Supplementary Information for

"wSDTNBI: a novel network-based inference method for virtual screening"

Contents

Supplementary Methods

Description of resource-spreading processes

To describe the four network-based methods used in this study (*i.e.*, NBI, SDTNBI, bSDTNBI and wSDTNBI) in a uniform way, it is necessary to focus on their common features. As mentioned in our previous studies,[1-5] NBI calculates prediction scores by simulating resource-spreading processes on a given DTI network, while SDTNBI and bSDTNBI calculate prediction scores by simulating resource-spreading processes on a given substructure-drug-target network. The newly developed wSDTNBI is based on bSDTNBI, and hence also based on resource-spreading processes. It is obvious that performing resource-spreading processes on networks is one of the most common features of the four network-based methods. Mathematically, resource-spreading processes can be described by matrix multiplication:

$$C = A \times W^k \tag{1}$$

Among the three matrices in this equation, A is the initial resource matrix describing the state of resource allocation before resource-spreading processes, W is the transfer matrix describing a resource-spreading process, k is the number of resource-spreading processes, and C is the final resource matrix describing the state of resource allocation after resource-spreading processes. As mentioned in our previous studies,[2-5] the value of k is set as 2 in this study.

To ensure the total amount of resources remains constant in resource-spreading processes, the sum of each row in W should be normalized to 1 before resource-spreading processes. It is noteworthy that rows full of zeros should not be normalized to avoid division by zero. Assuming that B is a non-normalized transfer matrix, the normalization of B and generation of W can be described as:

$$W(i,j) = \begin{cases} \dfrac{B(i,j)}{\sum_l B(i,l)} & if \sum_l B(i,l) \neq 0 \\ 0 & otherwise \end{cases} \tag{2}$$

For convenience of use, the aforementioned matrix normalization and matrix multiplication are combined and expressed as a function $f(A,B)$, which outputs a final

resource matrix after inputting an initial resource matrix A and a non-normalized transfer matrix B. The details of this function can be presented in pseudo code:

```
Matrix f(Matrix A, Matrix B) {
    for (int i : B.rows()) { // for each row in B
        float sum = 0;
        for (int j : B.columns()) {
            sum += B(i, j);
        }
        if (sum != 0) { // to avoid division by zero
            for (int j : B.columns()) {
                B(i, j) /= sum;
            }
        }
    }
    return A × B × B; // k = 2
}
```

With the support of this function, the four network-based methods are described in order from simple to complex, namely "NBI → SDTNBI → bSDTNBI → wSDTNBI" in the following sections.

Calculation of NBI scores

Among the four network-based methods, NBI is the simplest one. An unweighted DTI network is used as the only input.[1]

Denoting zero matrices as O, the initial resource matrix is defined as:

$$A_{NBI} = \begin{bmatrix} O & A_{DTI} \\ (A_{DTI})^T & O \end{bmatrix} \tag{3}$$

Then, the non-normalized transfer matrix is simply defined as:

$$B_{NBI} = A_{NBI} \tag{4}$$

Lastly, the final resource matrix is calculated as:

$$C_{NBI} = f(A_{NBI}, B_{NBI}) \tag{5}$$

The value of $C_{NBI}(i, N_D + j)$ is the NBI score for the interaction between drug D_i and target T_j. From these equations, it is easy to find that NBI only relies on the topological

information of the inputted DTI network and hence cannot predict potential targets for novel compounds outside the DTI network.

Calculation of SDTNBI scores

As an improved version of NBI, SDTNBI is more complex. Both an unweighted DTI network and a DSA network are used as inputs.[2] Substructures in the DSA network are used to bridge the gap between the DTI network and novel compounds outside the DTI network. Hence, SDTNBI can predict potential targets for not only drugs in the DTI network but also novel compounds.

At first, the initial resource matrix is defined as:

$$A_{SDTNBI} = \begin{bmatrix} O & A_{DSA} & A_{DTI} \\ (A_{DSA})^T & O & O \\ (A_{DTI})^T & O & O \end{bmatrix} \tag{6}$$

Then, two matrices similar to A_{DTI} and A_{DSA} are defined as:

$$B_{DTI}(i,j) = \begin{cases} A_{DTI}(i,j) & if \sum_{l=1}^{N_T} A_{DTI}(i,l) \neq 0 \\ 0 & otherwise \end{cases} \tag{7}$$

$$B_{DSA}(i,j) = \begin{cases} A_{DSA}(i,j) & if \sum_{l=1}^{N_T} A_{DTI}(i,l) \neq 0 \\ 0 & otherwise \end{cases} \tag{8}$$

Compared with A_{DTI} and A_{DSA}, the rows corresponding to novel compounds (*i.e.*, compounds without any known targets) in B_{DTI} and B_{DSA} are set to zero to prevent resources from spreading back to the nodes representing novel compounds in resource-spreading processes. It is obvious that B_{DTI} is always equal to A_{DTI}, while B_{DSA} is equal to A_{DSA} only if there is no novel compound.

Based on these two matrices, the non-normalized transfer matrix B_{SDTNBI} is generated as:

$$B_{SDTNBI} = \begin{bmatrix} O & B_{DSA} & B_{DTI} \\ (B_{DSA})^T & O & O \\ (B_{DTI})^T & O & O \end{bmatrix} \tag{9}$$

The generation of B_{SDTNBI} can also be viewed as a transformation from A_{SDTNBI} to

B_{SDTNBI}. To describe the transformation in more details, three sets are defined. Among them, I_D is a set containing the indices of the rows corresponding to the N_D drugs in A_{SDTNBI}, namely all integers in the interval $(0,N_D]$. The row indices in this set are also the column indices of the N_D drugs in A_{SDTNBI}, because A_{SDTNBI} is a symmetric matrix. Similarly, I_S is a set containing the indices corresponding to the N_S substructures in A_{SDTNBI}, namely all integers in the interval $(N_D,N_D+N_S]$. I_T is the set containing the indices corresponding to the N_T targets in A_{SDTNBI}, namely all integers in the interval $(N_D+N_S,N_D+N_S+N_T]$. Using these three sets, the transformation from A_{SDTNBI} to B_{SDTNBI} can be presented in pseudo code:

```
Matrix B_SDTNBI = A_SDTNBI;
for (int i : I_D) {
    float sum = 0;
    for (int j : I_T) { // to count the number of known targets
        sum += A_SDTNBI(i, j);
    }
    if (sum == 0) { // if this is a novel compound
        for (int l : B_SDTNBI.columns()) {
            B_SDTNBI(i, l) = 0;
            B_SDTNBI(l, i) = 0;
        }
    }
}
```

Lastly, the final resource matrix is calculated as:

$$C_{SDTNBI} = f(A_{SDTNBI},B_{SDTNBI}) \tag{10}$$

The value of $C_{SDTNBI}(i,N_D+N_S+j)$ is the SDTNBI score for the interaction between drug D_i and target T_j.

Calculation of bSDTNBI scores

As an improved version of SDTNBI, bSDTNBI is more complex. Three tunable parameters $\alpha \in [0,1]$, $\beta \in [0,1]$ and $\gamma \in (-\infty,+\infty)$ are introduced to address potential imbalances and improve performance.[3-5]

At first, two matrices are defined using A_{DTI}, A_{DSA} and parameter α:

$$A_{DSA}^{'}(i,j) = \alpha \cdot \frac{A_{DSA}(i,j)}{\sum_{l=1}^{N_S} A_{DSA}(i,l)} \tag{11}$$

$$A_{DTI}^{'}(i,j) = (1-\alpha) \cdot \frac{A_{DTI}(i,j)}{\sum_{l=1}^{N_T} A_{DTI}(i,l)} \tag{12}$$

Based on these two matrices, the initial resource matrix is defined as:

$$A_{bSDTNBI} = \begin{bmatrix} O & A_{DSA}^{'} & A_{DTI}^{'} \\ \left(A_{DSA}^{'}\right)^T & O & O \\ \left(A_{DTI}^{'}\right)^T & O & O \end{bmatrix} \tag{13}$$

From these equation, it is easy to find that parameter α is used to adjust the initial resources allocated to different types of nodes. A smaller α value means that less resources will be allocated to substructure nodes and more resources will be allocated to target nodes. A larger α value means that more resources will be allocated to substructure nodes and less resources will be allocated to target nodes.

The generation of $A_{bSDTNBI}$ can also be viewed as a transformation from the aforementioned A_{SDTNBI} to $A_{bSDTNBI}$. The details of this transformation can be presented in pseudo code:

```
Matrix AbSDTNBI = ASDTNBI;
for (int i : ID) {
    float sum = 0;
    for (int j : IS) {
        sum += AbSDTNBI(i, j);
    }
    if (sum != 0) {
        for (int j : IS) { // for each DSA
            AbSDTNBI(i, j) *= α / sum;
        }
    }
    sum = 0;
    for (int j : IT) {
        sum += AbSDTNBI(i, j);
    }
    if (sum != 0) {
        for (int j : IT) { // for each DTI
```

```
            A_bSDTNBI(i, j) *= (1 - α) / sum;
        }
    }
}
```

Then, the non-normalized transfer matrix is generated using the aforementioned B_{DTI}, B_{DSA} and the other two parameters β and γ:

$$B_{bSDTNBI}^{'} = \begin{bmatrix} O & \beta \cdot B_{DSA} & (1-\beta) \cdot B_{DTI} \\ \beta \cdot (B_{DSA})^T & O & O \\ (1-\beta) \cdot (B_{DTI})^T & O & O \end{bmatrix} \tag{14}$$

$$B_{bSDTNBI}(i,j) = B_{bSDTNBI}^{'}(i,j) \cdot \left[\sum_{l=1}^{N_D + N_S + N_T} B_{bSDTNBI}^{'}(l,j) \right]^{\gamma} \tag{15}$$

Among the two parameters, β is used to adjust the weights of different types of edges. A smaller β value means that smaller weights will be assigned to the edges representing DSAs and larger weights will be assigned to the edges representing DTIs. A larger β value means that larger weights will be assigned to the edges representing DSAs and less weights will be assigned to the edges representing DTIs. Meanwhile, parameter γ is used to adjust the influence of hub nodes (*i.e.*, nodes with high degree). A positive γ value means that the influence of hub nodes will be strengthen, while a negative γ value means that the influence of hub nodes will be weaken.

The generation of $B_{bSDTNBI}$ can also be viewed as a transformation from the aforementioned B_{SDTNBI} to $B_{bSDTNBI}$. The details of this transformation be presented in pseudo code:

```
Matrix B_bSDTNBI = B_SDTNBI;
for (int i : I_D) {
    for (int j : I_S) { // for each DSA
        B_bSDTNBI(i, j) *= β;
        B_bSDTNBI(j, i) *= β;
    }
    for (int j : I_T) { // for each DTI
        B_bSDTNBI(i, j) *= 1 - β;
        B_bSDTNBI(j, i) *= 1 - β;
    }
}
for (int j : B_bSDTNBI.columns()) {
```

```
    float sum = 0;
    for (int i : B_bSDTNBI.rows()) {
        sum += B_bSDTNBI(i, j);
    }
    if (sum != 0) {
        for (int i : B_bSDTNBI.rows()) {
            B_bSDTNBI(i, j) *= pow(sum, γ);
        }
    }
}
```

Lastly, the final resource matrix is calculated as:

$$C_{bSDTNBI} = f(A_{bSDTNBI}, B_{bSDTNBI}) \tag{16}$$

The value of $C_{bSDTNBI}(i, N_D + N_S + j)$ is the bSDTNBI score for the interaction between drug D_i and target T_j.

Calculation of wSDTNBI scores

As mentioned in the main text, wSDTNBI scores are calculated based on bSDTNBI scores. At first, the normalization of bSDTNBI scores in $C_{bSDTNBI}$ can be presented in pseudo code:

```
Matrix C_wSDTNBI = C_bSDTNBI;
for (int i : I_D) {
    int n = 0;
    vector<float> v;
    for (int j : I_T) { // to count the number of known targets
        if (W_DTI(i, j) != 0) {
            ++n;
        }
        v.emplace_back(C_wSDTNBI(i, j));
    }
    stable_sort(v.begin(), v.end(), greater<float>());
    const float threshold = v[n + δ]; // the (n + δ)-th largest
    for (int j : I_T) {
        if (C_wSDTNBI(i, j) > threshold) {
            C_wSDTNBI(i, j) = 1;
        } else {
```

```
            C_wSDTNBI(i, j) /= threshold;
        }
    }
}
```

In this step, parameter δ is used to adjust the distribution of normalized scores. Specifically, a smaller δ value means that less scores will be normalized to 1, while a larger δ value means that more scores will be normalized to 1. After this step, the value of $C_{wSDTNBI}(i,N_D+N_S+j)$ is the normalized bSDTNBI score for the interaction between drug D_i and target T_j.

Then, considering that A_{DSA} contains the molecular fingerprints of all drugs, the Tanimoto coefficients of all possible drug pairs can be calculated as:

$$D = A_{DSA} \times (A_{DSA})^T \tag{17}$$

$$T_C(i,j) = \frac{D(i,j)}{D(i,i) + D(j,j) - D(i,j)} \tag{18}$$

The value of $T_C(i,j)$ is the Tanimoto coefficient between two drugs D_i and D_j, which measures the drug similarity between these two drugs.

Lastly, using drug similarity matrix T_C in combination with W_{DTI}, the normalized bSDTNBI scores in $C_{wSDTNBI}$ can be further transformed into wSDTNBI scores. The details of this transformation can be presented in pseudo code:

```
for (int j : I_T) {
    vector<int> I_L;
    for (int i : I_D) {
        if (W_DTI(i, j) != 0) {
            I_L.emplace_back(i);
        }
    }
    for (int i : I_D) { // to select ε reference ligands
        set<pair<float, float>> pair_set;
        for (int l : I_L) {
            const auto pair = make_pair(T_c(i, l), W_DTI(l, j));
            if (pair_set.size() < ε) {
                pair_set.emplace(pair);
            } else if (pair.first > pair_set.begin()->first) {
                pair_set.erase(pair_set.begin());
```

```
                pair_set.emplace(pair);
            }
        }
        if (!pair_set.empty()) {
            float sum = 0;
            for (const auto pair : pair_set) {
                sum += pair.second;
            }
            // wSDTNBI score = normalized bSDTNBI score
            //               × similarity-based score
            C_wSDTNBI(i, j) *= sum / pair_set.size();
        } else {
            C_wSDTNBI(i, j) = 0;
        }
    }
}
```

In this step, parameter ε is used to adjust the number of reference ligands, which is similar to the parameter *K* in *K*-nearest neighbor method. After this step, the value of $C_{wSDTNBI}(i,N_D+N_S+j)$ is the wSDTNBI score for the interaction between drug D_i and target T_j.

References

1 F. X. Cheng, C. Liu, J. Jiang, W. Q. Lu, W. H. Li, G. X. Liu, W. X. Zhou, J. Huang and Y. Tang, *PLoS Comput. Biol.*, 2012, **8**, e1002503.

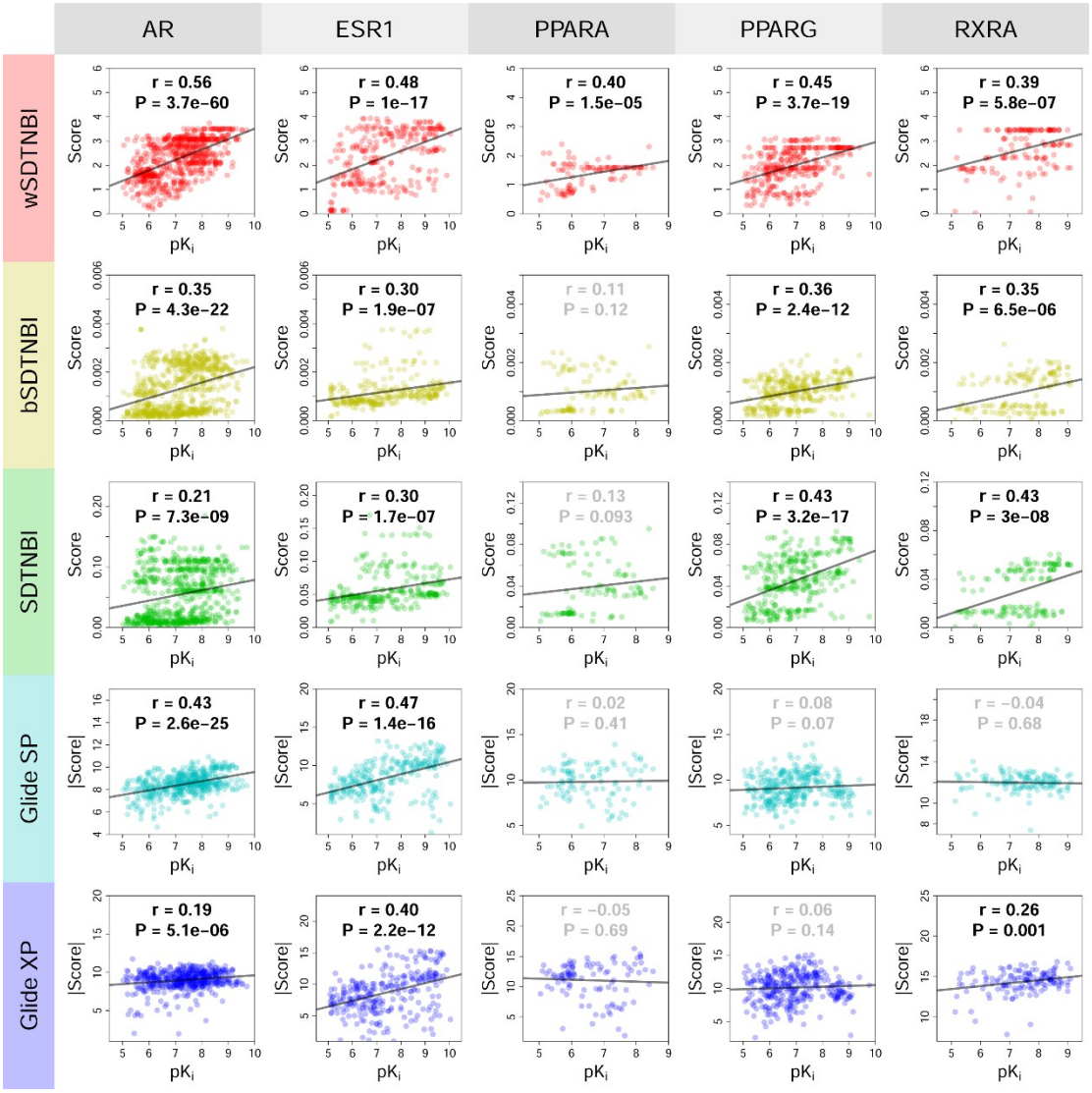
2 Z. R. Wu, F. X. Cheng, J. Li, W. H. Li, G. X. Liu and Y. Tang, *Brief. Bioinform.*, 2017, **18**, 333-347.

3 Z. R. Wu, W. Q. Lu, D. Wu, A. Q. Luo, H. P. Bian, J. Li, W. H. Li, G. X. Liu, J. Huang, F. X. Cheng and Y. Tang, *Br. J. Pharmacol.*, 2016, **173**, 3372-3385.

4 J. S. Fang, Z. R. Wu, C. P. Cai, Q. Wang, Y. Tang and F. X. Cheng, *J. Chem Inf. Model.*, 2017, **57**, 2657-2671.

5 Z. R. Wu, W. Q. Lu, W. W. Yu, T. D. Y. Wang, W. H. Li, G. X. Liu, H. K. Zhang, X. F. Pang, J. Huang, M. Y. Liu, F. X. Cheng and Y. Tang, *Pharmacol. Res.*, 2018, **129**, 400-413.

Supplementary Figure



Supplementary Fig. S1. The correlation between prediction scores and pK_i values in external validation. Data points calculated by wSDTNBI, bSDTNBI, SDTNBI, Glide SP and Glide XP were colored by red, yellow, green, cyan and blue, respectively. Pearson correlation coefficients (*r*) and *P* values were calculated by the Pearson correlation test in R (version 3.3.3).

Supplementary Tables

Supplementary Table S1. The statistics of the DSA networks constructed by generating different types of molecular fingerprints for the drugs in local and global DTI networks.

Network	Fingerprint	N_D	N_S	N_{DSA}	Sparsity (%)
Local	ECFP_0	5,560	69	51,163	13.34
	ECFP_2	5,560	2,878	146,087	0.91
	ECFP_4	5,560	24,444	241,833	0.18
	ECFP_6	5,560	66,130	332,205	0.09
	FCFP_0	5,560	13	26,396	36.52
	FCFP_2	5,560	377	100,713	4.80
	FCFP_4	5,560	8,172	188,833	0.42
	FCFP_6	5,560	36,915	277,353	0.14
	FP4	5,560	173	79,441	8.26
	KR	5,560	2,411	294,385	2.20
	MACCS	5,560	156	255,639	29.47
	PubChem	5,560	683	757,050	19.94
Global	ECFP_0	12,751	92	116,976	9.97
	ECFP_2	12,751	4,061	336,582	0.65
	ECFP_4	12,751	44,076	559,884	0.10
	ECFP_6	12,751	132,363	770,680	0.05
	FCFP_0	12,751	19	60,407	24.93
	FCFP_2	12,751	480	231,821	3.79
	FCFP_4	12,751	13,180	437,364	0.26
	FCFP_6	12,751	70,859	644,046	0.07
	FP4	12,751	190	183,142	7.56
	KR	12,751	2,868	651,086	1.78
	MACCS	12,751	157	582,725	29.11
	PubChem	12,751	706	1,775,375	19.72

N_D: the number of drugs, N_S: the number of substructures, N_{DSA}: the number of DSAs. The network sparsity is the ratio of the number of known edges (*i.e.*, N_{DSA}) to the

number of all possible edges (*i.e.*, $N_D \times N_T$).

Supplementary Table S2. The optimized parameters for bSDTNBI and wSDTNBI models.

Network	Fingerprint	α	β	γ	δ	ε
Local	ECFP_0	0.2	0.9	-0.7	20	3
	ECFP_2	0.3	0.1	-0.6	20	3
	FCFP_0	0.1	0.1	-0.6	20	3
	FCFP_2	0.3	0.6	-0.7	20	3
	FCFP_4	0.4	0.1	-0.6	20	3
	FP4	0.3	0.8	-0.7	20	3
	KR	0.3	0.1	-0.6	20	3
	MACCS	0.2	0.2	-0.7	20	3
	PubChem	0.4	0.1	-0.7	20	3
Global	ECFP_0	0.2	0.9	-0.6	20	4
	ECFP_2	0.3	0.3	-0.5	20	4
	FCFP_0	0.2	0.9	-0.6	20	4
	FCFP_2	0.4	0.9	-0.6	20	4
	FCFP_4	0.4	0.2	-0.5	20	4
	FP4	0.3	0.9	-0.6	20	4
	KR	0.4	0.3	-0.6	20	4
	MACCS	0.4	0.9	-0.7	20	4
	PubChem	0.5	0.4	-0.7	20	4

bSDTNBI models only use α, β and γ, while wSDTNBI models use all the five parameters.

Supplementary Table S3. The evaluation indicators of local models in 10-fold cross validation.

Method	Fingerprint	R (L = 20)	e_R (L = 20)	AUROC	r	MAE	RMSE
wSDTNBI	ECFP_0	0.550±0.012	25.9±0.6	0.953±0.002	0.458±0.020	0.88±0.02	1.17±0.02
	ECFP_2	0.592±0.013	27.9±0.6	0.959±0.002	0.555±0.017	0.80±0.01	1.07±0.02
	FCFP_0	0.515±0.012	24.3±0.6	0.951±0.002	0.319±0.024	0.99±0.02	1.30±0.02
	FCFP_2	0.558±0.013	26.3±0.6	0.954±0.002	0.512±0.022	0.83±0.02	1.11±0.02
	FCFP_4	0.588±0.013	27.7±0.7	0.959±0.002	0.557±0.020	0.80±0.02	1.07±0.02
	FP4	0.555±0.011	26.2±0.6	0.953±0.002	0.474±0.020	0.86±0.02	1.15±0.02
	KR	0.579±0.013	27.3±0.6	0.958±0.002	0.529±0.021	0.82±0.02	1.09±0.02
	MACCS	0.555±0.013	26.2±0.6	0.951±0.002	0.487±0.020	0.85±0.02	1.14±0.02
	PubChem	0.534±0.012	25.2±0.6	0.947±0.002	0.487±0.020	0.85±0.02	1.14±0.02
bSDTNBI	ECFP_0	0.818±0.009	38.6±0.5	0.975±0.002	0.139±0.026	1.86±0.03	2.22±0.03
	ECFP_2	0.886±0.007	41.8±0.4	0.982±0.002	0.136±0.025	1.87±0.03	2.23±0.03
	FCFP_0	0.797±0.009	37.6±0.5	0.972±0.002	0.153±0.025	1.84±0.03	2.20±0.03
	FCFP_2	0.847±0.008	39.9±0.4	0.979±0.002	0.134±0.025	1.86±0.03	2.23±0.03
	FCFP_4	0.894±0.007	42.2±0.4	0.983±0.002	0.123±0.024	1.87±0.03	2.24±0.03
	FP4	0.830±0.008	39.2±0.4	0.977±0.002	0.141±0.026	1.86±0.03	2.23±0.03
	KR	0.869±0.008	41.0±0.4	0.981±0.002	0.121±0.023	1.87±0.03	2.23±0.03
	MACCS	0.814±0.009	38.4±0.5	0.975±0.002	0.140±0.025	1.86±0.03	2.22±0.03
	PubChem	0.816±0.009	38.5±0.5	0.976±0.002	0.123±0.025	1.87±0.03	2.23±0.03
SDTNBI	ECFP_0	0.754±0.010	35.6±0.5	0.968±0.002	-0.010±0.022	1.75±0.03	2.14±0.03
	ECFP_2	0.793±0.010	37.4±0.5	0.974±0.002	0.051±0.021	1.77±0.03	2.15±0.03
	FCFP_0	0.757±0.010	35.7±0.5	0.966±0.002	0.003±0.022	1.72±0.03	2.11±0.03
	FCFP_2	0.743±0.010	35.1±0.5	0.969±0.002	-0.001±0.021	1.77±0.03	2.15±0.03
	FCFP_4	0.821±0.009	38.7±0.5	0.975±0.002	0.105±0.021	1.77±0.03	2.15±0.03
	FP4	0.745±0.010	35.1±0.5	0.968±0.002	-0.008±0.021	1.77±0.03	2.15±0.03
	KR	0.725±0.012	34.2±0.6	0.969±0.002	0.034±0.021	1.79±0.03	2.17±0.03
	MACCS	0.619±0.011	29.2±0.5	0.956±0.002	-0.011±0.021	1.81±0.03	2.19±0.03
	PubChem	0.544±0.011	25.7±0.5	0.947±0.003	0.009±0.023	1.84±0.03	2.21±0.03
NBI		0.901±0.007	42.5±0.4	0.981±0.002	0.115±0.024	1.55±0.03	1.94±0.03

Supplementary Table S4. The evaluation indicators of global models in 10-fold cross validation.

Method	Fingerprint	R (L = 20)	e_R (L = 20)	AUROC	r	MAE	RMSE
wSDTNBI	ECFP_0	0.459±0.009	41.5±0.8	0.950±0.002	0.469±0.012	0.84±0.01	1.12±0.02
	ECFP_2	0.509±0.008	46.0±0.8	0.957±0.001	0.545±0.010	0.78±0.01	1.04±0.01
	FCFP_0	0.404±0.008	36.5±0.7	0.945±0.002	0.372±0.014	0.91±0.01	1.21±0.02
	FCFP_2	0.459±0.009	41.5±0.8	0.950±0.002	0.514±0.012	0.80±0.01	1.07±0.01
	FCFP_4	0.510±0.009	46.1±0.8	0.958±0.001	0.552±0.011	0.77±0.01	1.04±0.01
	FP4	0.460±0.010	41.5±0.9	0.950±0.002	0.485±0.012	0.82±0.01	1.10±0.02
	KR	0.478±0.008	43.2±0.8	0.953±0.001	0.523±0.011	0.80±0.01	1.06±0.02
	MACCS	0.434±0.008	39.2±0.8	0.943±0.002	0.488±0.012	0.82±0.01	1.10±0.02
	PubChem	0.422±0.009	38.2±0.8	0.940±0.002	0.493±0.012	0.82±0.01	1.09±0.02
bSDTNBI	ECFP_0	0.701±0.007	63.3±0.7	0.968±0.001	0.149±0.015	1.61±0.02	2.00±0.02
	ECFP_2	0.780±0.007	70.6±0.7	0.975±0.001	0.146±0.015	1.62±0.02	2.01±0.02
	FCFP_0	0.675±0.008	61.0±0.8	0.965±0.001	0.155±0.015	1.61±0.02	2.00±0.02
	FCFP_2	0.731±0.007	66.0±0.7	0.972±0.001	0.135±0.015	1.62±0.02	2.01±0.02
	FCFP_4	0.799±0.007	72.2±0.7	0.976±0.001	0.145±0.015	1.62±0.02	2.01±0.02
	FP4	0.711±0.007	64.2±0.7	0.970±0.001	0.149±0.015	1.62±0.02	2.00±0.02
	KR	0.759±0.008	68.7±0.7	0.974±0.001	0.125±0.014	1.62±0.02	2.01±0.02
	MACCS	0.687±0.008	62.1±0.8	0.968±0.001	0.135±0.015	1.62±0.02	2.01±0.02
	PubChem	0.690±0.008	62.4±0.8	0.969±0.001	0.129±0.015	1.62±0.02	2.01±0.02
SDTNBI	ECFP_0	0.641±0.009	58.0±0.8	0.961±0.002	0.054±0.014	1.51±0.02	1.92±0.02
	ECFP_2	0.699±0.007	63.2±0.7	0.967±0.001	0.065±0.014	1.53±0.02	1.94±0.02
	FCFP_0	0.640±0.009	57.9±0.8	0.960±0.002	0.070±0.013	1.50±0.02	1.90±0.02
	FCFP_2	0.644±0.007	58.2±0.7	0.963±0.001	0.047±0.014	1.53±0.02	1.94±0.02
	FCFP_4	0.731±0.007	66.1±0.7	0.969±0.001	0.094±0.014	1.53±0.02	1.93±0.02
	FP4	0.633±0.008	57.3±0.8	0.961±0.001	0.048±0.014	1.52±0.02	1.93±0.02
	KR	0.631±0.009	57.1±0.8	0.963±0.001	0.048±0.014	1.55±0.02	1.95±0.02
	MACCS	0.512±0.008	46.3±0.8	0.950±0.002	0.027±0.014	1.56±0.02	1.96±0.02
	PubChem	0.401±0.008	36.2±0.7	0.941±0.002	0.020±0.014	1.59±0.02	1.98±0.02
NBI		0.795±0.007	71.9±0.7	0.969±0.002	0.174±0.015	1.37±0.02	1.77±0.02

Supplementary Table S5. The statistics of our cocrystal structure.

Ligand name		Ursonic acid
PDB No.		6J3N
Data collection		
	Space group	$P4_12_12$
	Cell dimensions	
	a, b, c (Å)	61.93 61.93 155.87
	α, β, γ (°)	90.0 90.0 90.0
	Temperature (K)	100
	Wavelength (Å)	0.979176
	Resolution (Å)	61.93 - 1.99
	No. reflections	21604 (2098)
	I / σI	19.600 (43.000)
	Completeness (%)	99.9 (99.8)
	Multiplicity	24.8 (19.1)
	Rmerge	0.107 (0.050)
Refinement		
	Resolution (Å)	48.49 - 1.99
	No. reflections	21604 (2098)
	Rwork / Rfree	0.1992 / 0.2333
	No. atoms	
	Protein	2048
	Ligand / ion	33
	Water	82
	R.m.s. deviations	
	Bond lengths (Å)	0.41
	Bond angles (°)	0.002
	Ramachandran statistics	
	Favoured	246
	Allowed	4
	Outliers	0

Supplementary Table S6. The results of the method comparison using our newly discovered RORγt ligands.

Method type	Method name and parameters	U*	O*	C*	A*	B*	S*	V*
Pharmacophore-based	PharmMapper (Human protein targets only, v2010)	×	×	×	×	×	×	×
	PharmMapper (All targets, v2010)	×	×	×	×	×	×	×
	PharmMapper (Druggable pharmacophore models, 2017)	×	×	×	×	×	×	×
	PharmMapper (Pharmacophore models whose K_d >= 6.0, v2017)	×	×	×	×	×	×	×
2D similarity-based	SEA (Atom-pair / dipole-pair fingerprint)	×	×	×	×	×	×	×
	SEA (Extended connectivity fingerprint)	×	×	×	×	×	×	×
3D similarity-based	ChemMapper (BindingDB)	×	×	×	×	×	×	×
	ChemMapper (ChEMBL)	×	×	×	×	×	×	×
	ChemMapper (DrugBank)	×	×	×	×	×	×	×
	ChemMapper (KEGG)	×	×	×	×	×	×	×
	ChemMapper (PDB)	×	×	×	×	×	√	√
2D and 3D similarity-based	SwissTargetPrediction	√	√	×	×	×	×	√
Machine learning-based	TargetHunter	×	×	×	×	√	×	×
	TargetNet (Daylight fingerprint)	×	×	×	×	×	×	×
	TargetNet (ECFP2 fingerprint)	×	×	×	×	×	×	×
	TargetNet (ECFP4 fingerprint)	×	×	×	×	×	×	×
	TargetNet (ECFP6 fingerprint)	×	×	×	×	×	×	×
Network-based	NBI (Global-NBI)	×	×	×	√	×	×	×
	SDTNBI (Global-SDTNBI- FCFP_4)	√	×	×	×	×	×	×
	bSDTNBI (Global-bSDTNBI-FCFP_4)	√	×	×	√	√	×	√
	wSDTNBI (Global-wSDTNBI-FCFP_4)	√	√	√	√	√	√	√

U*: Ursonic acid, O*: Oleanonic acid, C*: Ciclesonide, A*: AKT inhibitor VIII, B*: BX-471, S*: Spironolactone, V*: Veratramine.