# Response to the Reviewers

Firstly, we would like to sincerely thank the editor and the five reviewers for their professional work and precious time on our manuscript entitled "*DTI-CDF: a cascade deep forest model towards the prediction of drug-target interactions based on hybrid features*" (*BIB-19-0223*) and for the constructive comments and supportive recommendations. According to their valuable suggestions, the revised manuscript has been substantially improved. In addition, the grammatical errors and equations in manuscript have been corrected with great caution. This revision incorporates our significant efforts to carefully address each of those concerns made by the reviewers. The corresponding changes made in the revision (shown in red color) are presented in the revised manuscript.

Please check our detailed point-by-point response to the specific comments raised by the reviewers below.

Since the response to the reviewers should be uploaded as text which cannot show the tables and figures, the same copy of response letter (BIB-19-0223-Response to the Reviewers.pdf), which include the tables and figures, to the reviewers can be accessed at https://github.com//a96123155/DTI-CDF.

# Response to Reviewer #1

We appreciate the positive feedback and sincerely thank the reviewer for the thoughtful and supportive recommendations. Our responses to the comments from reviewer are given below.

**Comments to the Author**

In the manuscript under consideration, the authors proposed a novel drug-target interactions prediction method DTI-CDF based on a cascade deep forest model with hybrid features. The proposed model was tested under experimental settings, and the results show that DTI-CDF achieves significantly

higher performance compared with the state-of-the-art methods. It is a nice contribution to the field. Overall, the text is clear and well-written. However, there are some minor issues that should be revised.

*(1) Too much for the supervised learning part (Pages 5-6) in Introduction Section. I suggest the authors to separate similarity-based method from feature vector-based methods, which the logic will be clearer.*

**Reply:** We sincerely thank the reviewer for the useful suggestions, which are of great help to improve the quality of this manuscript. We have reviewed plenty of literatures about DTI research, especially based on machine learning methods such as to exhibit a comprehensive background. The original classification has been replaced with a logical clearer one. More specifically, we found that numerous methods cannot be classified into supervised and unsupervised categories straightforwardly. Therefore, we have re-classified the chemogenomic methods: we first introduce the network or graph-based methods and machine learning-based methods. There are large quantities of investigations are network/graph-based methods. Then, we further investigate machine learning-based methods, especially for semi-supervised methods since they do not need large number of labeled data. However, supervised learning methods are less studied although a number of deep learning methods are proposed but the performance is not highly satisfied. After a more careful review, it turns out that supervised learning method requires more investigation and probably will demonstrate a good performance compared to conventional methods. We believe the revised description will indeed meet with the requirement and more logical clear than the original one. The revised details can be seen from paragraph 2 in page 5 and paragraph 1 in page 6 (red color words displayed in revised manuscript).

*(2) Table 3, why not compare the proposed method with XGBoost?*

**Reply:** Thank you for your effective comments. Considering the existence of the Extreme Gradient Boosting (XGBoost, XGB) in the cascade deep forest (CDF) model, we have done an experiment comparing CDF with XGB.

XGB [1] is a machine learning technique based on the gradient boosting decision tree in the boosting class to solve the regression and classification problem, it is one of the most popular methods

for the Kaggle machine learning competition and has been widely adopted in various data mining fields, such as the field of bioinformatics [2-3]. Compared to the traditional gradient tree boosting algorithms, XGB has the following advancement: XGB adds a regularization term to the objective function. When minimizing the objective function, XGB not only uses the first order but also the second order gradient statistics. In addition, shrinkage [4] and feature subsampling [5,6] are introduced to further prevent overfitting.

Since XGB has the above advantages, we use it as a base learner for the CDF model. In order to verify that the performance of the CDF model is better than XGB, this study compares XGB and CDF under four data sets and three experimental conditions. The experimental results are shown in Table 1. All data is calculated using 15 decimal places, but for more intuitive, only two decimal places are displayed. The results show that the performance of XGB under 12 experimental conditions is lower than the CDF model proposed in this paper. Analysis of the reasons may be due to the following two points: (1) The model diversity of CDF is more abundant than that of XGB; (2) The multi-layer characteristics of CDF are more fully exploited for features. The comparison with conventional ensemble learning model (such as XGB) has been added in revised manuscript. More detailed can be seen from Section 3.2 in page 16 and 17. (red color words displayed in revised manuscirpt)

**Table 1.** Comparison among XGBoost (XGB) and cascade deep forest (CDF) model. For the sake of clarity, all data is represented by two decimal places, but the calculation uses 15 decimal places.

| Data set | Experimental setting | Model | AUPR | AUC | $F_2$-score | Average of AUPR, AUC and $F_2$-score |
|---|---|---|---|---|---|---|
| Nuclear receptors | $S_p$ | XGB | 0.90 | 0.95 | 0.84 | 0.90 |
| | | CDF | 0.93 | 0.98 | 0.84 | 0.92 |
| | $S_D$ | XGB | 0.76 | 0.87 | 0.69 | 0.77 |
| | | CDF | 0.79 | 0.92 | 0.77 | 0.82 |
| | $S_T$ | XGB | 0.74 | 0.88 | 0.72 | 0.78 |
| | | CDF | 0.76 | 0.91 | 0.72 | 0.80 |
| G-protein- | $S_p$ | XGB | 0.88 | 0.94 | 0.82 | 0.88 |

| | | | | | | |
|---|---|---|---|---|---|---|
| coupled receptors | | CDF | 0.90 | 0.98 | 0.84 | 0.90 |
| | $S_D$ | XGB | 0.75 | 0.86 | 0.67 | 0.76 |
| | | CDF | 0.80 | 0.95 | 0.75 | 0.83 |
| | $S_T$ | XGB | 0.77 | 0.90 | 0.70 | 0.79 |
| | | CDF | 0.79 | 0.96 | 0.76 | 0.83 |
| Ion channels | $S_p$ | XGB | 0.96 | 0.98 | 0.92 | 0.95 |
| | | CDF | 0.96 | 0.99 | 0.93 | 0.96 |
| | $S_D$ | XGB | 0.82 | 0.89 | 0.75 | 0.82 |
| | | CDF | 0.84 | 0.93 | 0.78 | 0.85 |
| | $S_T$ | XGB | 0.90 | 0.93 | 0.83 | 0.89 |
| | | CDF | 0.91 | 0.97 | 0.85 | 0.91 |
| Enzymes | $S_p$ | XGB | 0.95 | 0.97 | 0.92 | 0.95 |
| | | CDF | 0.95 | 0.98 | 0.93 | 0.96 |
| | $S_D$ | XGB | 0.87 | 0.92 | 0.78 | 0.86 |
| | | CDF | 0.87 | 0.95 | 0.83 | 0.88 |
| | $S_T$ | XGB | 0.91 | 0.94 | 0.87 | 0.91 |
| | | CDF | 0.91 | 0.96 | 0.88 | 0.92 |

**Reference:**

[1] Chen T, Guestrin C (2016), 'Xgboost: A scalable tree boosting system', *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, ACM, pp. 785-794.

[2] Zhong J, Sun Y, Peng W et al. XGBFEMF: an XGBoost-based framework for essential protein prediction, IEEE TRANSACTIONS ON NANOBIOSCIENCE 2018;17:243-250.

[3] Babajide Mustapha I, Saeed F. Bioactive molecule prediction using extreme gradient boosting, MOLECULES 2016;21:983.

[4] Friedman JH. Stochastic gradient boosting, COMPUTATIONAL STATISTICS & DATA ANALYSIS 2002;38:367-378.

[5] Breiman L. Random forests, MACHINE LEARNING 2001;45:5-32.

[6] Friedman JH, Popescu BE. Importance sampled learning ensembles, JOURNAL OF MACHINE LEARNING RESEARCH 2003;94305:1-32.

*(3) Why random forest and XGBoost were used as learners in CDF model? Is it possible for other tree-based models?*

**Reply:** We appreciate the reviewer's pointing out this issue. It is a pleasure to answer such a vital question. Kindly note that in the cascade deep forest (CDF) model, any tree-based model can be implemented as a base classifier. The reasons we chose random forest (RF) and XGBoost (XGB) are as follows.

One of the main types of classification methods is ensemble learning which completes learning tasks by constructing and combining multiple learner, it is usually possible to obtain generalization performance superior to that of a single learner, and to solve the imbalance classification problem to some extent. According to the generation method of individual learners, the current ensemble learning methods can be divided into two categories:

Boosting [1]: It is a family of algorithms that can promote weak learners to strong learners. The working mechanism of this family of algorithms is as follows: first, a base learner is trained from the initial training set, and then the training sample distribution is adjusted according to the performance of the base learner, so that the misclassified training samples of the previous base learner receive more attention in the follow-up. Then, the next base learner is trained based on the adjusted sample distribution. Repeat the above steps until the number of base learners reaches a predetermined value. Finally, all the base learners are weighted and combined. From the perspective of deviation-variance decomposition, boosting focuses on reducing bias. XGB [2] is a representative algorithm of the boosting family. It is a scalable tree boosting method that adds regular terms to the cost function, which can control the complexity of the model and prevent over-fitting. At the same time, it uses a second-order Taylor expansion approximation to the cost function, which makes the approximation of the objective function closer to the actual value, thus improving the prediction accuracy. In addition, XGB makes use of some techniques to speed up calculations.

Bagging [3]: Based on bootstrap sampling, the initial data set is subjected to multiple sampling with replacement, and each base learner is trained based on each sample set, then all base learners are combined. From the perspective of deviation-variance decomposition, bagging mainly focuses on reducing the variance. RF [4] is a representative variant of bagging. Based on the bagging, RF introduces the random feature selection in the training process of the decision tree. That is, the diversity of the RF comes not only from the sample disturbance but also from the feature disturbance, and this improves the generalization performance. Moreover, RF are easy to implement and exhibit powerful performance in many real-world tasks, especially in bioinformatics.

Many studies have shown that the diversity of base-learners improves the classification performance of ensemble learning [5], but not all the base-learners with diversity can achieve good classification results [6]. Moreover, the more base-learners mean the greater computational overhead. According to the idea of selective ensemble [6], only choose the part of the diverse base-learners that can complement each other would achieve better classification results. Therefore, individual learners should be "good and different" to get a good ensemble.

In order to construct a model with low variance and low deviation, find a balance point in over-fitting and under-fitting, this study combines boosting and bagging [7], we choose XGB and RF respectively as their representativeness. In order to reduce the complexity of the model, only these two models are considered as base-learners in this study. In the future, all possible individual learners can be enumerated, and individual learners with the best classification effect can be selected for ensemble. The analysis of based learner selection has been demonstrated clearly paragraphs 1, 2 and 3 in page 16 in Section 3.2. (red color words displayed in revised manuscirpt)

**Reference:**

[1] Schapire RE (1999), 'A brief introduction to boosting', *Ijcai*, pp. 1401-1406.

[2] Chen T, Guestrin C (2016), 'Xgboost: A scalable tree boosting system', *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, ACM, pp. 785-794.

[3] Breiman L. Bagging predictors, MACHINE LEARNING 1996;24:123-140.

[4] Breiman L. Random forests, MACHINE LEARNING 2001;45:5-32.

[5] Kuncheva LI, Whitaker CJ. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy, MACHINE LEARNING 2003;51:181-207.

[6] Zhou Z, Wu J, Tang W. Ensembling neural networks: many could be better than all, ARTIFICIAL INTELLIGENCE 2002;137:239-263.

[7] Galar M, Fernandez A, Barrenechea E et al. A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 2011;42:463-484.

*(4) Figure 3, why there is a high correlation between PathCS and PsePSSM on the dataset NR? I am wondering whether it is possible to give a basic explanation about this point.*

**Reply:** Thank you for mentioning the important point. Yes, from figure 3, it seems that the correlation between PathCS and PsePSSM on the dataset NR is high. However, the scale in the figure is $10^{-4}$. Although the correlation between PathCS and PsePSSM is higher than others, it is also very low and close to 0. Therefore, it does not mean that there is a high correlation between the PathCS and PsePSSM on the dataset NR, data in Table 2 is represented by scientific notation.

However, after careful and rigorous analysis, we decided to remove the two types of features, FP2 and PsePSSM, due to the following reasons:

In the data set construction process, for each data set, the sample space of this study is the Cartesian product space of drugs and targets, that is, the number of samples in each data set is the product of the number of drugs and the number of targets, each drug-target pair as a sample. Among them, the drug-target pair of known interaction is used as the positive sample, and the rest are processed as negative samples, which leads to the presence of noise in the negative samples (the real existence of drug-target interactions not yet discovered). Therefore, when the performance evaluation metrics of the model is very high in this study, it cannot be considered that the model performance is very good, because the model may not correctly judge and predict the noise sample, and the original intention of this study is to effectively predict the undiscovered drug-target interactions, not just to obtain a high performance model.

According to Table 2, the correlation between the three types of features is very low, indicating

that there is no information redundancy between the three types of features, that is, the addition of FP2 and PsePSSM enriches the feature information, which is also the reason for the improved performance of the model compared to that of using PathCS alone. Furthermore, to determine whether the model can effectively predict unknown but real drug-target interactions, we use drug-target interactions that have been reported but not yet included in the data set to judge. For each negative drug-target pair which is predicted as positive, we search it in KEGG [1] and DrugBank [2] databases to determine whether it is predicted successfully. In the prediction results of the model using only PathCS, more than 1000 drug-target pairs are determined as successful predictions as they can be found in the above two databases, and these new DTIs will be detailed in the website (https://github.com/a96123155/DTI-CDF/tree/master/3_new_DTIs). But models that use composite features (i.e. the combination of PathCS, FP2 and PsePSSM) do not achieve such encouraging results. Therefore, in order to strike a balance between high performance and effective prediction of undiscovered drug-target interactions, we decided to use only PathCS as the input feature of the model. Although the performance of the model is lost to some extent, the above-mentioned undiscovered drug-target interactions is effectively predicted which is more important, and the computational speed of the method is improved. Based on the above demonstration, we have decided not to utilize the feature of FP2 and PsePssM to enhance the practical prediction ability of our model. We have deleted the corresponding description on these two features from the original manuscript.

**Table 2.** Comparison of cross-correlation coefficients among three types of features. NR is short for nuclear receptors, GPCR for G-protein-coupled receptors, IC for ion channels, and E for enzymes.

| | PathCS-FP2 | | | PathCS-PsePSSM | | | FP2-PsePSSM | | |
|---|---|---|---|---|---|---|---|---|---|
| | $S_p$ | $S_D$ | $S_T$ | $S_p$ | $S_D$ | $S_T$ | $S_p$ | $S_D$ | $S_T$ |
| NR | 9.5E-6 | 1.2E-5 | 1.2E-5 | 8.9E-5 | 1.2E-4 | 1.2E-4 | 1.2E-8 | 1.2E-8 | 1.2E-8 |
| GPCR | 3.9E-7 | 4.6E-7 | 4.7E-7 | 6.5E-7 | 8.1E-7 | 8.1E-7 | 1.6E-10 | 1.6E-10 | 1.6E-10 |
| IC | 1.5E-7 | 1.7E-7 | 1.7E-7 | 2.0E-7 | 2.4E-7 | 2.4E-7 | 5.6E-11 | 5.6E-11 | 5.6E-11 |
| E | 1.1E-8 | 1.3E-8 | 1.3E-8 | 1.1E-8 | 1.6E-8 | 1.6E-8 | 3.5E-12 | 3.5E-12 | 3.5E-12 |

**Reference:**

[1] Kanehisa M, Furumichi M, Tanabe M et al. KEGG: new perspectives on genomes, pathways, diseases and drugs, NUCLEIC ACIDS RESEARCH 2016;45:D353-D361.

[2] Wishart DS, Knox C, Guo AC et al. DrugBank: a knowledgebase for drugs, drug actions and drug targets, NUCLEIC ACIDS RESEARCH 2007;36:D901-D906.

*(5) It will be better to present more future directions in the Conclusions Section.*

**Reply:** Thank you for your valuable suggestion focusing towards the betterment. There is involvement of many computational techniques involved in drug repositioning used in various conditions that depends on the existing knowledge level we have, on the target disease. Complete investigations providing outlines to these computational methods, we generate an outline of DTI prediction, which is an important aspect of the drug discovery process. In the future, we will continue to study in both data sets and algorithms. First, we will update the data set used in this study (this study used the 2008 version) and add drug-target interaction data from other target types (such as non-coding RNA [1-3]). On the other hand, we plan to design algorithms that are more suitable for highly unbalanced data sets (such as the introduction of data preprocessing techniques, etc.) and apply this method to the regression problem such as the drug-target affinity prediction. We have added the description of further directions in Conclusion Section. More detailed can be seen from paragraph 2 of Section 5 in page 20. (red color words displayed in revised manuscript)

**Reference:**

[1] Chen X, Guan N, Sun Y et al. MicroRNA-small molecule association identification: from experimental results to computational models, Brief. Bioinform 2018;16:1-15.

[2] Qu J, Chen X, Sun Y et al. Inferring potential small molecule–miRNA association based on triple layer heterogeneous network, Journal of Cheminformatics 2018;10:30.

[3] Yin J, Chen X, Wang C et al. Prediction of small molecule-microRNA associations by sparse learning and heterogeneous graph inference, MOLECULAR PHARMACEUTICS 2019.

# Response to Reviewer #2

We appreciate the positive feedback and sincerely thank the reviewer for the thoughtful and supportive recommendations. Our responses to the comments from reviewer are given below.

**Comments to the Author**

In this manuscript, Chu et al. developed a CDF model for predicting potential DTIs. Their experimental results show that the new model outperforms some previously published methods. In general, the presented methods and results are likely to be useful to other researchers. However, the review part of the computational methods in Introduction section need to be reorganized. Moreover, several questions on the methodology are critical to the conclusions drawn in this work, which need to be addressed. I am listing them as follows:

*1) In Page 4, the authors introduced two different ways to categorize the recent computation methods for prediction of DTIs in a paragraph, which make the review of types of methods very confusing. The author need rewrite this paragraph to clearly summarize the different types of computational methods.*

**Reply:** Introduction has been rewritten as per the mentioned comment. The errors have been corrected to remove the bewilderment and we hope that it will now satisfy the reviewer. We have rewritten the paragraph introducing computational methods. We believe the revised description is more logical clear. The detailed revision can be seen from paragraphs 2 and 3 in Introduction part. (red color words displayed in revised manuscript)

*2) When the authors introduced the machine learning-based methods on page 5, it is better for them to summarize the differences of the published methods in a table, which is more organized so that the readers can quickly understand them.*

**Reply:** Thank you for your effective comment. We understand that summarizing the differences in a table makes it clearer and easier to understand as well as more readable. Therefore, in order to improve our work. Table 1 has been used to summarize the supervised machine learning-based methods. We

have re-organized the description about machine learning-based methods in revised manuscript. We do not intend to put the large Table 1 into the manuscript directly for saving space. More details can be seen from paragraph 3 in Introduction part. (red color word displayed in revised manuscript)

**Table 1.** The supervised learning methods. S is short for similarity-based methods, F for feature-based methods. The matrix factorization methods and deep learning methods are not listed here.

| Ref. | Type | Similarities/Features | Classifiers | Additional description |
|---|---|---|---|---|
| [1] | S, F | Chemical structure and target sequence-based similarity | SVM | A bipartite local models (BLM) using a SVM-based classifier. |
| [2] | S,F | Chemical structure and target sequence-based similarity | regularized least square classifier | A kernel-based method using the Gaussian interaction profile kernel based on BLM. |
| [3] | S,F | Chemical, pharmacological, drug-drug interactions, target sequence and enzyme commission-based similarity | SVM, logistic regression | A kernel-based method using SVM and logistic regression based on BLM |
| [4] | S | Chemical structure and target sequence-based similarity | - | Further exploited BLM with Neighbor-based Interaction Profile Inferring, which adds a preprocessing component to infer training data from neighbors' interaction profiles. |
| [5] | F | Meta-path-based topological features | Random forest | Based on meta-path-based topological features, random forest classifier is employed to do prediction. |
| [6] | S, F | Path-category-based multi-similarities features | Random forest | Executes graph mining technique to acquire the comprehensive feature |

| | | | | |
|---|---|---|---|---|
| | | | | vectors and then applies the random forest model by using different graph-based features extracted from the drug-target heterogeneous graph. |
| [7] | F | Evolutionary and structural features | Adaboost | Explored sampling method to avoid data imbalance problem and used Adaboost model to do prediction. |
| [8] | F | The features of the drugs and targets have been calculated using the Rcpi package and the PROFEAT web server. | Ensemble learning | Carried out the ensemble learning method which uses decision tree and kernel ridge regression as base classifiers. |

**Reference:**

[1] Bleakley K, Yamanishi Y. Supervised prediction of drug–target interactions using bipartite local models, BIOINFORMATICS 2009;25:2397-2403.

[2] van Laarhoven T, Nabuurs SB, Marchiori E. Gaussian interaction profile kernels for predicting drug–target interaction, BIOINFORMATICS 2011;27:3036-3043.

[3] Kim S, Jin D, Lee H. Predicting drug-target interactions using drug-drug interactions, PLoS One 2013;8:e80129.

[4] Mei J, Kwoh C, Yang P et al. Drug–target interaction prediction by learning from local information and neighbors, BIOINFORMATICS 2012;29:238-245.

[5] Fu G, Ding Y, Seal A et al. Predicting drug target interactions using meta-path-based semantic network analysis, BMC BIOINFORMATICS 2016;17:160.

[6] Olayan RS, Ashoor H, Bajic VB. DDR: efficient computational method to predict drug–target interactions using graph mining and machine learning approaches, BIOINFORMATICS 2017;34:1164-1173.

[7] Rayhan F, Ahmed S, Shatabda S et al. iDTI-ESBoost: identification of drug target interaction using evolutionary and structural features with boosting, Scientific Reports 2017;7:17731.

[8] Ezzat A, Wu M, Li X et al. Drug-target interaction prediction using ensemble learning and dimensionality reduction, METHODS 2017;129:81-88.

*3) In section 2.2.1, why the subsequence length was set to 4 in protein kernels?*

**Reply:** We are pleased to answer such a vital question. As in the previous study [1], the $k = 3$ and $k = 4$ were tested, the differences between these two kernels are small in remote homology detection. In addition, $k = 4$ was selected from multiple similarity measures as it has better performance in the drug-target interaction prediction [2]. We hope that we are able to justify the reason of setting the length to 4 in protein kernels.

**Reference:**

[1] Leslie C, Eskin E, Noble WS. The spectrum kernel: A string kernel for SVM protein classification. Biocomputing 2002.: World Scientific, 2001, 564-575.

[2] Olayan RS, Ashoor H, Bajic VB. DDR: efficient computational method to predict drug–target interactions using graph mining and machine learning approaches, BIOINFORMATICS 2017;34:1164-1173.

*4) In section 2.2.2, how the PSSM was generated? Which database a protein is aligned with?*

**Reply:** The position-specific scoring matrix (PSSM) [1] is an amino acid substitution scoring matrix in protein multiple sequence alignment. In PSSM, each element is the score that amino acid mutation for each position in a protein sequence, which can be searched using PSI-BLAST [2] in the Swiss-Prot database. In generation process, the parameters of PSI-BLAST are set as: the threshold of E-value is 0.001, the maximum number of iterations for multiple searches is 3, and the rest of the parameters are set by default. The PSSM used in this study was referenced to [3].

However, after careful and rigorous analysis, we decided to remove the two types of features, FP2 and PsePSSM, for the following reasons:

In the data set construction process, for each data set, the sample space of this study is the Cartesian product space of drugs and targets, that is, the number of samples in each data set is the product of the number of drugs and the number of targets, each drug-target pair as a sample. Among them, the drug-

target pair of known interaction is used as the positive sample, and the rest are processed as negative samples, which leads to the presence of noise in the negative samples (the real existence of drug-target interactions not yet discovered). Therefore, when the performance evaluation metrics of the model is extremely high in this study, it cannot be considered that the model performance is good in fact because the model may not correctly judge and predict the noise samples. The original intention of this study is to effectively predict the undiscovered drug-target interactions, not only aims to obtain a high-accuracy model.

To determine whether the model can effectively predict unknown but real drug-target interactions, we use drug-target interactions that have been reported but not yet included in the data set to judge. For each negative drug-target pair which is predicted as positive, we search it in KEGG [4] and DrugBank [5] databases to determine whether it is predicted successfully. In the prediction results of the model using only PathCS, more than 1000 drug-target pairs are determined as successful predictions as they can be found in the above two databases, and these new DTIs will be detailed in the article. But models that use composite features (i.e. the combination of PathCS, FP2 and PsePSSM) do not achieve such encouraging results. Therefore, in order to strike a balance between high performance and effective prediction of undiscovered drug-target interactions, we decided to use only PathCS as the input feature of the model. Although the performance of the model is lost to some extent, the above-mentioned undiscovered drug-target interactions is effectively predicted which is more important, and the computational speed of the method is improved. Based on the above demonstration, we have decided not to utilize the feature of FP2 and PsePssM to enhance the practical prediction ability of our model. We have deleted the corresponding description on these two features from the original manuscript.

**Reference:**

[1] Jones DT. Protein secondary structure prediction based on position-specific scoring matrices, JOURNAL OF MOLECULAR BIOLOGY 1999;292:195-202.

[2] Altschul SF, Madden TL, Schäffer AA et al. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs, NUCLEIC ACIDS RESEARCH 1997;25:3389-3402.

[3] Shi H, Liu S, Chen J et al. Predicting drug-target interactions using Lasso with random forest based on evolutionary information and chemical structure, GENOMICS 2018.

[4] Kanehisa M, Furumichi M, Tanabe M et al. KEGG: new perspectives on genomes, pathways, diseases and drugs, NUCLEIC ACIDS RESEARCH 2016;45:D353-D361.

[5] Wishart DS, Knox C, Guo AC et al. DrugBank: a knowledgebase for drugs, drug actions and drug targets, NUCLEIC ACIDS RESEARCH 2007;36:D901-D906.

*5) There are a lot of types of molecular fingerprints. Why do them choose FP2, rather than other types of fingerprints?*

**Reply:** The basic idea to represent a molecule is using a bit vector that codifies the existence or absence of structure features, functional groups, pharmacophore features or molecular properties [1], called molecular fingerprints. The 2D molecular fingerprints can be divided into three types [1]: (a) topological path-based fingerprint. The feature extraction process consists of two steps: analyzing firstly all molecular fragments from the beginning of an atom until the path of a specified number of bonds, and secondly hashing each of the fragments to generate fingerprints; (b) topological circular fingerprint, also a type of hashing fingerprint. It records the environment of all molecular fragments from the beginning atom until the path of a specified radius of bonds, instead of searching the path among molecules; (c) substructure key-based fingerprint, a type of substructure-based fingerprint. The bit string is set according to the presence or absence of certain substructures or features in a given structure list. FP2 [2], ECFP4 [3], MACCS [4] are the representative fingerprints of the above three types of molecular fingerprints.

To select the fingerprint which suits this task, we use FP2, ECFP4 and MACCS besides PathCS and PsePSSM as input feature vector of cascade deep forest (CDF) model on $S_p$, $S_D$, and $S_T$ experimental settings under nuclear receptors dataset. The results show in Table 2. Apparently, the FP2 achieves the best performance with an acceptable dimension compared with fingerprints of ECFP4 and MACCS under different experimental settings with respect to AUPR and AUC metrics. Thus, we choose FP2 as the representative molecular fingerprint to train our model.

However, we finally decided not to adopt the FP2 and PsePSSM into feature vector. The specific

reasons can be seen in the reply to Comment 4. Based on the above demonstration, we have decided not to utilize the feature of FP2 and PsePssM to enhance the practical prediction ability of our model. We have deleted the corresponding description on these two features from the original manuscript.

**Table 2.** The comparison of FP2, ECPF4, and MACCS as molecule fingerprint.

| Fingerprint | Dimension | AUPR | | | AUC | | |
|---|---|---|---|---|---|---|---|
| | | $S_p$ | $S_D$ | $S_T$ | $S_p$ | $S_D$ | $S_T$ |
| FP2 | 256 | 0.96 | 0.94 | 0.94 | 0.99 | 0.98 | 0.98 |
| ECFP4 | 761 | 0.91 | 0.79 | 0.89 | 0.98 | 0.96 | 0.96 |
| MACCS | 166 | 0.89 | 0.78 | 0.87 | 0.97 | 0.95 | 0.96 |

**Reference:**

[1] Riniker S, Landrum GA. Open-source platform to benchmark fingerprints for ligand-based virtual screening, Journal of Cheminformatics 2013;5:26.

[2] O'Boyle NM, Banck M, James CA et al. Open Babel: An open chemical toolbox, Journal of Cheminformatics 2011;3:33.

[3] Morgan HL. The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service., Journal of Chemical Documentation 1965;5:107-113.

[4] Durant JL, Leland BA, Henry DR et al. Reoptimization of MDL keys for use in drug discovery, Journal of chemical information and computer sciences 2002;42:1273-1280.

*6) The architecture of CDF model is much similar to that of the deep learning-based methods. Why not compare the CDF model with deep learning-based methods on this task?*

**Reply:** Thanks for your imperative and effective comments, we have further compared the traditional deep neural network (DNN) model to the cascade deep forest (CDF) model.

Recently, DNNs or deep learning has achieved great success in many areas including bioinformatics [1]. However, it still has apparent deficiencies. It is well known that the training of DNNs usually requires a large amount of data, so it is difficult to directly apply it to tasks with small-

scale data. It is worth noting that although we are in the era of big data, many practical tasks still lack of a sufficient amount of labeled data due to high labeling cost, resulting in poor performance of DNNs in these tasks. Secondly, DNNs are relatively complex models, and the training processes generally require powerful computing devices. More importantly, DNNs require numerous hyper-parameters to optimize, and the learning performance is heavily dependent on their careful tuning such that causes the training process complex. In addition, the theoretical analysis of DNNs is extremely difficult because too many interference factors are combined with almost unlimited parameter configurations. It is worth noting that a large amount of training data is used, and the learning ability of the model should be large, which explains why DNNs are more complicated than ordinary learning models (such as SVM). If we are able to learn attributes such as other models, it is possible to achieve performance that does not have the above drawbacks and can compete with DNNs.

Our proposed CDF model [2] is a deep ensemble framework that cascades traditional machine learning models. Most machine learning models can be used as its base-learners (such as random forest, XGBoost, logistic regression, etc.) to achieve the deepening of traditional machine learning methods. Compared to DNNs, CDF models have few hyper-parameters, are easier to train, and are highly competitive in performance. In addition, unlike most type of DNNs with fixed model complexity, the CDF model can stop the increase of the number of layers by terminating the training properly, and the complexity of the model can be adaptively scaled, making the CDF model is not limited to large-scale training data like deep learning, it also performs well on small-scale training data. Moreover, if a tree-based approach is chosen as the base-learner, CDF will be easier to theoretically analyze than DNNs. To clarify this point, we compare the CDF model with the traditional deep learning model DNN which structure shown in Figure 1.

The comparison results are shown in Table 3. It shows that in our experiments, CDF achieved a highly competitive performance of DNN since all results in different experimental conditions better than DNN. The reason may be that the sample sizes of the four data sets used in this study are 1404 (NR), 21185 (GPCR), 42840 (IC), and 295480 (E), the data set scale ranges from $10^3$ to $10^5$. In addition, the number of positive and negative samples in each data set is highly unbalanced, too few positive samples make DNN which based on a large amount of training data unable to exert its advantages.

Moreover, the feature dimension used in this study is low, and deep learning has great advantages in the representation learning of ultra-high dimensional data. We have added a subsection to compare the performance of our model with DNN model. More detailed can be found in Section 3.1 page 14 and 15. (red color words displayed in revised manuscript)

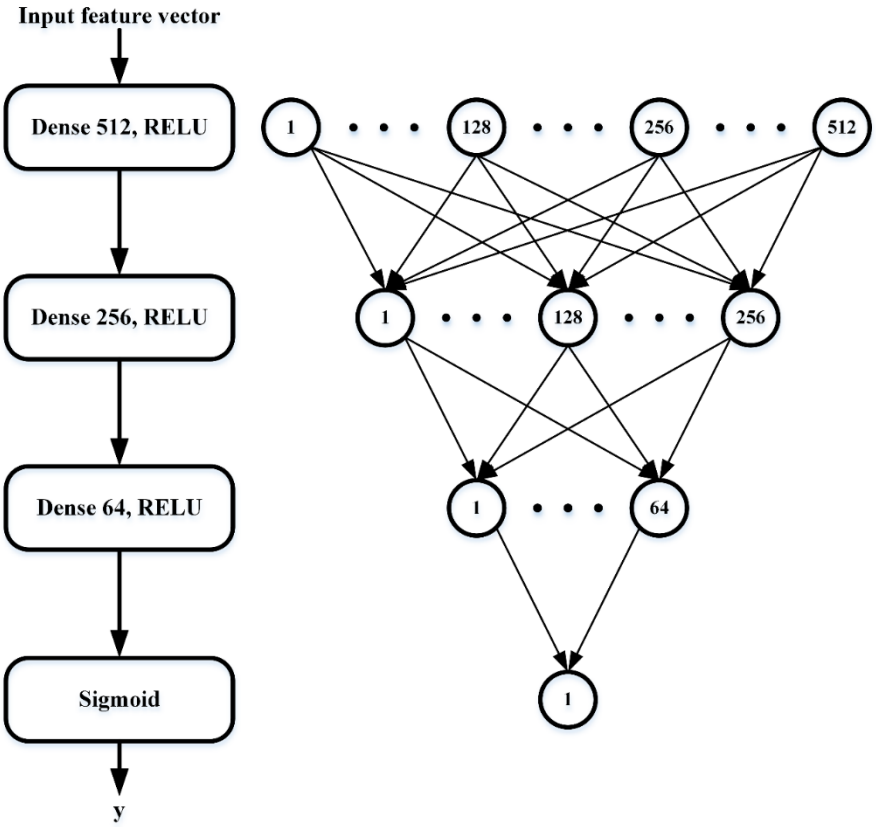



**Figure 1.** The structure of deep neural network (DNN) used in this study. The Dense represents fully connected layer, RELU and Sigmoid are activate functions.

**Table 3.** The comparison of cascade deep forest (CDF) model and deep neural network (DNN) model. For the sake of clarity, all data is represented by two decimal places, but the calculation uses 15 decimal places.

| Data set | Experimental setting | Model | AUPR | AUC | $F_2$-score | Average of AUPR, AUC and $F_2$-score |
|---|---|---|---|---|---|---|
| Nuclear receptors | $S_p$ | CDF | 0.93 | 0.98 | 0.84 | 0.92 |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | DNN | 0.85 | 0.97 | 0.74 | 0.85 |
| | $S_D$ | CDF | 0.79 | 0.92 | 0.77 | 0.82 |
| | | DNN | 0.77 | 0.92 | 0.68 | 0.79 |
| | $S_T$ | CDF | 0.76 | 0.91 | 0.72 | 0.80 |
| | | DNN | 0.68 | 0.90 | 0.43 | 0.67 |
| G-protein-coupled receptors | $S_p$ | CDF | 0.90 | 0.98 | 0.84 | 0.90 |
| | | DNN | 0.80 | 0.95 | 0.67 | 0.81 |
| | $S_D$ | CDF | 0.80 | 0.95 | 0.75 | 0.83 |
| | | DNN | 0.72 | 0.92 | 0.59 | 0.74 |
| | $S_T$ | CDF | 0.79 | 0.96 | 0.76 | 0.83 |
| | | DNN | 0.65 | 0.94 | 0.53 | 0.71 |
| Ion channels | $S_p$ | CDF | 0.96 | 0.99 | 0.93 | 0.96 |
| | | DNN | 0.90 | 0.99 | 0.79 | 0.89 |
| | $S_D$ | CDF | 0.84 | 0.93 | 0.78 | 0.85 |
| | | DNN | 0.79 | 0.91 | 0.70 | 0.80 |
| | $S_T$ | CDF | 0.91 | 0.97 | 0.85 | 0.91 |
| | | DNN | 0.85 | 0.96 | 0.72 | 0.84 |
| Enzymes | $S_p$ | CDF | 0.95 | 0.98 | 0.93 | 0.96 |
| | | DNN | 0.71 | 0.77 | 0.45 | 0.64 |
| | $S_D$ | CDF | 0.87 | 0.95 | 0.83 | 0.88 |
| | | DNN | 0.64 | 0.75 | 0.38 | 0.59 |
| | $S_T$ | CDF | 0.91 | 0.96 | 0.88 | 0.92 |
| | | DNN | 0.66 | 0.78 | 0.47 | 0.64 |

**Reference:**

[1] Min S, Lee B, Yoon S. Deep learning in bioinformatics, BRIEFINGS IN BIOINFORMATICS 2017;18:851-869.

[2] Zhou Z, Feng J. Deep Forest, arXiv preprint arXiv:1702.08835 2017.

*7) In the components of the CDF model, how do they get the optimal combinations of base-classifiers (learners) in a layer?*

**Reply:** In this study, CDF model only use random forest and XGBoost as base-classifiers, and the results, as shown in Table 4, indicate that different combinations achieve similar performance when they share the same set of parameters because the CDF model is not very sensitive for parameter setting [1]. It demonstrates the robustness of our CDF model against the model selection, which will avoid the large amount of work in the parameter tuning. More significantly, our CDF model preserves the universality over different data sets such that it can be migrated to other DTI applications. It indicates that the CDF model is not very sensitive to parameter setting. Thus, we do not need to do the large-scale parameter tuning including the selection of the optimal combinations of base-classifiers, which is also one of the advantages when compared to DNN. We have added a subsection to analyze the hyper-parameters optimization through plenty of experiments. More detailed can be found in Section 3.3 page 17 and 18. (red color words displayed in revised manuscript)

**Table 4.** The results of different combinations of base-classifiers in cascade deep forest model, where RF1-XGB1 represents one random forest and one XGBoost, and so on. For the sake of clarity, all data is represented by two decimal places, but the calculation uses 15 decimal places.

| Data set | Experimental setting | Combination | AUPR | AUC | $F_2$-score | Average of AUPR, AUC and $F_2$-score |
|---|---|---|---|---|---|---|
| Nuclear receptors | $S_p$ | RF2 | 0.93 | 0.98 | 0.84 | 0.92 |
| | | XGB2 | 0.92 | 0.98 | 0.83 | 0.91 |
| | | RF1-XGB1 | 0.91 | 0.98 | 0.83 | 0.91 |
| | | RF1-XGB2 | 0.93 | 0.98 | 0.83 | 0.91 |
| | | RF2-XGB1 | 0.92 | 0.98 | 0.84 | 0.91 |
| | | RF2-XGB2 | 0.92 | 0.98 | 0.83 | 0.91 |

| | | | | | | |
|---|---|---|---|---|---|---|
| | $S_D$ | RF2 | 0.79 | 0.92 | 0.77 | 0.82 |
| | | XGB2 | 0.78 | 0.92 | 0.69 | 0.80 |
| | | RF1-XGB1 | 0.77 | 0.91 | 0.73 | 0.81 |
| | | RF1-XGB2 | 0.78 | 0.90 | 0.71 | 0.80 |
| | | RF2-XGB1 | 0.79 | 0.91 | 0.75 | 0.82 |
| | | RF2-XGB2 | 0.79 | 0.89 | 0.73 | 0.80 |
| | $S_T$ | RF2 | 0.73 | 0.90 | 0.69 | 0.77 |
| | | XGB2 | 0.76 | 0.91 | 0.72 | 0.80 |
| | | RF1-XGB1 | 0.73 | 0.91 | 0.71 | 0.78 |
| | | RF1-XGB2 | 0.76 | 0.91 | 0.69 | 0.78 |
| | | RF2-XGB1 | 0.75 | 0.91 | 0.69 | 0.78 |
| | | RF2-XGB2 | 0.76 | 0.91 | 0.70 | 0.79 |
| G-protein-coupled receptors | $S_p$ | RF2 | 0.89 | 0.97 | 0.84 | 0.90 |
| | | XGB2 | 0.90 | 0.97 | 0.82 | 0.90 |
| | | RF1-XGB1 | 0.89 | 0.98 | 0.84 | 0.90 |
| | | RF1-XGB2 | 0.90 | 0.97 | 0.81 | 0.89 |
| | | RF2-XGB1 | 0.90 | 0.98 | 0.84 | 0.90 |
| | | RF2-XGB2 | 0.90 | 0.98 | 0.82 | 0.90 |
| | $S_D$ | RF2 | 0.77 | 0.94 | 0.73 | 0.81 |
| | | XGB2 | 0.80 | 0.94 | 0.69 | 0.81 |
| | | RF1-XGB1 | 0.80 | 0.95 | 0.75 | 0.83 |
| | | RF1-XGB2 | 0.79 | 0.93 | 0.69 | 0.80 |
| | | RF2-XGB1 | 0.79 | 0.94 | 0.74 | 0.82 |
| | | RF2-XGB2 | 0.79 | 0.94 | 0.72 | 0.82 |
| | $S_T$ | RF2 | 0.76 | 0.96 | 0.75 | 0.82 |
| | | XGB2 | 0.79 | 0.94 | 0.70 | 0.81 |
| | | RF1-XGB1 | 0.79 | 0.96 | 0.76 | 0.83 |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | RF1-XGB2 | 0.78 | 0.93 | 0.70 | 0.80 |
| | | RF2-XGB1 | 0.78 | 0.95 | 0.74 | 0.83 |
| | | RF2-XGB2 | 0.78 | 0.95 | 0.73 | 0.82 |
| Ion channels | $S_p$ | RF2 | 0.96 | 0.99 | 0.92 | 0.96 |
| | | XGB2 | 0.96 | 0.99 | 0.92 | 0.96 |
| | | RF1-XGB1 | 0.96 | 0.99 | 0.93 | 0.96 |
| | | RF1-XGB2 | 0.96 | 0.99 | 0.91 | 0.96 |
| | | RF2-XGB1 | 0.96 | 0.99 | 0.93 | 0.96 |
| | | RF2-XGB2 | 0.96 | 0.99 | 0.92 | 0.96 |
| | $S_D$ | RF2 | 0.82 | 0.92 | 0.78 | 0.84 |
| | | XGB2 | 0.84 | 0.92 | 0.75 | 0.84 |
| | | RF1-XGB1 | 0.84 | 0.93 | 0.78 | 0.85 |
| | | RF1-XGB2 | 0.83 | 0.91 | 0.75 | 0.83 |
| | | RF2-XGB1 | 0.83 | 0.92 | 0.77 | 0.84 |
| | | RF2-XGB2 | 0.83 | 0.92 | 0.77 | 0.84 |
| | $S_T$ | RF2 | 0.90 | 0.97 | 0.85 | 0.91 |
| | | XGB2 | 0.91 | 0.97 | 0.83 | 0.90 |
| | | RF1-XGB1 | 0.90 | 0.97 | 0.85 | 0.91 |
| | | RF1-XGB2 | 0.91 | 0.96 | 0.84 | 0.90 |
| | | RF2-XGB1 | 0.91 | 0.96 | 0.84 | 0.90 |
| | | RF2-XGB2 | 0.91 | 0.97 | 0.85 | 0.91 |
| Enzymes | $S_p$ | RF2 | 0.95 | 0.98 | 0.92 | 0.95 |
| | | XGB2 | 0.95 | 0.98 | 0.92 | 0.95 |
| | | RF1-XGB1 | 0.95 | 0.98 | 0.93 | 0.95 |
| | | RF1-XGB2 | 0.95 | 0.98 | 0.92 | 0.95 |
| | | RF2-XGB1 | 0.95 | 0.98 | 0.93 | 0.96 |
| | | RF2-XGB2 | 0.95 | 0.98 | 0.92 | 0.95 |

| | | | | | | |
|---|---|---|---|---|---|---|
| | $S_D$ | RF2 | 0.85 | 0.94 | 0.83 | 0.87 |
| | | XGB2 | 0.87 | 0.93 | 0.79 | 0.86 |
| | | RF1-XGB1 | 0.87 | 0.95 | 0.83 | 0.88 |
| | | RF1-XGB2 | 0.86 | 0.92 | 0.79 | 0.85 |
| | | RF2-XGB1 | 0.86 | 0.95 | 0.81 | 0.87 |
| | | RF2-XGB2 | 0.86 | 0.94 | 0.80 | 0.87 |
| | $S_T$ | RF2 | 0.90 | 0.96 | 0.88 | 0.91 |
| | | XGB2 | 0.91 | 0.95 | 0.87 | 0.91 |
| | | RF1-XGB1 | 0.91 | 0.96 | 0.88 | 0.92 |
| | | RF1-XGB2 | 0.91 | 0.94 | 0.87 | 0.91 |
| | | RF2-XGB1 | 0.90 | 0.96 | 0.88 | 0.91 |
| | | RF2-XGB2 | 0.91 | 0.96 | 0.88 | 0.91 |

**Reference:**

[1] Zhou Z, Feng J. Deep Forest, arXiv preprint arXiv:1702.08835 2017.

*8) In Section 3.1, the introduction about the statistical test should be better moved to Method section.*

**Reply:** Thank you for suggesting the significant correction. As per the mentioned comment, the statistical test has been moved to Section 2.6 in page 12 and 13. (red color words displayed in revised manuscript)

*9) In recent years, more and more non-coding RNAs (ncRNAs) have been identified and increasing evidences have shown that ncRNAs may affect gene expression and disease progression, making them a new class of targets for drug discovery. It thus becomes important to understand the relationship between ncRNAs and drug targets. You should discuss the potential possibility of using your novel model for the perdition of drug-related ncRNA targets and important literatures should be cited (PMIDs: 30325405, 29943160, and 31136190)*

**Reply:** The aspects of noncoding RNAs have been added to the revised manuscript as per the comment focusing on improving of our work.

This work particularly focuses on the target proteins, but there is another type of target (ncRNAs [1-3]) and the drugs for them are successfully developed. These are the RNAs that does not perform coding for proteins, and they contain subcategories including microRNAs (miRNAs), long coding RNAs and Intronic RNAs among several others. Few examples are use of miRNAs to treat the Hepatitis C virus and Alport nephropathy, while Duchenne muscular dystrophy and Usher syndrome has been treated by intronic RNAs. The behaviour and mechanism of each of ncRNAs is unique thus leading to different opportunities and challenges, all of which are discussed with examples in a latest article.

There is an expectation, that in future there may be more research in this field. NRDTD database is worthy of considering as it is used to store information on ncRNAs and their associated drugs. It is expected to research into ncRNAs as drug targets will observe the regular use of this database [4-8]. We have added the following references in paragraph 2 of Section5 in page 20. (red color words displayed in revised manuscript)

**References:**

[1] Chen X, Guan N, Sun Y et al. MicroRNA-small molecule association identification: from experimental results to computational models, Brief. Bioinform 2018;16:1-15.

[2] Qu J, Chen X, Sun Y et al. Inferring potential small molecule–miRNA association based on triple layer heterogeneous network, Journal of Cheminformatics 2018;10:30.

[3] Yin J, Chen X, Wang CC et al. Prediction of Small Molecule-MicroRNA Associations by Sparse Learning and Heterogeneous Graph Inference, Mol Pharm 2019;16:3157-3166.

[4] Thakral S, Ghoshal K. miR-122 is a unique molecule with great potential in diagnosis, prognosis of liver disease, and therapy both as miRNA mimic and antimir, CURRENT GENE THERAPY 2015;15:142.

[5] Gomez IG, MacKenna DA, Johnson BG et al. Anti-microRNA-21 oligonucleotides prevent Alport nephropathy progression by stimulating metabolic pathways, JOURNAL OF CLINICAL INVESTIGATION 2015;125:141-156.

[6] Kole R, Krieg AM. Exon skipping therapy for Duchenne muscular dystrophy, ADVANCED

DRUG DELIVERY REVIEWS 2015;87:104-107.

[7] Lentz JJ, Jodelka FM, Hinrich AJ et al. Rescue of hearing and vestibular function by antisense oligonucleotides in a mouse model of human deafness, NATURE MEDICINE 2013;19:345-350.

[8] Matsui M, Corey DR. Non-coding RNAs as drug targets, NATURE REVIEWS DRUG DISCOVERY 2017;16:167.

*10) References should be checked carefully. For example, the year of reference [2] should be 2016. In addition, there exists too many grammatical errors throughout the manuscript. The authors should correct the errors and polish the language in the manuscript.*

**Reply:** Thank you for making us correct our mistake through your valuable comment. We have carefully done the needed correction. Kindly check the reference.

# Response to Reviewer #3

We appreciate the positive feedback and sincerely thank the reviewer for the thoughtful and supportive recommendations. Our responses to the comments from reviewer are given below.

**Comments to the Author**

DTI-CDF is machine learning based method to predict drug target interactions based on a cascade deep forest model which integrates hybrid features, including multiple similarity-based features (FP2 fingerprint FP2) extracted from the heterogeneous graph, fingerprints of drugs, and evolution information of target protein (psePSSM) sequences

*1. There are many published methods, which integrated fingerprints (for chemicals) and evolutionary features for proteins. But very few has added protein structural features. In order to further increase model efficiency, it will be great if authors can also add protein structural features. Drugs bind to proteins on certain binding sites and information about those binding sites could be very effective for drug target interaction prediction. Information for those binding sites can be obtained by computing structural features of proteins. There are some toolboxes (e.g. Rosetta) which provide this information by giving pdb structure as input. Kindly add structural features and re-evaluate mode performance.*

**Reply:** We sincerely thank the reviewer for the useful suggestions, which are of great help to improve the quality of this manuscript. As advised by the reviewer, we have considered using the structural features of proteins as input to the model to enhance the predictive performance of the model, especially the potential binding sites of the protein. However, the three-dimensional structure of most of the proteins in this database has not been reported experimentally and the corresponding information cannot be obtained. In addition, the simple amino acid sequence predicts that the interaction of drugs with proteins (especially complex and unknown proteins) makes this study more widely available.

*2. It says on page 8: "Finally, we verify that the proposed DTI-CDF method is significantly better than the current state-of-art methods available". Can you please list down at least few successful predictions, which are also experimentally verified? You do not necessarily need to perform experimental verification, but it will be great if you can link at least some of your impactful predictions with successful experimental validations, which are already published by someone.*

**Reply:** Thank you for your graciousness and compelling advice. We would like you to kindly note that Table 1 shows the new interactions that DTI-CDF method predicted on four data sets which exist in one or two reference databases, that is, DrugBank [1] and KEGG [2]. We would like to mention that these databases are still being updated, as new DTIs are discovered, the number of new DTIs correctly predicted by this method may increase in the future. The encouraging result is that DTI-CDF can successfully detect 1352 new interactions that are not in the existing data set, which means that DTI-CDF is very effective in predicting new real DTIs from sparse matrices composed of very few DTIs. Both the new and verified DTIs are placed in the website (https://github.com/a96123155/DTI-

CDF/tree/master/3_new_DTIs), more details can be seen from it. We have added the description into Section 4 page 19. (red color words in revised manuscript)

**Table 1.** The details of the new drug-target interactions (DTIs) under the four data sets, including the number, the number that has been reported by the KEGG (abbreviated as K) and DrugBank (abbreviated as D) databases, and list five examples.

| | Nuclear receptors | | G-protein-coupled receptors | Ion channels | Enzymes |
|---|---|---|---|---|---|
| New DTIs | 84 | | 500 | 1158 | 1650 |
| Reported new DTIs | All | 74 | 338 | 460 | 480 |
| | K | 55 | 229 | 322 | 299 |
| | D | 72 | 303 | 284 | 326 |
| | Examples | (D00094,hsa6256)<br>(D00898,hsa2099)<br>(D00105,hsa2100)<br>(D01217,hsa5241)<br>(D02367,hsa2099)<br>… | (D00523,hsa185)<br>(D00136,hsa1813)<br>(D00454,hsa3362)<br>(D00454,hsa155)<br>(D02149,hsa151)<br>… | (D00548,hsa2554)<br>(D00035,hsa6534)<br>(D00547,hsa2890)<br>(D00775,hsa2892)<br>(D00035,hsa11254)<br>… | (D02335,hsa2677)<br>(D00650,hsa759)<br>(D03882,hsa240)<br>(D02561,hsa4129)<br>(D00448,hsa1147)<br>… |

**Reference:**

[1] Wishart DS, Knox C, Guo AC et al. DrugBank: a knowledgebase for drugs, drug actions and drug targets, NUCLEIC ACIDS RESEARCH 2007;36:D901-D906.

[2] Kanehisa M, Furumichi M, Tanabe M et al. KEGG: new perspectives on genomes, pathways, diseases and drugs, NUCLEIC ACIDS RESEARCH 2016;45:D353-D361.

*3. Did you trained your model only on approved drugs or also other compounds? If also on other compounds, then what are those? Can you please clarify this?*

**Reply:** We sincerely thank the reviewer for the useful suggestions, Yes, all drugs in the data set used in this study are approved drugs which can be searchable at KEGG DRUG database [1] which is a comprehensive drug information resource for drugs approved in Japan, the United States, and Europe. As the resource is highly reliable so the results obtained from testing on those drugs have high accuracy which overall enhances the reliability of the manuscript. We have clarified that these drugs are indeed approved. More details can be seen from paragraph 1 of Section 2.1 in page 7. (red color words in revised manuscript)

**Reference:**

[1] Kanehisa M, Furumichi M, Tanabe M et al. KEGG: new perspectives on genomes, pathways, diseases and drugs, NUCLEIC ACIDS RESEARCH 2016;45:D353-D361.

*4. It says on page 8: "In order to simulate more practically, in these four data sets, we consider the entire space of the DTIs, where the number of known DTIs (or positive samples) is much lower than the number of unknown DTIs (or negative samples)". How do you define negative samples, it's not clear to me? Can you please clarify it?*

**Reply:** We sincerely thank the reviewer for asking such a vital. In order to improve the quality of the manuscript and clarify the doubts raised by the respected reviewer we have made the following changes. We believe that the changes made by us will clear the confusion and will make the manuscript easily understandable. To simulate the natural condition, the negative data contains all unknown or non-existing drug-target interactions, and set the label as 0. We have clarified the definition of negative samples in revised manuscript in Section 2.1, paragraph 2 in page 7. (red color words in revised manuscript)

*5. It says on page 10: "To solve this problem, we adopt the method of neighbor-based interaction-profile inferring [23] to calculate this kernel.". Can you please add 1-2 sentences on how neighbor-based interaction-profile was just to fill in missing drug target pairs?*

**Reply:** Thank you for your prominent advice. We hope that the addition made by us in order to improve the quality of the manuscript will satisfy the reviewer. The inference of the similarity of an unknown drug (or target) to a particular target (or drug) is done using the five neighbors of the unknown drug (or target), expressed as the ratio of the sum of the similarities of neighbors that interact with a particular target (or drug) to the sum of all five neighbors' similarities. We have carefully demonstrated the mathematical definition and specific clarification for the kernel calculation in paragraph 2 of Section 2.2 in page 9. (red color words in revised manuscript)

**Reference:**

[1] Mei J, Kwoh C, Yang P et al. Drug-target interaction prediction by learning from local information and neighbors, BIOINFORMATICS 2012;29:238-245.

*6. You mentioned that there were 6 kernels, 3 for side effects, probably one for protein kernels and one for Gaussian interaction profile, which makes 5. What is 6th kernel for?*

**Reply:** We sincerely thank the reviewer for raising this important question. As it is already understood that the 5th kernel is for Gaussian interaction profile (GIP) we would like to clear the doubt by mentioning that, Gaussian interaction profile (GIP) includes GIP similarity for drugs and for targets which makes the kernels 5 and 6.

*7. It says on page 13: "The GIP similarity is constructed according to training data" How did you compute GIP similarity. Is it based on shared target space between drug pair or how, kindly explicitly mention it?*

**Reply:** We sincerely thank the reviewer for the useful suggestions, which are of great help to improve the quality of this manuscript. The similarity profiles consist of two parts: drug interaction profiles and target interaction profiles. The construction of interaction profiles is based on the assumption that similar drugs (or targets) tend to interact with the similar targets (or drugs). Based on the similarity

assumption, the interaction profiles are constructed as following: (1) identifying the drugs and targets in the interaction network as the vertices; (2) the edges that have drug-target interaction are attached with a value 1, and conversely those having no drug-target interactions are attached with a value 0. In fact, the drug interaction profiles can be obtained by transposing the target interaction profiles. The target interaction profile $y_{di}$ of one drug $d_i$ is turned out to be a column vector of target interaction profiles. Furthermore, Gaussian interaction profile (GIP) can be computed by the Gaussian kernel when give two target interaction profiles: $K_{GIP}(d_i, d_j) = e^{-\gamma_d \left\| y_{d_i} - y_{d_j} \right\|^2}$, where the parameter $\gamma_d$ controls the kernel bandwidth given by

$$\gamma_d = \tilde{\gamma}_d \left( \frac{1}{n_d} \sum_{i=1}^{n_d} |y_{d_i}|^2 \right)$$

where $n_d$ is the number of drugs. This kernel is independent of the size of the data set because of normalization, and $\tilde{\gamma}_d$ could be set with cross-validation to adapt different situations. In this study, the $\tilde{\gamma}_d$ is set to 1 simply. The GIP similarity for targets can be calculated analogously. We have added the specific calculation of GIPs in paragraph 2 of Section 2.2 in page 9. (red color words in revised manuscript)

**Reference:**

[1] van Laarhoven T, Nabuurs SB, Marchiori E. Gaussian interaction profile kernels for predicting drug-target interaction, BIOINFORMATICS 2011;27:3036-3043.

*8. Please define NR, GPCR, IC, E in figure caption in figure 2*

**Reply:** Thank you for your effective advice. We have explained all abbreviations in the caption.

# Response to Reviewer #4

We appreciate the positive feedback and sincerely thank the reviewer for the thoughtful and supportive recommendations. Our responses to the comments from reviewer are given below.

**Comments to the Author**

Chu et al. present a relevant new method/in silico model for target prediction. The manuscript clearly presents original research work and after reading the journal guidelines I doubt it is in the scope of this journal, which defines itself as a review journal. However, the decision is clearly with the Editorial Office.

Some severe language issues are found throughout the manuscript that make the text not only difficult to read but many statements also ambiguous. Just one example: pg7, ln 53 refers to "the above two machine learning-based methods", which the reader will likely be unable to identify unambiguously.

*A major comment is that the authors should show the performance of their model also on external data/holdout data, not just during cross-validation.*

**Reply:** We would be obliged if you draw your kind attention towards this point. The evaluation of the generalization error of a model can be completed by experimental tests. Therefore, it is necessary to use the test set to test the learner's discriminating ability to the new sample, and then use the test error on the test set as an approximation of the generalization error. It is generally assumed that the test samples are independent and identically distributed. It should kindly be noted that the test set should be as mutually exclusive as the training set, that is, the test sample should not appear in the training set as much as possible or not used in the training process. For the data set $\boldsymbol{D}$, the training set $\boldsymbol{S}$ and the test set $\boldsymbol{T}$ can be generated from the appropriate processing, and the following methods are commonly used:

Hold-out: The data set $\boldsymbol{D}$ is directly divided into two mutually exclusive sets, one of which is used as the training set $\boldsymbol{S}$ and the other as the test set $\boldsymbol{T}$, that is, $\boldsymbol{D} = \boldsymbol{S} \cup \boldsymbol{T}$, $\boldsymbol{S} \cap \boldsymbol{T} = \emptyset$. After training the model on $\boldsymbol{S}$, $\boldsymbol{T}$ is used to evaluate its test error as an estimate of the generalization error. It should be noted that different partitioning methods will result in different training sets and test sets. Correspondingly, the results of the model evaluation will also be different. Therefore, the estimation results obtained by a single usage of hold-out are not stable enough. When using hold-out, it is

generally necessary to use several random divisions to repeat the experimental evaluation and take the average value as the evaluation result of the hold-out.

Cross-validation: The cross-validation first divides the data set $\boldsymbol{D}$ into $k$ mutually similar size subsets, namely $\boldsymbol{D} = \boldsymbol{D_1} \cup \boldsymbol{D_2} \cup ... \cup \boldsymbol{D_k},\ \boldsymbol{D_i} \cap \boldsymbol{D_j} = \emptyset\ \ (i \neq j)$. Each time, the union of $k$-1 subsets is used as the training set, and the remaining subset is used as the test set, so that the $k$-group training set and the test set can be obtained and performed, finally the mean of $k$ test results returned. Obviously, the stability and fidelity of cross-validation evaluation results depend to a large extent on the value of $k$. To emphasize this, cross-validation is usually called $k$-fold cross validation, and the most common value of $k$ is 10, called 10-fold cross-validation. Similar to hold-out, there are also multiple ways to divide data set $\boldsymbol{D}$ into $k$ subsets. In order to reduce the differences introduced by the difference in sample partitioning, $k$-fold cross validation usually uses different partitions randomly and repeats $p$ times. The final evaluation result is the mean of the $k$-fold cross-validation results for the $p$-repeats, such as the 10-fold cross-validation with 5 replicates.

Although repeated 5 times of 10-fold cross-validation and repeated 50 hold-outs were performed 50 times of training and testing, studies have shown that from the perspective of statistical inference, the results obtained from 10-fold cross-validation are the best which makes it the method in model selection [1]. Therefore, this study used a random state of 5 to perform a 10-fold cross-validation.

When constructing the data set, we use known drug-target interactions as positive samples, unknown or non-existing interactions as negative samples. The goal of this study was not only to obtain a high-performance drug-target interaction prediction method for the constructed data set, but also to predict potential drug-target interactions that have not yet been experimentally discovered or reported. Therefore, each negative sample needs to be predicted in the test set. To achieve this goal, we combined hold-out and cross-validation methods, using 10-fold cross-validation to divide the training set and test set to ensure that each negative sample will appear in a certain test set and be predicted. Subsequently, the corresponding training set and test set are respectively trained and tested by hold-out, that is, the model selection is determined by the performance of the training set, and the test set is only used for performance evaluation of the model and prediction of unknown interaction. In order to ensure the stability and fidelity of the evaluation results, the above process is repeated 5 times using different

random states, and then the average value is taken as the final evaluation result.

This study did not use external data because each model was specific to a particular target and it was difficult to find other matched drug-target interaction pairs for independent testing. In addition, the practical prediction ability of our method was also confirmed by mapping with the latest version of online biological databases such as DrugBank [2] and KEGG [3], to ensure that whether the data noise (i.e. the reported drug-target interactions but not in the data set in this study) in the negative sample can be correctly identified. Table 1 shows the new interactions that the DTI-CDF method predicts on four data sets that exist in one or more reference databases, and it shows encouraging results and proves the effectiveness of the method. The revisions can be seen from Section 2.5 paragraph 1 in page 11 and Section 4 paragraph 1 in page 19. (red color words in revised manuscript)

**Table 1.** The details of the new drug-target interactions (DTIs) under the four data sets, including the number, the number that has been reported by the KEGG (abbreviated as K) and DrugBank (abbreviated as D) datasets.

| | | Nuclear receptors | G-protein-coupled receptors | Ion channels | Enzymes |
|---|---|---|---|---|---|
| New DTIs | | 84 | 500 | 1158 | 1650 |
| Reported new DTIs | All | 74 | 338 | 460 | 480 |
| | K | 55 | 229 | 322 | 299 |
| | D | 72 | 303 | 284 | 326 |

**Reference:**

[1] Kohavi R (1995), 'A study of cross-validation and bootstrap for accuracy estimation and model selection', *Ijcai*, Montreal, Canada, pp. 1137-1145.

[2] Wishart DS, Knox C, Guo AC et al. DrugBank: a knowledgebase for drugs, drug actions and drug targets, NUCLEIC ACIDS RESEARCH 2007;36:D901-D906.

[3] Kanehisa M, Furumichi M, Tanabe M et al. KEGG: new perspectives on genomes, pathways, diseases and drugs, NUCLEIC ACIDS RESEARCH 2016;45:D353-D361.

*Minor comments:*

*- Abstract: "Computational prediction of DTIs has become a popular alternative strategy to the experimental methods."They are complementary, not alternative strategies. They can provide guidance, prioritize experiments, but never replace experiments.*

**Reply:** Thank you for the prominent advice. We highly agree with the point that Computational prediction cannot replace experimental methods and experimental methods still remains the method of utmost importance. Therefore, without denying with the fact mentioned by reviewer we are presenting the revised abstract of the manuscript. Revised abstract is: "Computational prediction of DTIs can effectively complement experimental wet-lab techniques for identification of DTIs, which are typically time- and resource-consuming." in page 3 Abstract part. (red color words in revised manuscript)

*- In section 2.4, an explanation of the three experimental settings, sp, sd and st, should be added.*

**Reply:** Thank you for the crucial suggestion. With an intention to make the manuscript widely readable and easily understandable an explanation of $S_p$, $S_D$ and $S_T$ have been added. We have clarified the three abbreviations in Section 2.4. (red color words in revised manuscript)

*- In Figure 2 and elsewhere abbreviations should be explained in the figure captions. Some readers screening the figures will otherwise be unable to interpret the figures without reading the full article.*

**Reply**: Thank you for your effective advice. All the abbreviations in captions have been clarified.

# Response to Reviewer #5

We appreciate the positive feedback and sincerely thank the reviewer for the thoughtful and supportive recommendations. Our responses to the comments from reviewer are given below.

**Comments to the Author**

Inference of the drug-target interactions is critical for the discovery of new drugs or novel application of existing drugs. Experimental investigation of the interactions between drug and targets are very costly and even infeasible in many scenarios. Computational prediction of drug-target interactions can significantly narrow down the searching list of true interactions in the studied biological system. However, existing Drug-interaction prediction methods are suffering from low precision and high false positive rate, which is becoming a bottleneck in the drug discovery field. In this work, the authors collected a list of highly informative features associated with the drug-target interactions and used a set of stacked supervised random forest classifiers to predict the interactions between drug and targets. The authors demonstrated the superiority of their methods on several benchmarking datasets compared with other state-of-the-art methods. The software that the authors have developed is freely available at Github. I believe that many biomedical researchers in the drug discovery and development field, including myself, can be benefited enormously by this fantastic work. However, I still have a few relatively minor comments, and I hope the authors may find them useful when revising the manuscript.

*(1) In this study, the authors introduced 3 sets of features. Many of those features might be redundant. Although generally speaking, redundant features won't affect the random forest prediction performance much, it would still affect our evaluation of these features. Some features could be highly correlated and might be unnecessary. By removing those redundant features, the learning would be relatively faster. Also, it will decrease harmful bias and improve the interpretability of the model. Even a simple correlation analysis between all the collected features in this study can help the readers better interpret the model, especially the features.*

**Reply:** In machine learning applications, data features generally contain irrelevant features and redundant features, which can be interfering the training process. The process of selecting relevant features from a given set of features is called feature selection. It should be noted that the feature selection process must ensure that important features are not lost, otherwise the subsequent learning process will not achieve good performance due to the lack of important information. Common feature

selection methods can be divided into the following three categories:

(1) Filter: Firstly, the feature set is selected from the initial features by the feature selection process, and then the learner is trained with the filtered features. The feature selection process is independent of the subsequent learners. Its representative method is Relief [1].

(2) Wrapper: The performance of the learner is used as the evaluation criterion of the feature subset, which is the biggest difference from the filter feature selection method. The purpose of this method is to select a subset of features that are most beneficial to the performance for a given learner. Las Vegas Wrapper [2] is a typical wrapper feature selection method.

(3) Embedding: Different from the above two methods, this method integrates the feature selection process and the learner training process, and both them are completed in the same optimization process, that is, the feature selection is automatically performed during the training of the learner. The most typical one is the decision tree algorithm [3], which must select a feature at each step of the tree growth process. The basis of the selection is usually the purity of the divided child nodes. The purer the divided child nodes, the division effect better. Thus, the process of decision tree generation is the process of feature selection.

This study considers data characteristics and individual learners, and finally adopts the embedding feature selection method. The three types of features (i.e. PathCS, FP2 and PsePSSM) of this study were designed for the task that predicting drug-target interactions, so irrelevant features may not be considered. PsePSSM and FP2 take into account the continuity of the sequence or structure in the calculation, resulting in redundancy within the feature. Although redundant features do not work in many cases, removing them can reduce the burden of the learning process. However, when the redundant features correspond exactly to the "intermediate concept" required to complete the learning task, the redundant features can reduce the difficulty of the learning task, which is beneficial. Therefore, this study focuses on the selection of redundant features.

On the other hand, the cascade deep forest model used in this study is based on decision trees, its individual learners, including random forest and XGBoost, have embedded feature selection mechanisms, that is column subsampling technology. For each node of the base decision tree, $k$ attributes are randomly selected from the attribute set of the node as the feature set, and then an optimal

attribute is selected for division. The parameter *k* is determined by the performance of the training set, which controls the degree of introduction of randomness. This method makes the training process of each base learner adopt different feature subsets, which not only increases the diversity of the model but also reduces the computational cost, and eliminates redundant features to some extent to reduce the risk of over-fitting. Moreover, due to the large number of trees contained in the model, almost every feature participates in model training to prevent data loss.

Furthermore, we delete FP2 and PsePSSM in this study and only use PathCS as the input feature vector, for the following reason:

In the data set construction process, for each data set, the sample space of this study is the Cartesian product space of drugs and targets, that is, the number of samples in each data set is the product of the number of drugs and the number of targets, each drug-target pair as a sample. Among them, the drug-target pair of known interaction is used as the positive sample, and the rest are processed as negative samples, which leads to the presence of noise in the negative samples (the real existence of drug-target interactions not yet discovered). Therefore, when the performance evaluation metrics of the model is extremely high in this study, it cannot be considered that the model performance is good in fact because the model may not correctly judge and predict the noise samples. The original intention of this study is to effectively predict the undiscovered drug-target interactions, not only aims to obtain a high accuracy model.

To determine whether the model can effectively predict unknown but real drug-target interactions, we use drug-target interactions that have been reported but not yet included in the data set to judge. For each negative drug-target pair which is predicted as positive, we search it in KEGG [4] and DrugBank [5] databases to determine whether it is predicted successfully. In the prediction results of the model using only PathCS, more than 1000 drug-target pairs are determined as successful predictions as they can be found in the above two databases, and these new DTIs will be detailed in the article. But models that use composite features (i.e. the combination of PathCS, FP2 and PsePSSM) do not achieve such encouraging results. Therefore, in order to strike a balance between high performance and effective prediction of undiscovered drug-target interactions, we decided to use only PathCS as the input feature of the model. Although the performance of the model is lost to some extent,

the above-mentioned undiscovered drug-target interactions is effectively predicted which is more important, and the computational speed of the method is improved. This also leads to a significant reduction in the feature dimension, so it is sufficient to use only embedding method as the feature selection method.

**Reference:**

[1] Kira K, Rendell LA. A practical approach to feature selection. Machine Learning Proceedings 1992.: Elsevier, 1992, 249-256.

[2] Liu H, Setiono R (1997), 'Feature selection and classification-a probabilistic wrapper approach', *Proceedings of 9th International Conference on Industrial and Engineering Applications of AI and ES*, pp. 419-424.

[3] Safavian SR, Landgrebe D. A survey of decision tree classifier methodology, IEEE transactions on systems, man, and cybernetics 1991;21:660-674.

[4] Kanehisa M, Furumichi M, Tanabe M et al. KEGG: new perspectives on genomes, pathways, diseases and drugs, NUCLEIC ACIDS RESEARCH 2016;45:D353-D361.

[5] Wishart DS, Knox C, Guo AC et al. DrugBank: a knowledgebase for drugs, drug actions and drug targets, NUCLEIC ACIDS RESEARCH 2007;36:D901-D906.

*(2) In this paper, the authors barely talked about why does the stacked random forest model can improve the performance over a conventional random forest model. I would like to see a direct performance comparison with a conventional random forest model. I am also curious about the time/memory efficiency of the stacked random forest model compared with a simple random forest and other methods benchmarked in this study. It will help the users to evaluate the gain in the prediction performance vs. the sacrifice in running efficiency and choose appropriate methods based on their specific needs.*

**Reply:** Most studies about deep learning are based on neural network models, where many layers of parameterized nonlinear differentiable modules are trained by backpropagation. Recently, it has been shown that deep learning can also be realized by non-differentiable modules without backpropagation

training called deep forest. However, compared to traditional machine learning methods, high memory requirements and high time costs inhibit the training of large models. For memory aspect, the gc.set_keep_model_in_mem(False) command is provided during the implementation of the deep forest model so that the model can be not stored in memory.

In this study, random forest (RF), XGBoost (XGB) and cascade deep forest (CDF) of the first three small data sets (i.e. nuclear receptors, G-protein-coupled receptors and ion channels) are trained with a PC with an Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz, and that of the biggest data set (i.e. enzymes) is trained on two Intel(R) Xeon(R) Gold 6230 CPU @ 2.10GHz. In addition, the deep neural network (DNN) is trained on a NVIDIA's GeForce GTX 1080 Graphics Cards GPU. In order to help users to evaluate the gain in the prediction performance and the sacrifice in running efficiency, we compare the performance, process time on RF, XGB and CDF. We didn't compare CDF with DNN on time because DNN is trained on GPU. And some studies indicate that deep forest model is faster than multilayer perceptron on IMDB dataset [1] with CPU. In addition, RF and XGB are parallel ensemble methods [2], the efficiency of CDF can be improved further with optimized parallel implementation. It is also noteworthy that the above comparison is somewhat unfair to CDF, because DNN have too many hyper-parameters, and the learning performance is heavily dependent on their careful tuning that makes the training very difficult, many different architectures have been tried for DNN to achieve the reported performance but these time costs are not included. The comparison results are shown in Table 1. Although the time of CDF is not quite satisfactory, it is acceptable as the best performance. Thus, it is worthwhile to sacrifice time in exchange for higher accuracy.

In order to reduce the calculation cost of CDF, optimization can be achieved through distributed implementation, hardware foundation and algorithms, which is also an important direction for future development. At the algorithm level, an approach called gcForestCS [3] has been proposed to improve the efficiency of deep forest by confidence screening mechanism. It varies the model complexity from low to high as the level increases in the cascade, which further reduces the memory requirement and time cost. The experiments show that the proposed approach achieves highly competitive predictive performance with significantly reduced time cost and memory requirement by up to one order of magnitude. We do not intend to place the time and memory cost analysis in Table 1 into the revised

manuscript. Instead, we are more concerned with the practical performance of our model although sometimes it requires more time and memory cost. However, it is still worthwhile to exploit new models to enhance the DTI performance. We have added the performance comparison of our model with random forest, XGBoost and DNN in Section 3.1 and 3.2 page 14-17.

**Table 1.** The performance and process time among random forest (RF), XGBoost (XGB), deep neural network (DNN) and cascade deep forest (CDF).

| Data set | Experimental setting | Model | AUPR | AUC | $F_2$-score | Average of AUPR, AUC and $F_2$-score | Time (hours) |
|---|---|---|---|---|---|---|---|
| Nuclear receptors | $S_p$ | RF | 0.86 | 0.97 | 0.81 | 0.88 | 0.02 |
| | | XGB | 0.90 | 0.95 | 0.84 | 0.90 | 0.01 |
| | | DNN | 0.85 | 0.97 | 0.74 | 0.85 | - |
| | | CDF | 0.93 | 0.98 | 0.84 | 0.92 | 0.64 |
| | $S_D$ | RF | 0.78 | 0.91 | 0.75 | 0.81 | 0.02 |
| | | XGB | 0.76 | 0.87 | 0.69 | 0.77 | 0.01 |
| | | DNN | 0.77 | 0.92 | 0.68 | 0.79 | - |
| | | CDF | 0.79 | 0.92 | 0.77 | 0.82 | 0.81 |
| | $S_T$ | RF | 0.68 | 0.89 | 0.66 | 0.74 | 0.02 |
| | | XGB | 0.74 | 0.88 | 0.72 | 0.78 | 0.01 |
| | | DNN | 0.68 | 0.90 | 0.43 | 0.67 | - |
| | | CDF | 0.76 | 0.91 | 0.72 | 0.80 | 0.24 |
| G-protein-coupled receptors | $S_p$ | RF | 0.77 | 0.97 | 0.71 | 0.82 | 0.03 |
| | | XGB | 0.88 | 0.94 | 0.82 | 0.88 | 0.11 |
| | | DNN | 0.80 | 0.95 | 0.67 | 0.81 | - |
| | | CDF | 0.90 | 0.98 | 0.84 | 0.90 | 2.66 |
| | $S_D$ | RF | 0.71 | 0.94 | 0.66 | 0.77 | 0.03 |
| | | XGB | 0.75 | 0.86 | 0.67 | 0.76 | 0.10 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | DNN | 0.72 | 0.92 | 0.59 | 0.74 | - |
| | | CDF | 0.80 | 0.95 | 0.75 | 0.83 | 0.95 |
| | $S_T$ | RF | 0.64 | 0.95 | 0.65 | 0.75 | 0.04 |
| | | XGB | 0.77 | 0.90 | 0.70 | 0.79 | 0.11 |
| | | DNN | 0.65 | 0.94 | 0.53 | 0.71 | - |
| | | CDF | 0.79 | 0.96 | 0.76 | 0.83 | 3.39 |
| Ion channels | $S_p$ | RF | 0.90 | 0.99 | 0.85 | 0.91 | 0.07 |
| | | XGB | 0.96 | 0.98 | 0.92 | 0.95 | 0.31 |
| | | DNN | 0.90 | 0.99 | 0.79 | 0.89 | - |
| | | CDF | 0.96 | 0.99 | 0.93 | 0.96 | 4.81 |
| | $S_D$ | RF | 0.79 | 0.92 | 0.73 | 0.82 | 0.06 |
| | | XGB | 0.82 | 0.89 | 0.75 | 0.82 | 0.29 |
| | | DNN | 0.79 | 0.91 | 0.70 | 0.80 | - |
| | | CDF | 0.84 | 0.93 | 0.78 | 0.85 | 2.68 |
| | $S_T$ | RF | 0.85 | 0.97 | 0.80 | 0.87 | 0.12 |
| | | XGB | 0.90 | 0.93 | 0.83 | 0.89 | 0.30 |
| | | DNN | 0.85 | 0.96 | 0.72 | 0.84 | - |
| | | CDF | 0.91 | 0.97 | 0.85 | 0.91 | 9.25 |
| Enzymes | $S_p$ | RF | 0.88 | 0.98 | 0.70 | 0.85 | 0.29 |
| | | XGB | 0.95 | 0.97 | 0.92 | 0.95 | 2.87 |
| | | DNN | 0.71 | 0.77 | 0.45 | 0.64 | - |
| | | CDF | 0.95 | 0.98 | 0.93 | 0.96 | 40.96 |
| | $S_D$ | RF | 0.79 | 0.95 | 0.73 | 0.82 | 0.58 |
| | | XGB | 0.87 | 0.92 | 0.78 | 0.86 | 3.71 |
| | | DNN | 0.64 | 0.75 | 0.38 | 0.59 | - |
| | | CDF | 0.87 | 0.95 | 0.83 | 0.88 | 14.83 |
| | $S_T$ | RF | 0.88 | 0.98 | 0.70 | 0.85 | 0.58 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | XGB | 0.91 | 0.94 | 0.87 | 0.91 | 3.85 |
| | | DNN | 0.66 | 0.78 | 0.47 | 0.64 | - |
| | | CDF | 0.95 | 0.98 | 0.93 | 0.96 | 26.31 |

**Reference:**

[1] Zhou Z, Feng J. Deep Forest, arXiv preprint arXiv:1702.08835 2017.

[2] Zhou Z. Ensemble methods: foundations and algorithms.: Chapman and Hall/CRC, 2012.

[3] Pang M, Ting K, Zhao P et al. (2018), 'Improving deep forest by confidence screening', *2018 IEEE International Conference on Data Mining (ICDM)*, IEEE, pp. 1194-1199.

*(3) I am suspecting that the performance improvement of the stacked random forest model is coming from the aggregation of the random forests and XGB models at different levels. In that case, it would be conceptually similar to a super ensemble model of the random forests and XGB models. A comparison between the stacked random forest model and the super ensemble of the random forests and XGB models (use the same # of random forests and XGB models, and average all the predicted probabilities) would help to clarify. I will really appreciate if the authors can somehow explain the potential reasons for using such a stacked random forest model.*

**Reply:** To verify the effect of the multi-leveled properties of cascade deep forest (CDF) on model performance, we compared the performance of a multi-levels CDF model with a single-level CDF model (i.e. the super ensemble model proposed by reviewers), where the individual learner consisted of one random forest (RF) and one XGBoost (XGB). The results shown in Table 2. For the 12 experimental cases (4 data sets and 3 experimental settings), there are 8 multi-levels CDF models better than the single-level CDF model. There are two cases where the single-level CDF model performs better and the other two have the same performance. It is worth noting that in order to clarify the effects of multi-levels effects, the multi-levels model forces the number of levels greater than 3. But in fact, single-level is also part of the CDF model, because the CDF is automatically terminated when the performance of the second level is not increased compared to the first level. That is to say, the CDF model can implement both the super ensemble model described by the reviewer and the depth

cascading, which depends on the performance.

**Table 2.** The comparison of the multi-levels cascades deep forest (CDF) model (abbreviated as M) with a single-level CDF model (abbreviated as *S*). The individual learners of the above two models are one random forest and one XGBoost.

| Data set | Experimental setting | CDF model | AUPR | AUC | $F_2$-score |
|---|---|---|---|---|---|
| Nuclear receptors | $S_p$ | M | 0.91 | 0.98 | 0.83 |
| | | S | 0.91 | 0.98 | 0.85 |
| | $S_D$ | M | 0.77 | 0.91 | 0.73 |
| | | S | 0.78 | 0.91 | 0.71 |
| | $S_T$ | M | 0.73 | 0.91 | 0.71 |
| | | S | 0.74 | 0.91 | 0.71 |
| G-protein-coupled receptors | $S_p$ | M | 0.89 | 0.98 | 0.84 |
| | | S | 0.89 | 0.98 | 0.84 |
| | $S_D$ | M | 0.79 | 0.95 | 0.75 |
| | | S | 0.78 | 0.94 | 0.75 |
| | $S_T$ | M | 0.78 | 0.96 | 0.76 |
| | | S | 0.78 | 0.95 | 0.75 |
| Ion channels | $S_p$ | M | 0.96 | 0.99 | 0.93 |
| | | S | 0.95 | 0.99 | 0.92 |
| | $S_D$ | M | 0.83 | 0.93 | 0.78 |
| | | S | 0.83 | 0.92 | 0.77 |
| | $S_T$ | M | 0.90 | 0.97 | 0.85 |
| | | S | 0.90 | 0.97 | 0.84 |
| Enzymes | $S_p$ | M | 0.95 | 0.98 | 0.93 |
| | | S | 0.94 | 0.98 | 0.92 |
| | $S_D$ | M | 0.86 | 0.95 | 0.83 |

| | | | | | |
|---|---|---|---|---|---|
| | | S | 0.86 | 0.95 | 0.83 |
| | $S_T$ | M | 0.91 | 0.96 | 0.88 |
| | | S | 0.90 | 0.96 | 0.88 |

*(4) If I understand correctly, in the stacked random forest model, the authors concatenate the predicted probability vector to the original input feature vector. The combined vector was used as the input for the next layer. However, this is not clearly described in the manuscript, at least not shown directly in the model overview figure 1 (the red bars shown at the bottom of the model are different from the original input feature vector shown at the top left). Also, as the predicted probability vector is extremely short (only 6 in the structured shown in figure 1, conceptually 3 and other 3 are redundant), I am really curious about the impact of these 6 features compared with hundreds of features in the original input vector. Could the authors provide the importance scores for the 6 features in the predicted probability vector?*

**Reply:** Thank you for your revise, we have modified all red bars in the figure 1 to the same, as shown in Figure 3 of Comment 6.

Feature scoring is an important way to enhance the interpretability of machine learning methods. This study used random forest (RF) and XGBoost (XGB) as base learners of cascade deep forest (CDF), which made it better than traditional learners through the diversity of models, samples and features. RF and XGB are representative of Bagging and Boosting, respectively, and the methods for measuring the importance of each feature are different. Because the two basic learners score features differently, it is difficult to unify their respective scoring metrics into one feature scoring space.

The CDF model proposed by the research is the most advanced ensemble deep model of traditional machine learning methods. Its application in drug-target interactions has better performance than previous research, which is derived from the integration of advantages on traditional base learners. In the usual $n$-classification problem, the redundancy of the prediction probability is $1/n$ (one of the $n$ prediction probabilities can be obtained by the other $n-1$). Since drug-target interaction prediction is generally a two-class problem, the redundancy rate is $1/2$. It is worth noting that this redundancy

rate may slow down the training efficiency of the model to some extent, but intuitively, this redundancy has little or no negative impact on the accuracy of drug-target interaction prediction. Since the structure of the CDF model has been fixed, the problem of feature redundancy is that the method cannot be avoided at the beginning of design. This open question we expect future research to give a solution that can be one of the goals of future optimization.

*(5) Did the authors remove the outliers in the data when they are training the model? If not, I would recommend doing that.*

**Reply:** Thanks for your fruitful advice.

Most data sets contain one or several unusual observations. It is considered an outlier when the observation is different from most data or is less likely under the assumed probability model of the data. Unusual observations of data for a single feature are observations that are very large or very small relative to other points. For example, if a normal distribution is assumed, any observation of the absolute value of the standard deviation is usually identified as an outlier. However, due to the many features, the situation becomes complicated. In the high dimension, when each dimension is considered separately, there may be outliers that do not exhibit anomalous observations and therefore are not detected from the univariate criteria. Therefore, all features need to consider together using a multivariate approach.

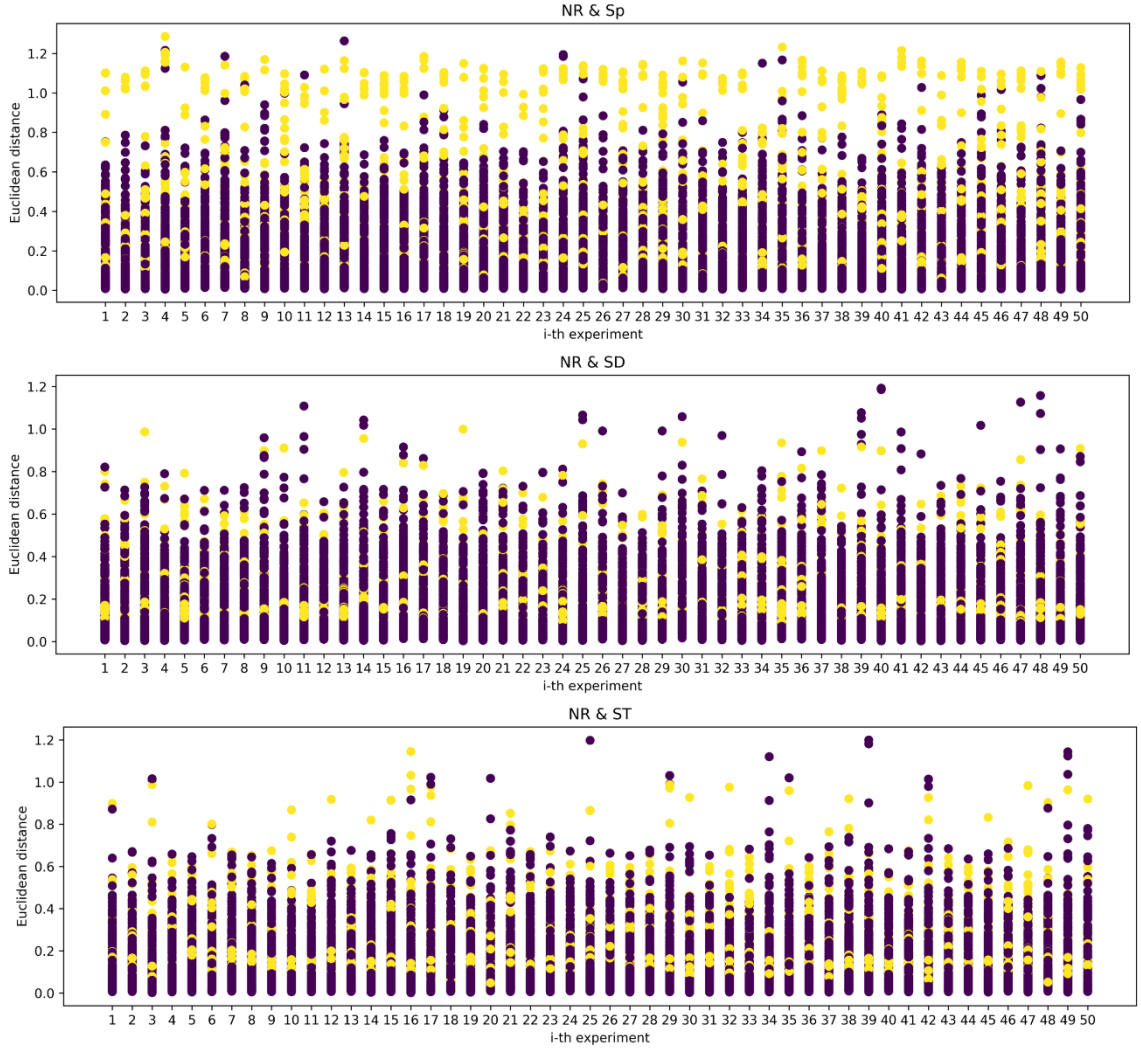
There are many methods for detecting outliers. Common methods are statistical-based methods and lower dimension projections [1]. Many statistical-based outlier detection methods assume a normal distribution or by using a central limit theorem, requiring that the number of features be greater than 30 [2-3]. Thus, in this study, we used principal component analysis (PCA) [1] which belong to projection-based method to do outlier detection.

The key idea of this approach is to use PCA to find outliers that violate the correlation between data samples. To find these outliers, the PCA-based algorithm projects the raw data from the original space into the principal component space and then pulls the projection back into the original space. If only the first $k$ principal components are used for projection and reconstruction, the distance (or error) between raw data and reconstruction data is small for most data, but for the outliers, the distance is

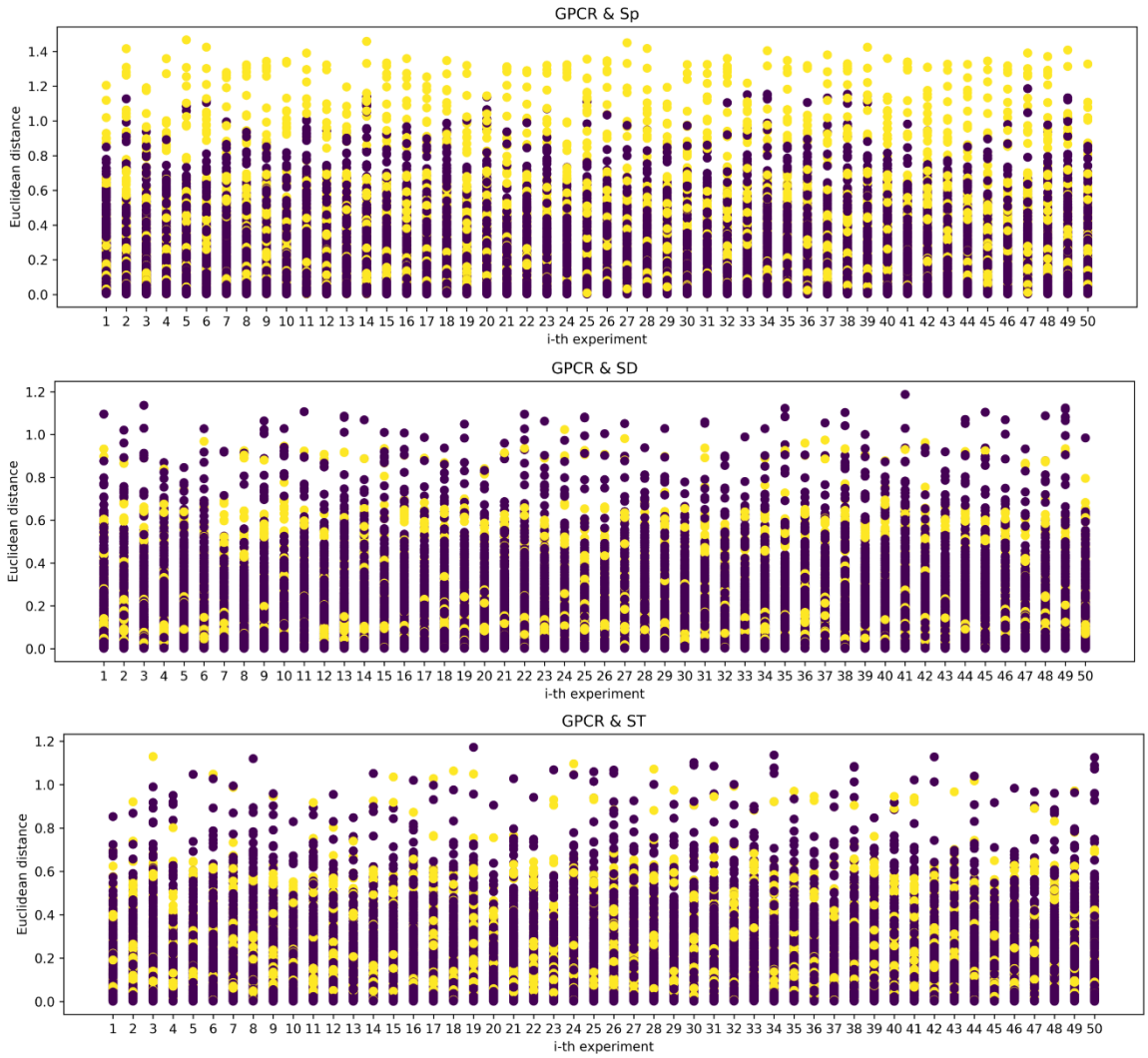
relatively large. This is because the first *k* principal components reflect the variance of the normal sample, and the latter principal components reflect the variance of the abnormal points. *k* is a hyper-parameter and this study is set to 3. The most familiar distance metric is the Euclidean distance [4].

Suppose *M* is a data set with *N* number of *p*-dimensional samples whose covariance matrix is *X*. According to the singular value decomposition, $X = PDP^T$, where *P* is an orthogonal matrix of $p \times p$, and each of its columns is a eigenvector of *X*. *D* is a diagonal matrix of $p \times p$, containing the eigenvalue $\lambda_1, \dots, \lambda_p$. The eigenvalues of *D* are sorted as a descending order, and each column of *P* is also adjusted accordingly, such that the *i*-th column of *P* corresponds to the *i*-th diagonal value of *D*. If top-*k* principal components are selected, the projection of data set *M* in the principal component space can be written as $Y^k = M \times P^k$, where $P^k$ is the first *k* column of the matrix *P*, and the dimension is $p \times k$, then $Y^k$ is a matrix of $N \times k$ dimensions. Next, the data set is reconstructed, $R = Y^k \times (P^k)^T$, where $R$ is a data set formed after reconstruction using top-*k* principal components, which is an $N \times p$ matrix. Finally, the outlier score of the data set *M* are defined as the Euclidean distances between *M* and *R*.

We performed outlier detection on all training sets used in this study (5 replicates of 10-fold cross validations performed in 4 data sets and 3 experimental settings, total $4 \times 3 \times 5 \times 10 = 600$). The result is shown in Reply Figure 1-4.

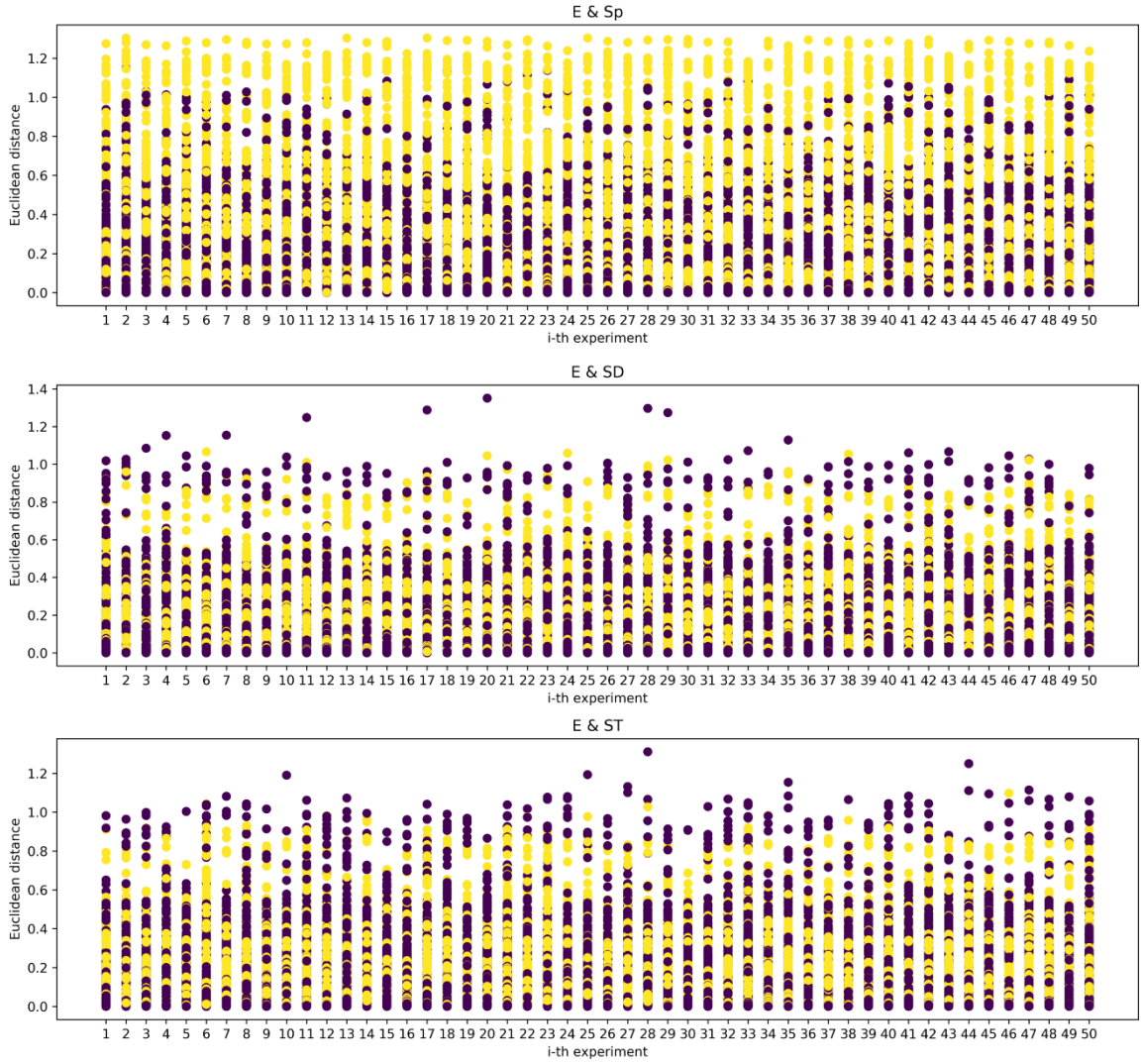



**Reply Figure 1.** The outlier detection results of nuclear receptors (NR) data set under three experimental settings ($S_p$, $S_D$, $S_T$), each of them has 50 experiments because of the 5 repeated 10-fold cross-validations. The yellow point and purple point are representing as the outlier score of according positive samples and negative samples, respectively. The outlier score is calculated as Euclidean distance, and the larger the value and away from other data points, the point is considered to be an outlier.




**Reply Figure 2.** The outlier detection results of G-protein-coupled receptors (GPCR) data set under three experimental settings ($S_p$, $S_D$, $S_T$), each of them has 50 experiments because of the 5 repeated 10-fold cross-validations. The yellow point and purple point are representing as the outlier score of according positive samples and negative samples, respectively. The outlier score is calculated as Euclidean distance, and the larger the value and away from other data points, the point is considered to be an outlier.


**Reply Figure 3.** The outlier detection results of ion channels (IC) data set under three experimental settings ($S_p$, $S_D$, $S_T$), each of them has 50 experiments because of the 5 repeated 10-fold cross-validations. The yellow point and purple point are representing as the outlier score of according positive samples and negative samples, respectively. The outlier score is calculated as Euclidean distance, and the larger the value and away from other data points, the point is considered to be an outlier.




**Reply Figure 4.** The outlier detection results of enzymes (E) data set under three experimental settings ($S_p$, $S_D$, $S_T$), each of them has 50 experiments because of the 5 repeated 10-fold cross-validations. The yellow point and purple point are representing as the outlier score of according positive samples and negative samples, respectively. The outlier score is calculated as Euclidean distance, and the larger the value and away from other data points, the point is considered to be an outlier.

As can be seen from the above four figures, most of the 50 experiments with different random state partitioned data sets under each experimental condition have no outliers. Moreover, even if there are outliers in the training data of some of the experiments, after averaging 50 experiments, it will not affect the performance of the final model much.

In addition, the sample point sparse area (i.e. the possible outlier point area) in the figure contains

both positive and negative samples, while positive samples are experimentally validated samples, and the number is very small, so it cannot be removed. Correspondingly, negative samples that are close to them should not be removed as outliers.

On the other hand, we suspect that these negative samples that can be used as outliers may be potential drug-target interactions that have not yet been discovered. In the data set construction process, for each data set, the sample space of this study is the Cartesian product space of drugs and targets, that is, the number of samples in each data set is the product of the number of drugs and the number of targets, each drug-target pair as a sample. Among them, the drug-target pair of known interaction is used as the positive sample, and the rest are processed as negative samples, which leads to the presence of noise in the negative samples or the outliers in the training data (the real existence of drug-target interactions not yet discovered). And the original intention of this study is to effectively predict the undiscovered drug-target interactions which represents in this study as noise or outliers, not just to obtain a high-performance model. This study holds the principle of simulating data conditions in the natural state to find new unknown drug-target interactions, so the outliers are not removed.

Finally, we reviewed the relevant literature on drug-target interaction studies and have not found any work to remove outliers.

**Reference:**

[1] Aggarwal CC (2015), 'Outlier analysis', *Data mining*, Springer, pp. 237-263.

[2] Ye N, Chen Q. An anomaly detection technique based on a chi‐square statistic for detecting intrusions into information systems, QUALITY AND RELIABILITY ENGINEERING INTERNATIONAL 2001;17:105-112.

[3] Ye N, Emran SM, Chen Q et al. Multivariate statistical analysis of audit trails for host-based intrusion detection, IEEE TRANSACTIONS ON COMPUTERS 2002;51:810-820.

[4] Shyu M, Chen S, Sarinnapakorn K et al. (2003), 'A novel anomaly detection scheme based on principal component classifier', MIAMI UNIV CORAL GABLES FL DEPT OF ELECTRICAL AND COMPUTER ENGINEERING.

*(6) In figure 1, 'layer #1" to 'layer # N" should be placed at the bottom of the classifiers instead of the predictions.*

**Reply:** Thank you for your effective opinion, we have modified the figure 1 as follows (shown in Figure 2):

(1) "layer #1" to "layer # N" are placed at the bottom of the classifiers instead of the predictions.
(2) the red bars shown at the bottom of the model are the same as the original input feature vector shown at the top left.
(3) In revised manuscript, the figure 1 has been changed to figure 2.

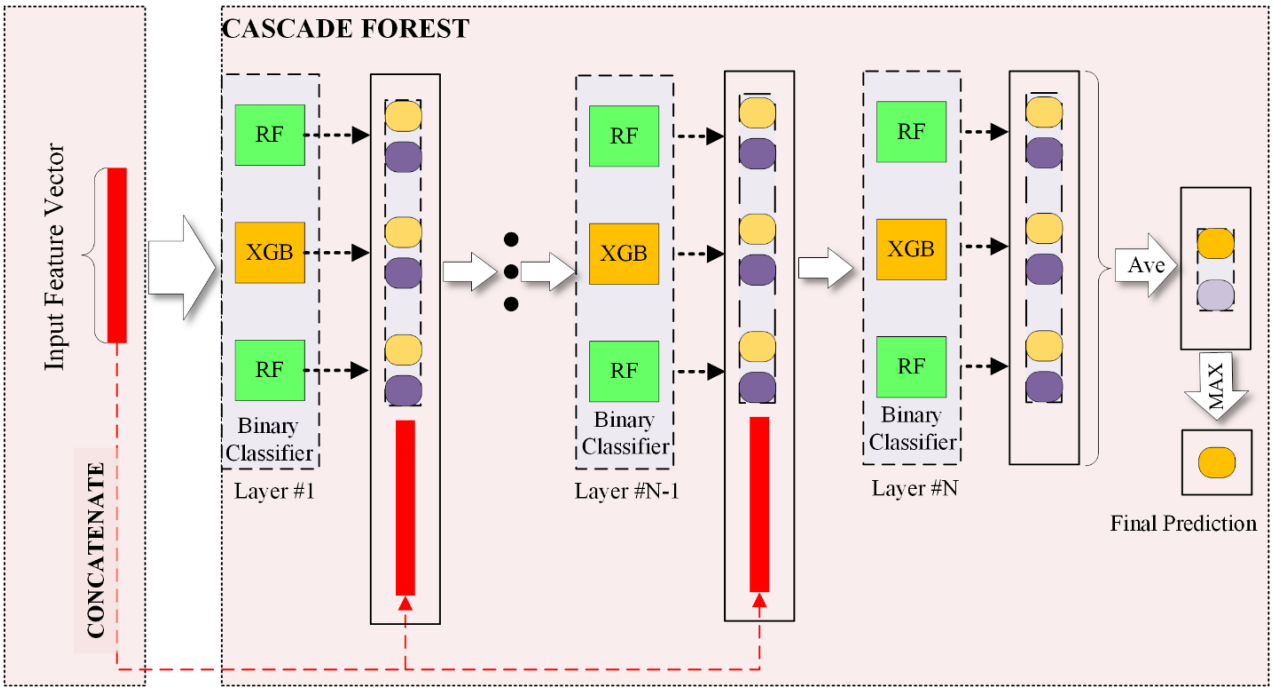



**Figure 2.** This machine learning model is composed of an input feature vector, a cascade deep forest (CDF) classifier, and a final prediction. In particular, CDF is the core unit of the model, which has six variants in this study. In each variant, each layer consists of a different number of random forest (RF) and XGBoost (XGB) binary classifiers and different layers own the same structure. The figure shows one special model in which each layer has two RF learners and one XGB learner, denoted as RF2-XGB1. Other variants are RF2, XGB2, RF1-XGB1, RF1-XGB2, RF2-XGB2, respectively.

*(7) In Figure 2 caption, please explain NR, GPCR, IC, and E are four different benchmarking datasets, same for figure 3.*

**Reply:** Thank you for your effective advice. We have explained all abbreviations in the caption. And

we removed Figure 3 in the manuscript due to the remove of FP2 and PsePSSM, the reason shown in Comment 1.

*(8) Please provide a detailed example of how to use the method at the GitHub page.*

**Reply:** All script and instructions with tutorial have been uploaded to GitHub page.

We appreciate for editor's and reviewers' warm work earnestly, which are valuable in improving the quality of our manuscript. We hope that the corrections will meet with approval. Once again, thank you very much for your comments and suggestions.