CBOW) and skip-gram model through neural networks. CBOW is the model that predicts the probability of occurrence of the current word based on the contextual words. Its learning objective can be described as maximizing the log-likelihood function:

$$\mathcal{L} = \sum_{w \in C} \log p\big(w \big| Context(w)\big) \tag{1}$$

where w denotes the word in corpus C and w is the substructure of the drug in M2V. $Context(\cdot)$ represents the word vector of contextual words, and in M2V, the substructure. The CBOW output layer uses hierarchical softmax, so the conditional probability of w can be defined as:

$$p\big(w \big| Context(w)\big) = \prod_{j=2}^{l_w} p\big(d_j^w \big| X_w, \theta_{j-1}^w\big) \tag{2}$$

where l_w represents the number of nodes contained in the path, d_j^w is the encoding of the word w, and j represents the encoding corresponding to the jth node of the path in which w is located. θ_{j-1}^w is the parameter vector corresponding to the non-leaf node of the path in which w is located. Taking the log-likelihood of the objective function:

$$\mathcal{L} = \sum_{w \in C} \log p\big(w \big| Context(w)\big) \tag{3}$$

According to logistic regression we can get

$$\mathcal{L} = \sum_{w \in C} \log \prod_{j=2}^{l_w} \left[\left(\sigma\left({X_w}^T \theta_{j-1}^w\right)\right)^{1-d_j^w} \cdot \left(1 - \sigma\left({X_w}^T \theta_{j-1}^w\right)\right)^{d_j^w}\right]$$

$$= \sum_{w \in C} \sum_{j=2}^{l_w} \left[\left(1 - d_j^w\right) \cdot \log\left[\sigma\left({X_w}^T \theta_{j-1}^w\right)\right] + d_j^w \cdot \log\left[1 - \sigma\left({X_w}^T \theta_{j-1}^w\right)\right]\right] \tag{4}$$

The simplification yields

$$\mathcal{L}(w, j) = \left(1 - d_j^w\right) \cdot \log\left[\sigma\left({X_w}^T \theta_{j-1}^w\right)\right] + d_j^w \cdot \log\left[1 - \sigma\left({X_w}^T \theta_{j-1}^w\right)\right] \tag{5}$$

The Skip-gram model, on the other hand, uses a word as the input to predict the context. The difference from CBOW is that: 1) the input layer is no longer multiple word vectors, but one word vector; and 2) the projection layer directly passes the word vectors from the input layer to the output layer. Thus, the neural probabilistic language model can be described as:

$$p(Context(w)|w) = \prod_{u \in Context(w)} p(u|w) \tag{6}$$

where u is a word in w context, and since skip-gram is a bag-of-words model, each u is unordered and independent of each other. Based on hierarchical softmax, the conditional probability of each u can be expressed as:

$$p(u|w) = \prod_{j=2}^{l_u} p\big(d_j^u \big| V_w, \theta_{j-1}^w\big) \tag{7}$$

Similarly, it can be simplified to:

$$p\big(d_j^u \big| V_w, \theta_{j-1}^w\big) = \left(\sigma\left({V_w}^T \theta_{j-1}^u\right)\right)^{1-d_j^u} \cdot \left(1 - \sigma\left({V_w}^T \theta_{j-1}^u\right)\right)^{d_j^u} \tag{8}$$

Therefore, the objective function can be defined as:

$$\mathcal{L} = \sum_{w \in C} \log \prod_{u \in Context(w)} \prod_{j=2}^{l_w} \left[\left(\sigma\left({V_w}^T \theta_{j-1}^u\right)\right)^{1-d_j^u} \cdot \left(1 - \sigma\left({V_w}^T \theta_{j-1}^u\right)\right)^{d_j^u}\right]$$

$$= \sum_{w \in C} \sum_{u \in Context(w)} \sum_{j=2}^{l_u} \left[\left(1 - d_j^u\right) \cdot \log\left[\sigma\left({V_w}^T \theta_{j-1}^u\right)\right] + d_j^u \cdot \log\left[1 - \sigma\left({V_w}^T \theta_{j-1}^u\right)\right] \right] \quad (9)$$

The simplification yields

$$\mathcal{L}(w,u,j) = \left(1 - d_j^u\right) \cdot \log\left[\sigma\left({V_w}^T \theta_{j-1}^u\right)\right] + d_j^u \cdot \log\left[1 - \sigma\left({X_w}^T \theta_{j-1}^u\right)\right] \quad (10)$$

To verify the validity of the introduced individual feature de-scriptors. We conducted the following experiments: first, we validated the effect of five drug characterization methods (three molecular fingerprints: ECFP, TPAPF and TPATF, two neural network models mol2vec and GIN) and two miRNA characteri-zation methods (standardized k-mer and KSNMF). As shown in the Table S2, it was found that the performance of the models obtained by training the combination of the descriptors GIN+k-mer, TPAPF+k-mer, TPATF+k-mer, ECFP+KSNMF, GIN+ KSNMF, TPATF+ KSNMF, and Mol2Vec+KSNMF exceeded 0.93 (AUC) or 0.86 (Acc.) in five- fold cross-validation. Therefore, the pairs of feature descriptors are combined again and the model performance is shown in the Table 4. It can be seen that GIN+ M2V+ Kmer+ KSNMF obtains the best results, so in this paper we use this combination of features as the final descriptor for the training model.

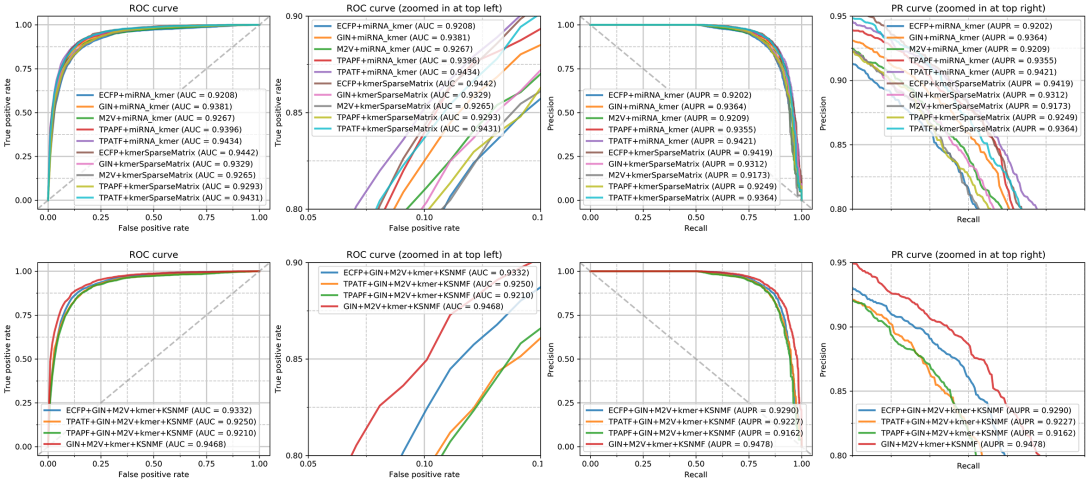

Fig S2. The performance of different feature descriptors. The upper half is the performance of Descriptor *ECFP+ K-mer, GIN+ K-mer, M2V+ K-mer, TPAPF+ K-mer, TPATF+ K-mer, ECFP+KSNMF, GIN+ KSNMF, M2V+ KSNMF, TPAPF+ KSNMF, TPATF+ KSNMF*; and the lower half is the performance of Descriptor *ECFP+GIN+M2V+K-mer+KSNMF, TPATF+GIN+M2V+K-mer+KSNMF, TPAPF+GIN+M2V+ K-mer+ KSNMF, GIN+M2V+ K-mer+ KSNMF*.

Table S2. Performance comparison of different descriptor pairs which are *ECFP+ K-mer, GIN+ K-mer, M2V+ K-mer, TPAPF+ K-mer, TPATF+ K-mer, ECFP+KSNMF, GIN+ KSNMF, M2V+ KSNMF, TPAPF+ KSNMF, TPATF+ KSNMF*.

Method	AUC	AUPR	Acc.	Pre.	Rec.	F1-score
ECFP+ K-mer	0.9208±0.0087	0.9193±0.0105	0.8509±0.0111	0.8707±0.0084	0.8242±0.0189	0.8467±0.0125
GIN+ K-mer	0.9381±0.0059	0.9366±0.0074	0.8688±0.0091	0.8801±0.007	0.8539±0.0151	0.8668±0.0099
M2V+ K-mer	0.9267±0.0055	0.9201±0.0079	0.8609±0.004	0.8493±0.0077	0.8777±0.0035	0.8632±0.0032
TPAPF+ K-mer	0.9396±0.0056	0.9344±0.0066	0.8759±0.0113	0.8773±0.0096	0.8742±0.0194	0.8756±0.012
TPATF+ K-mer	0.9434±0.0099	0.9409±0.0115	0.8799±0.0149	0.8717±0.0153	0.8909±0.0156	0.8812±0.0146
ECFP+KSNMF	0.9442±0.0042	0.9414±0.0046	0.8802±0.006	0.8717±0.0106	0.8919±0.0108	0.8816±0.0057
GIN+ KSNMF	0.9329±0.0049	0.9305±0.0078	0.8604±0.0068	0.8509±0.0097	0.8742±0.0159	0.8623±0.0073
M2V+ KSNMF	0.9265±0.0028	0.9162±0.0031	0.8582±0.0064	0.8614±0.009	0.8539±0.0155	0.8575±0.0072
TPAPF+ KSNMF	0.9293±0.0078	0.9235±0.0092	0.8568±0.0145	0.8300±0.0158	0.8976±0.0171	0.8624±0.0138

TPATF+ KSNMF	0.9431±0.0076	0.9351±0.0127	0.8739±0.0105	0.8666±0.013	0.8840±0.0134	0.8751±0.0103

Table S3. Performance comparison of different descriptor pairs which are *ECFP+GIN+M2V+k-mer+KSNMF, TPATF+GIN+M2V+k-mer+KSNMF, TPAPF+GIN+M2V+ k-mer+ KSNMF, GIN+M2V+ k-mer+ KSNMF.*

Method	AUC	AUPR	Acc.	Pre.	Rec.	F1-score
F1[a]	0.9327±0.0094	0.9262±0.0107	0.8672±0.0144	0.8492±0.0146	0.8931±0.0187	0.8705±0.0143
F2[b]	0.9210±0.0113	0.9150±0.0111	0.8592±0.0102	0.8417±0.0121	0.8849±0.0089	0.8627±0.0096
F3[c]	0.9250±0.0079	0.9222±0.0098	0.8592±0.0081	0.8662±0.0136	0.8498±0.0109	0.8578±0.0077
F4[d]	0.9468±0.0078	0.9477±0.0110	0.8832±0.0119	0.8859±0.0151	0.8798±0.0113	0.8828±0.0118

[a] F1 is *ECFP+GIN+M2V+k-mer+KSNMF*

[b] F2 is *TPATF+GIN+M2V+ k-mer+ KSNMF*

[c] F3 is *TPAPF+GIN+M2V+ k-mer+ KSNMF*

[d] F4 is *GIN+M2V+ k-mer+ KSNMF*

3. De novo experiments

As shown in the Table S3, the proposed model can still obtain the accuracy of more than 0.8 and 0.7 respectively. The results show the excellent performance of our framework under cold-start conditions. It is worth noting that the negative samples here are randomly selected. We sort miRNAs or drugs according to the frequency in the association, and gave priority to the miRNAs or drugs with fewer occurrences as the test samples, which is closer to the reality. To further test the predictive performance of the proposed model under cold start conditions, we conduct *de novo* experiments on each miRNA and drug. It can be seen from Fig. S3A and Fig. S3B that the medians of the AUC, AUPR and Pre all exceed 0.8. In all *de novo* experiments, the AUC and AUPR obtained in most experiments exceeded 0.8. About 60% of the experiments obtained accuracy of more than 0.65.

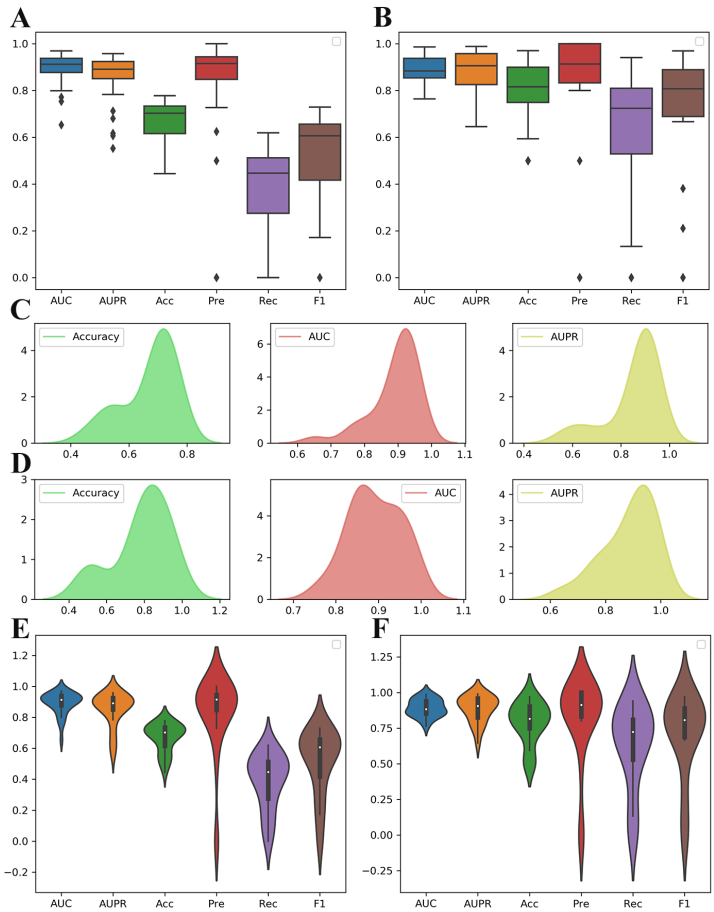

Fig S3. The performance of NASMDR in the *de novo* experiment on each miRNA and each drug. **A.** The boxplot of six evaluation indexes (drug); **B.** The boxplot of six evaluation indexes (miRNA); **C.** The density diagrams of Acc, AUC, AUPR (drug); **D.** The

density diagrams of Acc, AUC, AUPR (miRNA); **E.** The violin plot of six evaluation indexes (drug); **F.** The violin plot of six evaluation indexes (miRNA).

4. Comparison with different dimensionality-reduction methods

There are methods to reduce the dimensionality of large datasets to ensure computational efficiency, such as backselection, removal of variables exhibiting high correlation and high number of missing values. So far, the popular methods are principal component analysis and autoencoder. In order to obtain a more task-fitted abstract representation of the features, we constructed a six-layer Stacked autoencoder (SAE) to compare with the PCA method. As shown in Table S2, the PCA method obtained better results compared to the constructed stacked autoencoder. Since getting better abstract features by stacked autoencoder requires careful design, which is not the focus of the proposed model. Therefore in this paper we use PCA as the dimensionality-reduction method.

Table S4. Performance comparison of different dimensionality-reduction methods.

Method	AUC	AUPR	Acc.	Pre.	Rec.	F1-score
SAE	0.9239±0.0131	0.9174±0.0156	0.8637±0.0123	0.8611±0.0145	0.8675±0.0106	0.8643±0.0118
PCA	0.9468±0.0078	0.9477±0.0110	0.8832±0.0119	0.8859±0.0151	0.8798±0.0113	0.8828±0.0118

REFERENCES

Carhart, R.E., Smith, D.H. and Venkataraghavan, R. Atom pairs as molecular features in structure-activity studies: definition and applications. *Journal of Chemical Information and Computer Sciences* 1985;25(2):64-73.

Figueras, J. Morgan revisited. *Journal of chemical information and computer sciences* 1993;33(5):717-718.

Kearsley, S.K*., et al.* Chemical similarity using physiochemical property descriptors. *Journal of Chemical Information and Computer Sciences* 1996;36(1):118-127.

Landrum, G. Rdkit documentation. *Release* 2013;1(1-79):4.

Rogers, D. and Hahn, M. Extended-connectivity fingerprints. *Journal of chemical information and modeling* 2010;50(5):742-754.

Sud, M. MayaChemTools: an open source package for computational drug discovery. *Journal of chemical information and modeling* 2016;56(12):2292-2297.