Random Forest refinement of pairwise potentials for protein-ligand decoy detection

Jun Pei,[§] *Zheng Zheng,*[§‡] *Hyunji Kim,*[§a] *Lin Frank Song,*[§] *Sarah Walworth,*[§b] *Margaux R. Merz,*[§c] *and Kenneth M. Merz Jr.**[§†]

[§]Department of Chemistry, Michigan State University, 578 S. Shaw Lane, East Lansing, Michigan 48824, United States

[†]Institute for Cyber Enabled Research, Michigan State University, 567 Wilson Road, East Lansing, Michigan 48824, United States

Abstract

An accurate scoring function is expected to correctly select the most stable structure from a set of pose candidates. One can hypothesize that a scoring function's ability to identify the most stable structure might be improved by emphasizing the most relevant atom pairwise interactions. However, it is hard to evaluate the relevant importance for each atom pair using traditional means. With the introduction of machine learning methods, it has become possible to determine the relative importance for each atom pair present in a scoring function. In this work, we use the Random Forest (RF) method to refine a pair potential developed by our laboratory (GARF[6]) by identifying relevant atom pairs that optimize the performance of the potential on our given task. Our goal is to construct a machine learning (ML) model that can accurately differentiate the native ligand binding pose from candidate poses using a potential refined by RF optimization. We successfully constructed RF models on an unbalanced data set with the 'comparison' concept and, the resultant RF models were tested on CASF-2013.[5] In a comparison of the performance of our RF models against 29 scoring functions, we found our models outperformed the other scoring functions in predicting the native pose. In addition, we used two artificial designed potential models to address the importance of the GARF potential in the RF models: (1) a scrambled probability function set, which was obtained by mixing up atom pairs and probability functions in GARF, and (2) a uniform probability function set, which share the same peak positions with GARF but have fixed peak heights. The results of accuracy comparison from RF models based on the scrambled, uniform, and original GARF potential clearly showed that the peak positions in the GARF potential are important while the well depths are not.

Introduction

Molecular docking is a widely used method in structure based drug design (SBDD), and scoring functions are essential components in molecular docking programs. Some well-known scoring functions are used in the GOLD,[1] SurFlex Dock,[2] Glide,[3] and AutoDock Vina docking packages.[4] A highly performant scoring function should have several properties:[8] (1) the most stable ligand binding pose should have the lowest rank, (2) it should distinguish between ligands that bind from nonbinding ones, (3) the scores generated from the scoring function are correlated with the experimentally determined binding affinities. Extant scoring functions can be classified into four broad categories: force field based,[12-22] knowledge based,[23-34] empirical,[9, 35-40] and machine learning based[8, 10, 11, 41-49] scoring functions. Force field based scoring functions typically employ a classical force field, which use relatively simple energy equations to describe bond, angle, torsion, and nonbond interactions in a protein ligand complex, to represent a three dimensional structure at the molecular level. Alternatively, knowledge-based scoring functions employ a statistical analysis on the radial distribution functions of atom pairs, which are extracted from a protein-ligand structure database, to obtain "pure" interactions between atom pairs. Empirical scoring functions assume the binding affinity between a protein and ligand can be decomposed into basic components with different coefficients, which can be obtained through multivariate regression analysis on a set of protein ligand complexes with experimentally determined structures and binding affinities.[9] Machine learning based scoring functions employ a variety of machine learning/deep learning methods along with a variety of information from protein ligand systems to predict the binding poses and affinities for a given protein ligand systems.

With the ever-increasing amounts of data, force field based scoring functions are limited by their relatively high computing costs, while the accuracy of these methods is continually challenged by the increasingly diverse data sets that require specific parameterizations for high accuracy. On the other hand, recently reported scoring functions based on machine learning techniques showed promising performance using multiple different input types, such as topology representations[10] and three-dimensional "pictures"[11] of protein ligand complexes. Hence, it is possible that the performance of traditional scoring functions might be improved by combining the information encoded within them with novel machine learning models. From a chemistry point of view, a lot of relevant information (for example, potential functions) are contained in force field and knowledge based scoring functions which might prove useful in building machine learning models. Using traditional methods, it is almost impossible to assign different importance factors to each atom pair in a potential database such that the most important atom pairs would be emphasized in a calculation. However, this can be accomplished using machine learning algorithms.

In this work, we focus on using Random Forest(RF) models to assign different importance factors to each atom pair in order to improve the scoring function's ability to rank the most stable binding pose as the lowest. The GARF potential data base, which is generated using a graphical-model-based approach with Bayesian field theory,[6] was used as an example of pair wise potential data base. We successfully constructed RF models on unbalanced data sets with the 'comparison' concept, and RF models were tested on a well-known protein ligand decoy set, CASF-2013,[5] to evaluate their ability to select the most stable structure. The accuracy comparison results between our RF models and another 29 scoring functions suggest that the RF models have a greater ability to correctly identify the native ligand binding pose among a

collection of decoy poses. In addition, based on the GARF potential data base, we constructed two artificial probability function sets to address the importance of the potential data base used to generate the RF models. In particular, a scrambled probability function set was used to test if the GARF potential is critical in building RF models, and a uniform probability function set was used to further understand the most important information contained in GARF. The accuracy comparison results showed that the peak positions in the GARF potential data base is critical to build an accurate RF model. In the end, the influence of training set size was also tested. The results showed that accuracy converged after the training set's size is larger than 60 % of the whole data set, which provides a strong evidence for the fact that only peak positions are the most important information in RF models.

Method

Descriptor for intermolecular interactions

In a n-body system, if all independent pair wise probabilities are known, the overall probability of the whole n-body system can be calculated as follows:

$$p_n = \prod_{i,j=1,\, i\neq j}^{n} c_{ij} \times p_{ij}, \quad (1)$$

where p_{ij} represents the independent probability for particle pair i and j, c_{ij} is the corresponding coefficient for p_{ij}. If a protein ligand complex is considered as a n-body system, with the

knowledge of all independent pair wise probabilities (including the pair wise probabilities of bond, angle, torsion, and nonbonded interactions), the overall probability of a protein-ligand complex can be obtained as:

$$p_{protein-ligand} = p_{protein} \times p_{ligand} \times p_{intermolecular\ of\ protein-ligand} \quad (2)$$

where $p_{protein-ligand}$ represents the overall probability of a protein-ligand complex, $p_{protein}$ and p_{ligand} are the probability of protein and ligand, respectively. $p_{intermolecular\ of\ protein-ligand}$ is the probability of the intermolecular interactions between the protein and ligand. If the protein is treated as a rigid body, $p_{protein}$ will be the same for both native and decoy protein-ligand complexes, hence, it is a constant in equation (2). On the other hand, p_{ligand} can be expanded as:

$$p_{ligand} = (\prod_{bond} c_{ij} \times p_{ij})(\prod_{angle} c_{kl} \times p_{kl})(\prod_{torsion} c_{mn} \times p_{mn})(\prod_{nonbond} c_{pq} \times p_{pq}) \quad (3)$$

$c_{\alpha\beta}$ and $p_{\alpha\beta}$ represent the scaling factor and the probability of atom pair α and β, the subscripts *ij*, *kl*, *mn*, and *pq* correspond to bond, angle, torsion, and non-bonded atom pairs, respectively. In theory, $p_{\alpha\beta}$ could be found in a potential database, however, given the scarcity of data, the GARF potential does not contain the pair-wise probabilities for atom pairs that only exist in a ligand. Hence, in this work, p_{ligand} is a constant in equation (2), in other words, the ligand molecule is also treated as a rigid body.

With the product of $p_{protein}$ and p_{ligand} as a constant *C*, equation (2) can be rewritten as:

$$p_{protein-ligand} = C \times p_{intermolecular\ of\ protein-ligand} = C \times (\prod_{intermolecular} c_{st} \times p_{st}) \quad (4)$$

c_{st} and p_{st} are the weighting coefficient and the probability of intermolecular atom pair *s* and *t*. Taking logarithm on both sides of equation (4) we get:

$$\log(p_{protein-ligand}) = log(C) + \sum_{inter-molecular} log(c_{st} \times p_{st}) \quad (5)$$

the p_{st} terms are obtained from the GARF potential function database, which contains all intermolecular potential functions between the protein and ligand. c_{st} is a to-be-determined parameter obtained using machine learning.

Fig. 1 shows, as an example, the probability function for N and C.3 from the GARF database. The *x-axis* in **Fig. 1** represents the distance between N and C.3 and the *y-axis* represents the probabilities corresponding to the various distances. The probabilities at different distances can be calculated directly from the probability function. In this work, instead of using one probability for a given distance, we used an averaged probability over a small region centered on the selected distance. For example, if the atom pair N_C.3 is found in a protein-ligand complex at a distance of 4.0 Å, 201 different probabilities will be calculated for the distances between 3.5 and 4.5 Å with an interval of 0.005 Å. Then, the logarithm value of the averaged probability (average value for 201 probabilities) is calculated to represent the probability of atom pair N_C.3 at a distance of 4.0 Å. In general, the calculation can be summarized as follows:

$$p_{A-B}(r_1) = log\left[\sum_{r_1-0.5}^{r_1+0.5} GARF_{A-B}(r_{AB})\right] - log(201) \quad (6)$$

$GARF_{A\text{-}B}$ is the probability function from the GARF database and r_1 is the distance between atom A and B in a protein-ligand system (in above example, r_1 equals 4.0 Å), while r_{AB} represents a distance in the range of $r_1 \pm 0.5$ Å over an interval of 0.005 Å.

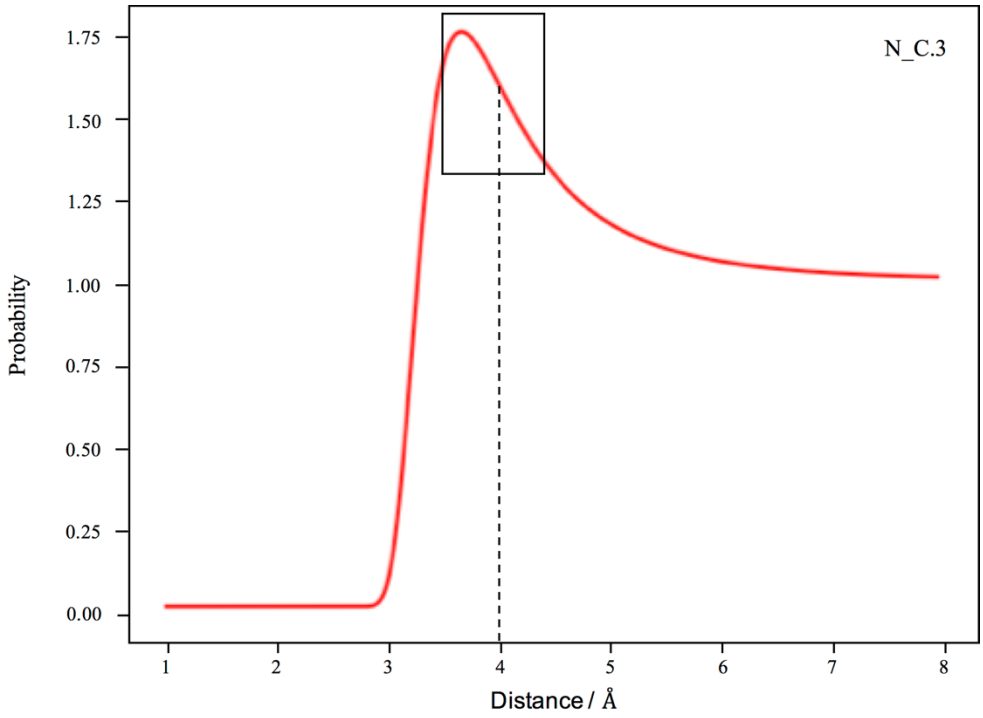

Fig. 1 Probability function of atom pair N and C.3 in GARF. x-axis is the distance between N and C.3 and the y-axis is probability. The selected region shows a sampling range of 4.0 ± 0.5 Å.

A given atom pair may appear multiple times in a given protein-ligand complex, so we sum over all atom pairs of the same type to obtain an overall probability. Thus, the sum of all

individual probabilities for a given atom pair is used as the final probability for that specific atom pair in the protein-ligand complex. Therefore, the final probability can be expressed as:

$$p_{A-B} = \sum_{i=1}^{k} p_{A-B}(r_i) \quad (7)$$

where, p_{A-B} is the final probability of atom pair AB. Indices of r_1 to r_k are the distances between atom A and B found in a protein ligand complex structure, r_i represents a distance from r_1 to r_k. Based on this formulation, a three-dimensional protein-ligand structure can be converted to a one dimensional column vector of pairwise probabilities. **Fig. 2** shows the protocol of converting the protein ligand structure into a column vector with atom pair wise probabilities.

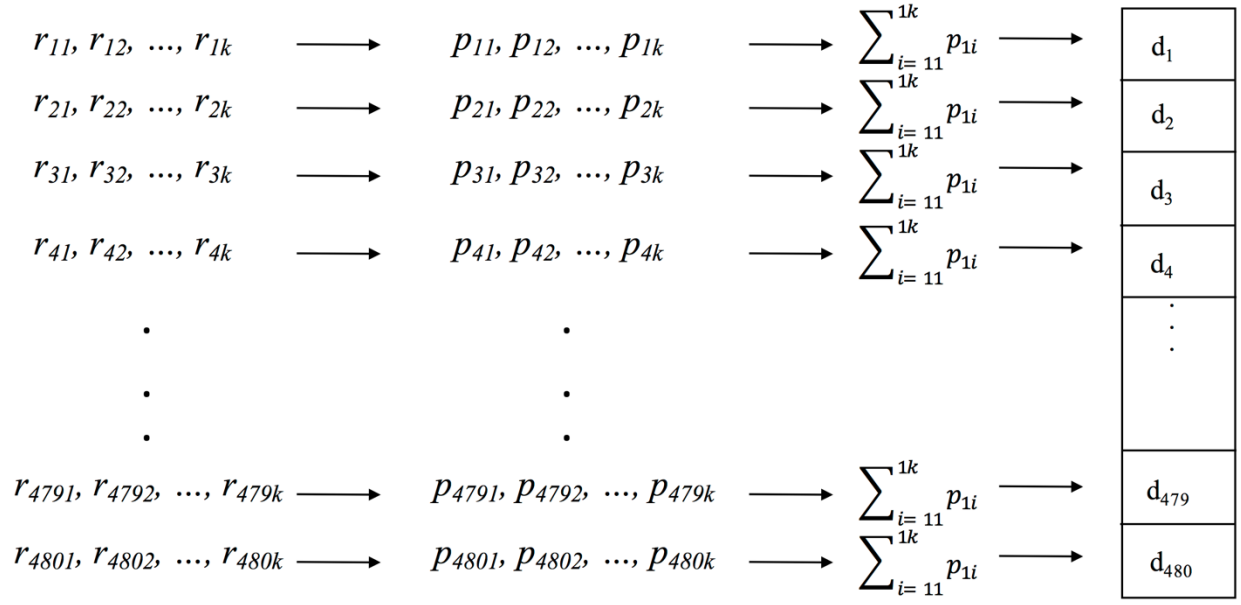

Fig. 2 A general protocol to generate the column vector for a protein ligand complex. The subscript *1k, 2k, …, 480k* represents the occurrence number of each atom pair found in the structure.

Random Forest model

Supervised learning models can be split into two broad categories; regression and classification models.[50] In this work, the goal is to construct a classifier that can accurately determine the native pose from decoy poses. The most straightforward way to do this is to create a model that predicts whether a ligand pose is native or not. However, the low number of native poses eliminates the possibility of constructing an efficient classifier (in one decoy set, there is only one native pose amongst hundreds of decoys). Therefore, it is necessary to find out a way to make the number of samples in each class similar. In this work, a 'comparison' concept was used to transfer a decoy set into two classes with the same number of members.

The one criteria for our data sets in the protein ligand complex of the native pose is always more stable than the decoy poses. In other words, the complex of the native binding pose has the higher probability than all the others. Accordingly, the probability of the native pose minus the probabilities of any decoy pose should be always greater than zero, and the reverse will be less than zero. In this way, if all the results larger than zero are labeled as "class 1" and the reverse are defined as "class 0", those two classes will have same number of members, which is ideal for constructing a classifier model.

Fig. 3 shows the general protocol for generating a RF model. As an example we took one protein ligand decoy set, which contains one ligand native pose and *m* ligand decoy poses. First, the GARF potential database was used to convert all 3 dimensional protein ligand structures into

column vectors following the steps described in **Fig. 2**. These column vectors are termed descriptor vectors. Then, the results of the descriptor vector from the native structure minus the corresponding vectors from every decoy are called “class 1”, which corresponds to “more stable than”. Similarly, the inverse are labeled “class 0” which it represents “less stable than”. Consequently, class 1 and class 0 have the same number of members, which is perfect to build up a classification model. In this work, the RF algorithm was selected to build up the classifier. The goal of the classifier is to accurately compare any two random structures in a given set. For instance, in order to compare two protein ligand complexes, which share the same protein and ligand structures, the descriptor vectors are calculated for each protein ligand complex. A final descriptor vector is generated using the descriptor vector of the first complex minus the corresponding one from second structure. Then, the final descriptor vector was treated as an input for the RF model to give its final prediction. If the predicted result is class 1, that means the first complex is more stable than the second one; on the other hand, if the prediction is class 0, that indicates the first complex is less stable than the second one.

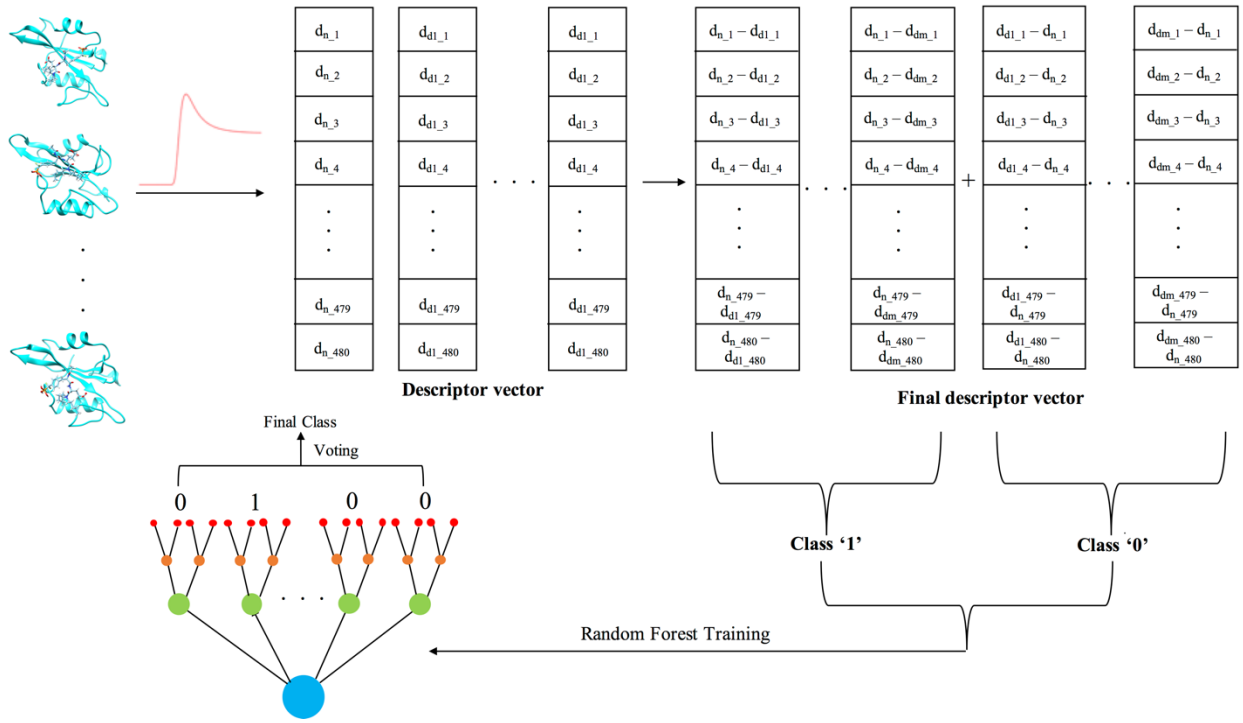

Fig. 3 A general protocol to construct a random forest classifier on an unbalanced decoy set.

It is possible to compare two complexes with the RF model described above, however, there is still a gap between the RF model and blind tests. The goal of a blind test is to identify the complex with ligand native pose among a large number of complexes with ligand decoy poses. Hence, another protocol needs to be introduced to do a blind test based on the RF model and **Fig. 4** outlines the workflow. Here, one decoy set containing *n* protein-ligand complexes is shown as an example. The goal is to identify the complex with the ligand in its native pose. There are four steps to identify the selected complex: (1) all *n* protein-ligand complexes were converted to *n* descriptor vectors using the procedure described previously. (2) Compare each complex structure with all the other complexes. As an example we generate the comparison result for the first complex S_1 with all other structures. Table_S1 shown in **Fig. 4** is obtained by using the descriptor vector of the first complex minus the descriptor vectors of the other complexes. Each

column in that table represent a comparison between the first complex with one of the other structures, hence, Table_S1 has *n-1* columns, which contains the information of all comparisons between the first complex and all other complexes in the decoy set. In total, there are *n* comparison tables for each complex in the decoy set. (3) Tables obtained in the previous step were used as inputs for our RF model resulting in a row vector representing the comparison result for each table. Every element in the resultant row vector represents the comparison result when a specific complex was compared with another structure. "1" means "more stable than" and "0" is "less stable than". Next, the sum of the row vector is defined as the "Score", and the score of complex S_1 is called "Score_S1". (4) Based on these "Score" values, a rank of all complexes can be obtained. In this way the most stable complex can be identified from a collection of protein-ligand complexes with RF models.

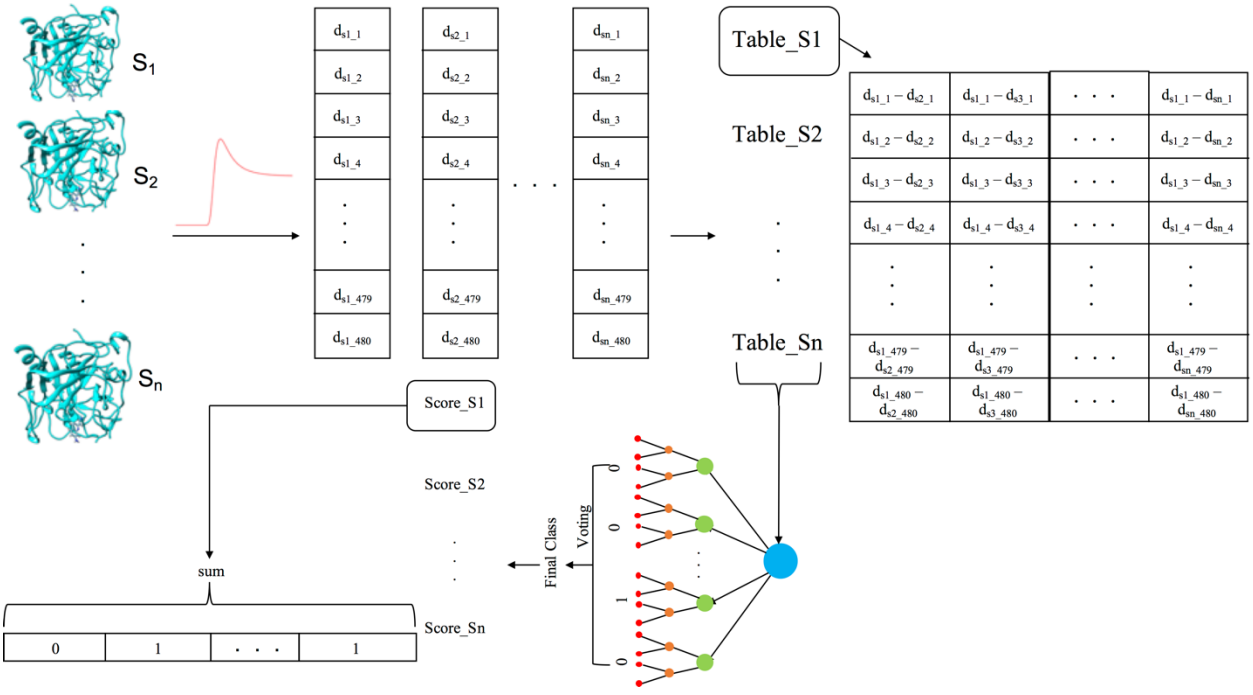

Fig. 4 A general protocol of calculating scores for each protein ligand complex in a blind test using a RF model.

Random Forest model with decoy comparison information

Our RF model is focused on identifying the native binding pose of a ligand among all decoy poses. However, it is not effective in identifying the best decoy due to the lack of comparison information between the best decoy structure and the other decoy poses. In order to include the comparison information between decoy poses into the RF analysis we made the following assumption. The assumption is that the ligand decoy pose with the lowest RMSD is the most stable decoy structure (best decoy pose). **Fig. 5** shows the protocol of adding comparisons between the best decoy pose and other decoy poses. For example, a decoy set contains *m* decoy structures and one native pose, two kinds of comparisons were considered when the model was trained: (1) the comparison between the native binding pose and all other decoy poses, in total there are 2*m* comparisons (*m* comparisons for each class); (2) without the native binding pose, the best decoy pose was compared with all other decoy poses for a total of 2(*m* – 1) comparisons. Then, RF models, which were trained on these comparisons, were used to select the best decoy through the protocol of **Fig. 4**.

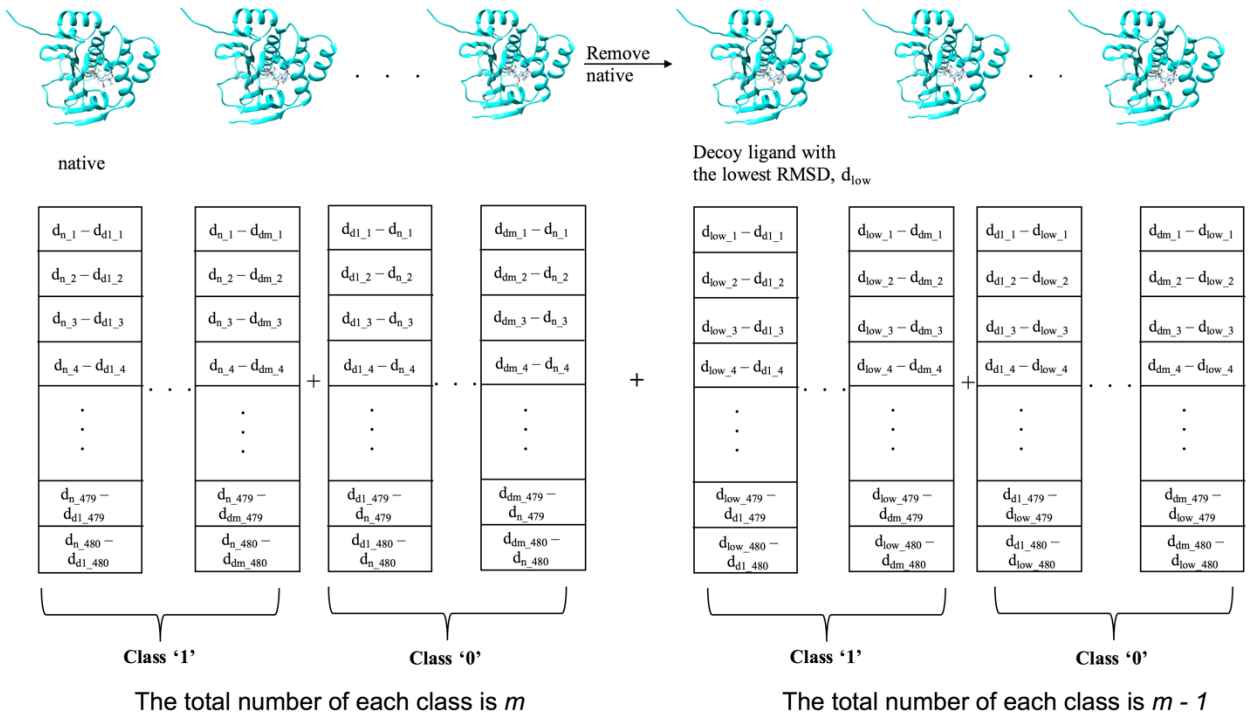

Fig. 5 The protocol used to include the comparison information between best decoy binding pose and other decoy poses.

Decoy set

In this work, 191 systems were selected out of the 195 systems in CASF-2013[5] due to formatting issues with our program. CASF-2013[5] is known as the 'Comparative Assessment of Scoring Functions', it includes data sets for testing the scoring, docking, screening, and ranking powers of scoring functions. Here, we only used the data sets, which were designed to test the docking power of scoring functions. The decoy ligand binding poses were prepared with three popular molecular docking programs: GOLD(v5), Surflex-Dock implemented in SYBYL(v8.1), and the docking module built in MOE(v2011). These three programs have different algorithms for ligand pose sampling, therefore, the resultant decoy set is more complete and avoids the bias

inherent in using only one program. In total, we used 191 protein ligand systems, 15802 ligand decoy poses, and 31604 native-decoy comparisons.

GARF potential

Herein, we used the GARF[6] potential to calculate the pairwise probabilities for each protein ligand complex. GARF is a potential database developed by our group. It employed a graphical-model-based approach with Bayesian field theory to construct atom pairwise potential functions. There are 20 atom types for the protein atoms and 24 atom types for the ligands. All definitions of the atom types are listed in **Table s3** in the supporting information. Further details regarding GARF can be found in the original article.[6]

Machine learning and validation

The sklearn.ensemble.RandomForestClassifier function from Scikit-learn was used to create the proposed classification model.[7] One training-testing iteration includes: (1) Randomly split the whole data set into two parts, 80% as the training data set and 20% as the test set. (2) A grid search with five-fold cross validation was done on the training set in order to identify the best set of hyperparameters for the RF model. (3) The RF model with the best set of hyperparameters was then validated on the test set. Ten independent iterations were performed on the CASF-2013[5] decoy set in order to avoid bias from our data partitioning scheme.

In order to evaluate the performance of the RF models, a concept called 'accuracy' was used in this work. In supervised machine learning, usually a 'confusion matrix' is applied to evaluate the performance of a classifier. The format of a confusion matrix is presented in **Table 1**:

Table 1. General form of a confusion matrix

	Predicted (class 1)	**Predicted (class 0)**
Actual (class 1)	TP	FN
Actual (class 0)	FP	TN

There are four values in the confusion matrix, which are TP, FP, FN, and TN. TPs (True Positives) refers to the cases whose predicted classes are 'class 1' - same as their actual classes. FPs (False Positives) are the cases with a predicted class of 'class 1' even though their actual class is 'class 0'. FNs (False Negatives) represent cases whose predicted class is 'class 0', however, their actual class is 'class 1'. Finally, TNs (True Negatives) refers to the case where the predicted class is 'class 0' which is the same as their actual class. Accuracy can be calculated based on these four numbers from the confusion matrix using:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{8}$$

Usually, accuracy is not an ideal judge of the performance of a classifier because the difference between two actual classes might be large. For example, if class 1 has 90 examples and class 0 only has 10 examples, a "naïve" classifier, which predicts every sample as class 1, can achieve an accuracy of 90%. On the other hand, in this work, our two classes have equal members,

hence, accuracy can be used to judge the performance of the proposed RF model. Ten independent iterations were done on the CASF-2013[5] decoy data set yielding ten accuracies to average over. Here, we used the averaged, highest, and lowest accuracies to represent the general performance of our RF models.

Potential analysis

In order to test the importance of using the GARF potential database, two sets of artificial probability functions were constructed.

(1) Scrambled probability function set

A scrambled probability function database was set up based on GARF in order to test whether the GARF database is critical to the performance of the RF models. In the GARF database, each atom pair has its probability function. The scrambling process mixes up the atom pair names. For example, before scrambling, one probability function might represent a hydrogen bond interaction, and after mixing up the atom pair names, the same probability function might be used to describe the interaction between two carbon atoms. Ten independent RF models were built using the scrambled data base. The averaged, highest, and lowest accuracies were selected to represent the general performance of RF on the scrambled data set.

(2) Uniform probability function set

A uniform probability function set was constructed based on the GARF database in order to address the most important information buried in the data set. The uniform probability function only contains the peak positions for each atom pair found in GARF. There are two steps to artificially construct a uniform probability function set. First, peak positions $r_{peak\_A\_B}$ (A and B represent two atom names in GARF) of each atom pair were collected from the GARF data set. Then, probability functions were designed using:

$$p_{AB} = e^{\frac{(E1*\left(\frac{3}{r_{AB}}\right)^{12} - E2*\left(\frac{3}{r_{AB}}\right)^{6})}{-RT}} \quad (9)$$

In equation (9), p_{AB} and r_{ABis} are the probability function and distance of atoms *A* and *B*, E_1 and E_2 are two to-be-determined variables. If the peak height in the uniform probability function set is fixed, two equations are created to solve for the values of E_1 and E_2.

$$\frac{(E1*3^{12}*\frac{-12}{r_{peak\_A\_B}{}^{13}} - E2*3^{6}*\frac{-6}{r_{peak\_A\_B}{}^{7}})}{-RT} = 0 \quad (10)$$

$$e^{\frac{(E1*\left(\frac{3}{r_{\mathrm{peak\_A\_B}}}\right)^{12} - E2*\left(\frac{3}{r_{\mathrm{peak\_A\_B}}}\right)^{6})}{-RT}} = Constant \quad (11)$$

Equation (10) represents the maximum of the probability function at the peak position $r_{peak\_A\_B}$ and, equation (11) shows that the uniform probability function shares the same peak height. The *Constant* in equation (11) can be set to any positive value, in this work, it was set to 2. E_1 and E_2 can be obtained by solving equations (10) and (11), and in this

way the uniform probability functions for each atom pair can be determined, based solely on the peak positions.

Result and Discussion

Accuracy

The most important goal of a scoring function is to accurately identify the native structure among an plethora of decoy structures. In order to evaluate the ability of a scoring function to identify the native structure, the concept of accuracy is used in this work. If a decoy set contains 100 decoy structures but only one native, the scoring function is expected to make 200 correct comparisons to identify the native pose. The higher the accuracy of the comparison, the better the performance of the scoring function. The third column in **Table. 2** shows the comparison of accuracies from RF models and 29 other scoring functions. The averaged, highest, and lowest accuracy of the RF models are 0.953, 0.969, 0.942. The averaged accuracy value is higher than all of the other tested scoring function, and the lowest accuracy value is still higher than all of the other accuracies. It is clear that the RF models have a higher accuracy, which means that the RF models perform better than all other scoring functions in comparing the ligand native pose to decoy poses.

Table. 2 Comparisons between RF models and 29 other scoring functions.

		accuracy	Native's ranking	1st decoy RMSD

RF models	Averaged	0.953	4.49	3.87
	Highest	0.969	5.54	4.47
	Lowest	0.942	3.54	3.38
Conventional SFs	GOLD-ASP	0.924	6.13	1.74
	GOLD-ChemPLP	0.923	6.25	1.51
	DS-PLP1	0.917	6.68	1.80
	DS-PLP2	0.914	7.07	1.87
	MOE-Affinity_dG	0.900	8.23	2.42
	Xscore-HMScore	0.891	8.89	2.45
	Xscore-Average	0.886	9.33	2.38
	GOLD-ChemScore	0.882	9.58	1.72
	DS-PMF04	0.874	10.58	3.38
	SYBYL-PMF	0.874	10.53	3.42
	Xscore-HPScore	0.871	10.63	2.75
	MOE-Alpha	0.870	10.38	1.85
	Xscore-HSScore	0.869	10.80	2.64
	DS-LigScore2	0.867	10.67	1.83
	MOE-London_dG	0.863	11.38	2.52
	DS-PMF	0.857	11.89	3.48
	MOE-ASE	0.856	11.88	2.91
	GlideScore-SP	0.832	13.20	1.72
	DS-LigScore1	0.823	14.28	2.31
	GlideScore-XP	0.823	14.07	1.86
	GOLD-GoldScore	0.819	14.70	1.88
	DS-LUDI2	0.807	15.62	2.23
	DS-LUDI1	0.799	16.35	2.34
	DS-LUDI3	0.783	17.48	2.87

	SYBYL-ChemScore	0.782	17.70	2.40
	SYBYL-Gscore	0.725	22.64	3.13
	dSAS	0.692	25.48	3.96
	DS-Jain	0.685	25.63	2.90
	SYBYL-Dscore	0.674	26.70	4.03

Native ranking

Other than accuracy, another criteria for evaluating a scoring function is the ranking of the ligand native pose. In other words, a scoring function is expected to give the ligand native pose the lowest rank. The fourth column in **Table. 2** shows the result of ligand native pose ranking from each method. The averaged, highest, and lowest ligand native pose ranking from RF models are 4.49, 5.54, and 3.54, respectively. The confidence interval of the native pose's ranking from RF models is [3.54, 5.54]. It is clear that all 29 scoring functions have ligand native rankings higher than the averaged native ranking obtained from the RF models, and of these rankings they are larger than the highest native ranking from the RF models. Thus, it can be concluded that the RF models perform better in selecting the ligand native pose than existing models.

If the accuracy values are compared with the ligand native pose rankings, a correlation between those two sets of data can be found. The higher the accuracy, the lower the native pose ranking. The most important goal of a scoring function is to identify the most stable ligand pose (native pose), therefore, the minimum standard for a scoring function is to correctly compare

native pose to decoy ones. Using our previous example of a decoy set containing 100 decoy structures and one native pose we have 200 comparisons between the native and decoy poses. Hence, minimally the scoring function should make 200 correct comparisons to obtain the native structure. With more correct comparisons, the native pose has a higher chance to be found at a lower rank. For example, if the scoring function makes ten mistakes, the accuracy is around 0.95, and the ligand pose ranking would be ≥ 5.

RMSD of the best decoy

Besides accuracy and native pose ranking, there is another criteria, RMSD of the best decoy structure, which is used to judge the performance of a scoring function. The best ligand decoy pose refers to the decoy structure that is selected by a scoring function as the structure among all decoy poses most similar to the native pose. Scores generated by a scoring function are expected to be correlated with the quality or native-likeness of a structure. The RMSD value between the ligand native binding pose and a decoy binding pose is often used to represent the quality of that decoy pose. If the RMSD is below a predefined cutoff (RMSD < 2 Å), the decoy binding pose is believed to be "native-like". The last column in **Table. 2** shows the RMSD values from each of the scoring functions. The averaged, highest, and lowest RMSD values from RF models are 3.87 Å, 4.47 Å, and 3.38 Å, respectively. The confidence interval for the RF models is [3.38, 4.47]. It is clear that there are 26 scoring functions that can identify ligand decoy poses with RMSDs lower than 3.38 Å, and two scoring functions provide RMSD values within the confidence range of the RF models. In general, 28 scoring functions perform better than our initial RF models in selecting the best decoy structure.

The RF models used in **Table. 2** only contain comparisons between native and decoy poses, while comparison information between decoy poses was not considered when the models were trained. Hence, we conclude, that these RF models do not have enough information to find the "best" ligand decoy poses among a large number of decoy structures. In order to improve our RF models' ability to identify the best decoy structure, comparison information between decoy poses should be added when training the RF models. Here, we make an assumption that, among all decoy poses, the pose with the lowest RMSD is perhaps the most stable of all the decoys because it is most "native-like". With this assumption, the comparison between the best decoy and other decoy poses could be generated. Instead of just using comparisons between native and all decoy poses, the new training set also included comparisons between the best decoy pose and all other decoy poses. **Table. 3** shows the result when different number of decoy structures were identified as the most stable poses. Four sets of training data were used: (1) only including comparisons between the native and decoy poses; (2) including comparisons between the native and decoy poses, and between the decoy structure with the lowest RMSD with all other decoy poses; (3) including comparisons between the native and decoy poses, between the two lowest RMSD decoy poses and all other decoy poses; (4) including comparisons between the native and decoy poses and, between the three lowest RMSD decoy poses with all other decoy poses. **Table. 3** gives the overall performance on accuracy, ligand native pose ranking, and the best decoy RMSD. With the inclusion of decoy structures in the training set, the accuracy of the RF models and the ligand native binding pose's ranking were slightly negatively affected. On the other hand, the best decoy pose's RMSD dropped dramatically. The averaged, highest, and lowest RMSD of the best decoy pose from RF models trained on data set only including

comparisons between native and decoy binding poses are 3.87 Å, 4.47 Å, and 3.38 Å, respectively. Alternatively, the corresponding values from RF models including the three lowest RMSD decoy structures are 2.27 Å, 2.44 Å, and 1.73 Å, respectively (confidence interval is [1.73, 2.44]). By including low RMSD decoy structure comparisons we obtain RF models (see **Table. 2**) that give better first decoy RMSDs than 13 scoring functions, a further 15 scoring functions have first decoy RMSDs with the confidence interval of the RF model and only one scoring function gave a RMSD smaller than 1.73 Å. Hence, we conclude that the overall performance (*i.e.*, accuracy, native rank, and low RMSD first decoy) of RF models can be improved by including lowest RMSD decoy comparisons in the fitting of the model.

Table. 3 Comparison between RF models with considering different number of decoy pose in training set.

		Accuracy	Native's ranking	1st decoy RMSD
With no decoy structure	Averaged	0.953	4.49	3.87
	Highest	0.969	5.54	4.47
	Lowest	0.942	3.54	3.38
With one lowest RMSD decoy structure	Averaged	0.958	4.28	2.41
	Highest	0.974	7.49	2.72
	Lowest	0.921	3.03	2.13
With two lowest RMSD decoy structure	Averaged	0.950	5.03	2.50
	Highest	0.957	6.08	2.99
	Lowest	0.937	4.39	1.95
With three lowest RMSD decoy structure	Averaged	0.947	5.21	2.27
	Highest	0.963	6.56	2.44
	Lowest	0.930	3.97	1.73

Based on previous discussion, it is clear that with a higher accuracy, a scoring function can give the native binding pose a lower rank. If the accuracy values are compared with the RMSD of the best decoy, it is obvious that those two sets of data do not appear to correlate. Some scoring functions are better at selecting the native pose but provide a relatively larger RMSD value, whereas other scoring functions do a better job selecting the best decoy structure but do not have the ability to identify the native binding pose. This leads to a basic philosophical question: which one is more important, accuracy or RMSD? Both of them should be important in the limit that all decoy poses can be obtained. However, it is almost impossible to generate all relevant decoy poses using contemporary approaches. In our opinion, the basic requirement for a scoring function is that the function can accurately identify the native pose. To some degree, RMSD might be useful in judging if a structure has a low free energy, but it is obvious that a decoy structure can have a high free energy while enjoying a low RMSD value. Hence, if two scoring functions were compared solely on identifying the best decoy and one gives a RMSD larger than 2 Å while another is less than 2 Å, it is unclear, at least to us, how to judge which one is better. On the other hand, accuracy, the factor that represents the performance of a scoring function when comparing native and decoy poses, is a clear standard. The explicit hypothesis we are making when docking and scoring is that the native structure always has a lower free energy than the decoys. When comparing two scoring functions, the better scoring function should be the one with a higher accuracy. Put another way, when creating, for example, ML models for a self-driving car what is more important – accurately identifying an obstacle or being close to identifying an obstacle? Therefore, we believe that accuracy is the more important criteria.

Uniform probability function

The RF models perform better than all other scoring function on accuracy and native binding pose ranking. It is interesting to consider if the GARF potential is critical in these RF models. Two tests were set up in order to test the importance of the GARF potential database. First, a scrambled probability function set was constructed based on GARF followed, by a uniform probability function set to test whether GARF's peak position is more important or if the peak height is more critical.

The scrambled probability function set was generated by randomly mixing up the atom pairs in the GARF potential database. Taking the 480 atom pairwise potential functions in GARF we randomly scrambled the atom pair names. For example, before scrambling, one probability function represented the interaction between N and O.co2, while after scrambling, the same probability function might be used to describe the interaction between C and F. Hence, the scrambled probability function set is physically unrealistic. Based on the scrambled probability function set, ten independent RF models were constructed following the same procedure described in the methods section. Since the scrambled function set is physically unrealistic, it is expected that the performance of these RF models would be worse than models using the original GARF potential.

There are two kinds of information embedded in the GARF potential, peak positions (well position) and peak heights (well depth). Which is more important – or are both important? To address this a uniform probability function set was built up to probe this fundamental

question. Uniform probability functions share the same peak positions with the original GARF potential, but set all the peak heights at a constant value eliminating the impact of prior peak heights. If the obtained RF models based on a uniform probability function set performs similarly to models obtained with the original GARF, peak positions will be more important than peak height. Alternatively, if the obtained RF models perform more poorly than original the models peak height is significant.

Table. 4 compares the accuracy result from RF models based on the original, scrambled, and uniform GARF potential database. If we compare the accuracy values between RF models based on original and scrambled GARF, it is clear that the averaged, highest, and lowest accuracies from RF models with a scrambled probability function perform poorer. The accuracy value did not drop as much as we have seen in the past[51] because the GARF potential only contains intermolecular interactions found in protein ligand systems. Moreover, the 480 peak positions found in GARF are all in the range of [2.5, 5.1] with 355 peak positions in the range of [3.4, 4.4] (see **Table. s1**). Therefore, the scrambled peak positions in the scrambled probability function set might be similar to the original positions in GARF. It is reasonable to expect that the accuracy of RF model based on scrambled probability function set is lower than the corresponding values from original models. On the other hand, if we compare the accuracy values from the uniform probability function set to the values provided by the original RF models, it is obvious that the averaged accuracy values from those two sets of models are the same. This further supports the notion that peak position is more important than well depths in given a potential function used to build a RF model.[51]

Table. 4 Comparison between RF models with different probability function sets

	RF models		
	Averaged accuracy	Highest accuracy	Lowest accuracy
Original GARF	0.953	0.969	0.942
Scrambled GARF	0.933	0.951	0.911
Uniform GARF	0.953	0.980	0.918

Influence of training set size

Usually in the field of supervised machine learning, especially when the data set does not contain a large number of data points, it is common to split the data set into training (80% of total, 16% cross validation set, five-fold cross validation in training data) and test sets (20% of total). The 80:20 ratio works well in most cases, but we wanted to test whether the RF models can achieve a similar accuracy with a smaller training set. **Table. s2** shows the accuracy result from RF models based on the original and uniform GARF data base trained on data sets of differing sizes. **Fig. 6** is the corresponding plot obtained using the data of **Table. s2**. The blue and orange lines in **Fig. 6** represent the performance of RF models based on the original and uniform GARF database, respectively. Both lines show that by increasing the size of the training set, the accuracy of RF models generally increased. Accuracy values converge with training sets >60% and the RF models based on the original and uniform GARF potential have the same trend.

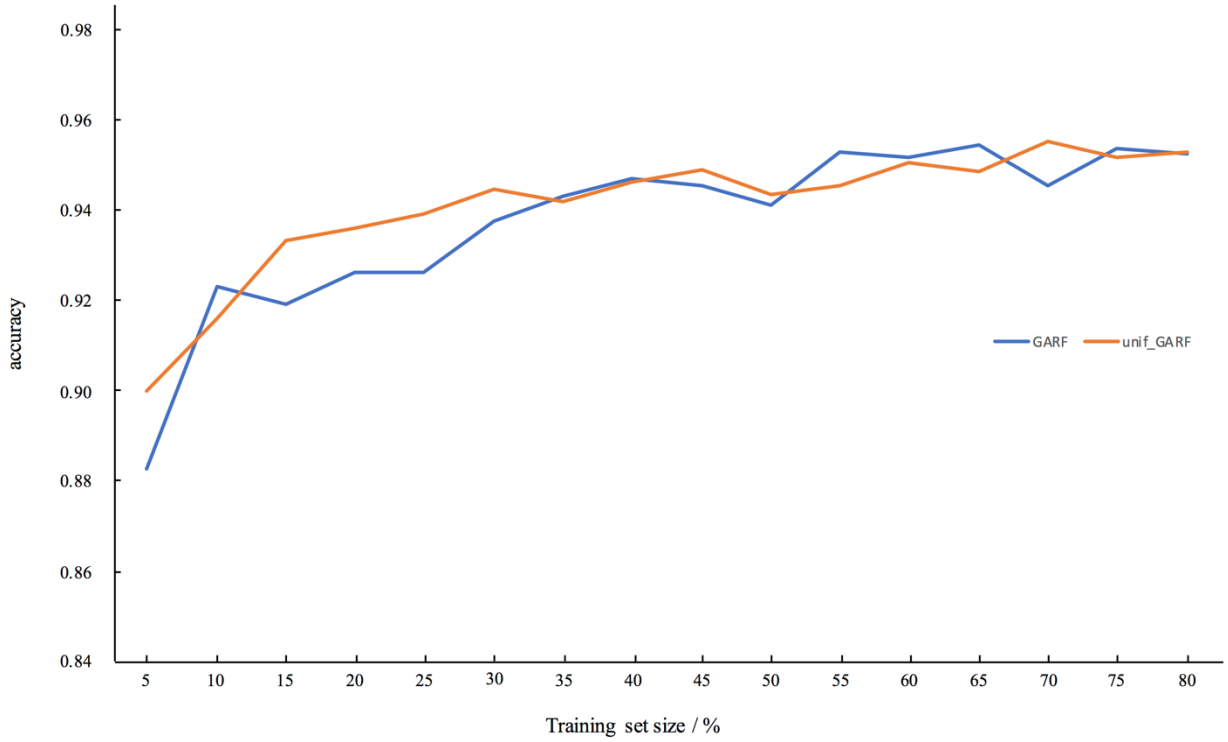

Fig. 6 Accuracy trend from RF models based on original(blue line) and uniform(orange line) GARF data sets.

Conclusions

In this work, we constructed RF models on unbalanced data sets utilizing the 'comparison' concept to identify native protein-ligand poses. Using RF, the GARF potential database was refined by assigning different importance factors to each atom pair in that potential. The resultant RF models were tested on a well-known protein-ligand decoy set, CASF-2013,[5] which includes decoy structures generated from three docking packages using different docking algorithms. The results suggest that our RF models outperformed other scoring functions on accuracy and native binding pose selection. By including comparisons between the best decoy pose and the remaining decoy pose structures, the RMSD value of the best decoy was reduced. We also tested the importance of GARF in creating the corresponding RF models. The use of a

scrambled GARF probability function to build a RF model provided evidence for the significance of the GARF potential, while the uniform GARF potential indicated that peak position (or the well position) is most relevant in building a RF model. Finally, we tested the influence of training set size, which showed that the accuracy converged when ~60% of the data set was used in building the RF model. Overall, we showed that potential function based RF models perform at a high level when identifying a native pose from a collection of decoys.

ASSOCIATED CONTENT

Supporting Information

Supplementary description of summary of peak positions and number of probability functions at each peak positions in GARF (Table s1), accuracy values for different training set size from RF models with original and uniform GARF (Table s2), atom types in GARF potential data base (Table s3), summary of importance factor for each atom pair in CASF-2013 (Table s4).

AUTHOR INFORMATION

Corresponding Author

* E-mail: kmerz1@gmail.com

ORCID

Jun Pei: 0000-0002-0204-0896d
Zheng Zheng: 0000-0001-5221-3209
Hyunji Kim: 0000-0002-7251-2545
Lin Frank Song: 0000-0002-1854-5215
Sarah Walworth: 0000-0002-6158-9598
Margaux R. Merz: 0000-0001-5707-7031
Kenneth M. Merz Jr: 0000-0001-9139-5893

Present Address

‡Z.Z.: School of Chemistry, Chemical Engineering and Life Science, Wuhan University of Technology, 122 Luoshi Road, Wuhan 430070, PR China

[a]H.K.: Department of Biomedical Engineering, The George Washington University, 2121 I St NW, Washington, DC 20052, United States
[b]S.W.: Department of Psychology and Neuroscience University of Colorado Boulder Muenzinger D244, 345 UCB, Boulder, Colorado 80302, United States
[c]M.M.: Greenhills School, 850 Greenhills Drive Ann Arbor 48105, United States

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

The authors thank the high-performance computing center (HPCC) at Michigan State University for providing computational resources; HK and SW thank the NSF through the iCER-ACRES REU grant (Grant OAC-1560168) for summer research support; MM thanks the Advanced Research Program sponsored by Greenhills School for support during this project.

References

(1) Jones, G.; Willett, P.; Glen, R. C.; Leach, A. R.; Taylor, R. Development and validation of a genetic algorithm for flexible docking. *J. Mol. Biol*. **1997**, *267*, 727−748.
(2) Jain, A. N. Surflex: Fully automatic flexible molecular docking using a molecular similarity-based search engine. *J. Med. Chem*. **2003**, *46*, 499−511.
(3) Friesner, R. A.; Banks, J. L.; Murphy, R. B.; Halgren, T. A.; Klicic, J. J.; Mainz, D. T.; Repasky, M. P.; Knoll, E. H.; Shelley, M.; Perry, J. K.; Shaw, D. E.; Francis, P.; Shenkin, P. S. Glide: A New Approach for Rapid, Accurate Docking and Scoring. 1. Method and Assessment of Docking Accuracy. *J. Med. Chem*. **2004**, *47*, 1739−1749.
(4) Shivakumar, D.; Williams, J.; Wu, Y.; Damm, W.; Shelley, J.; Sherman, W. Prediction of Absolute Solvation Free Energies using Molecular Dynamics Free Energy Perturbation and the OPLS Force Field. *J. Chem. Theory Comput*. **2010**, *6*, 1509−1519. PMID: 26615687.
(5) Li, Y.; Liu, Z.; Li, J.; Han, L.; Liu, J.; Zhao, Z.; Wang, R. Comparative assessment of scoring functions on an updated benchmark: 1. compilation of the test set. *J. Chem. Inf. Model.* **2014,** *54(6),* 1700–1716.
(6) Zheng, Z.; Pei, J.; Bansal, N.; Liu, H.; Song, L. F.; Merz, K. M. Generation of Pairwise Potentials Using Multidimensional Data Mining. *J. Chem. Theory Comput.* **2018,** *14(10),* 5045–5067.
(7) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; Duchesnay, É. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **2011**, *12*, 2825–2830.
(8) Jiménez, J.; Škalič, M.; Martínez-Rosell, G.; De Fabritiis, G.. KDEEP: Protein-Ligand Absolute Binding Affinity Prediction via 3D-Convolutional Neural Networks. *J. Chem. Inf. Model.* **2018,** *58(2),* 287–296.

(9) Wang, R.; Lai, L.; Wang, S. Further development and validation of empirical scoring functions for structure-based binding affinity prediction. *J. Comput. Aided. Mol. Des*. **2002**, *16*, 11−26.
(10) Cang, Z., Wei, G. W. Integration of element specific persistent homology and machine learning for protein-ligand binding affinity prediction. *Int. J. Numer Meth. Biomed. Engng*. **2018,** *34(2)*, 1–17.
(11) Ragoza, M.; Hochuli, J.; Idrobo, E.; Sunseri, J.; Koes, D. R. Protein-Ligand Scoring with Convolutional Neural Networks. *J. Chem. Inf. Model.* **2017,** *57(4)*, 942–957.
(12) Meng, E. C.; Shoichet, B. K.; Kuntz, I. D. Automated docking with grid-based energy evaluation. *J. Comput. Chem.* **1992**, *13*, 505− 524.
(13) Makino, S.; Kuntz, I. D. Automated flexible ligand docking: method and its application for database search. *J. Comput. Chem.* **1997**, *18*, 1812−1825.
(14) Goodsell, D. S.; Morris, G. M.; Olson, A. J. Automated docking of flexible ligands: applications of AutoDock. *J. Mol. Recog*. **1996,** *9*, 1− 5.
(15) Ortiz, A. R.; Pisabarro, M. T.; Gago, F.; Wade, R. C. Prediction of Drug Binding Affinities by Comparative Binding Energy Analysis. *J. Med. Chem*. **1995**, *38*, 2681−2691.
(16) Yin, S.; Biedermannova, L.; Vondrasek, J.; Dokholyan, N. V. MedusaScore: An Accurate Force Field-Based Scoring Function for Virtual Drug Screening. *J. Chem. Inf. Model.* **2008**, *48*, 1656−1662.
(17) Jones, G.; Willett, P.; Glen, R. C.; Leach, A. R.; Taylor, R. Development and validation of a genetic algorithm for flexible docking. *J. Mol. Biol.* **1997**, *267*, 727−748.
(18) Aqvist, J.; Medina, C.; Samuelsson, J. E. New method for predicting binding affinity in computer-aided drug design. *Protein Eng.* **1994**, *7*, 385−391.
(19) Almlof, M.; Brandsdal, B. O.; Aqvist, J. Binding Affinity Prediction with Different Force Fields: Examination of the Linear Interaction Energy Method. *J. Comput. Chem.* **2004**, *25*, 1242−1254.
(20) Carlson, H. A.; Jorgensen, W. L. Extended linear response method for determining free energies of hydration. *J. Phys. Chem.* **1995**, *99*, 10667−10673.
(21) Jones-Hertzog, D. K.; Jorgensen, W. L. Binding affinities for sulfonamide inhibitors with human thrombin using monte carlo simulations with a linear response method. *J. Med. Chem*. **1997**, *40*, 1539−1549.
(22) Kollman, P. A.; Massova, I.; Reyes, C.; Kuhn, B.; Huo, S.; Chong, L.; Lee, M.; Lee, T.; Duan, Y.; Wang, W.; Donini, O.; Cieplak, P.; Srinivasan, J.; Case, D. A.; Cheatham, T. E. Calculating Structures and Free Energies of Complex Molecules: Combining Molecular Mechanics and Continuum Models. *Acc. Chem. Res*. **2000**, *33*, 889− 897.
(23) DeWitte, R. S.; Shakhnovich, E. I. SMoG: de Novo Design Method Based on Simple, Fast, and Accurate Free Energy Estimates. 1. Methodology and Supporting Evidence. *J. Am. Chem. Soc*. **1996**, *118*, 11733−11744.
(24) Grzybowski, B. A.; Ishchenko, A. V.; Shimada, J.; Shakhnovich, E. I. From Knowledge-Based Potentials to Combinatorial Lead Design in Silico. *Acc. Chem. Res.* **2002**, *35*, 261−269.
(25) Muegge, I.; Martin, Y. C. A general and fast scoring function for protein-ligand interactions: A simplified potential approach. *J. Med. Chem.* **1999**, *42*, 791−804.
(26) Muegge, I. A knowledge-based scoring function for protein- ligand interactions: Probing the reference state. *Perspectives in Drug Discovery & Design.* **2000,** *20*, 99−114.
(27) Muegge, I. Effect of ligand volume correction on PMF scoring. *J. Comput. Chem.* **2001**, *22*, 418−425.

(28) Gohlke, H.; Hendlich, M.; Klebe, G. Knowledge-based scoring function to predict protein-ligand interactions. *J. Mol. Biol*. **2000**, *295*, 337−356.
(29) Velec, H. F. G.; Gohlke, H.; Klebe, G. DrugScore(CSD): Knowledge-Based Scoring Function Derived from Small Molecule Crystal Data with Superior Recognition Rate of Near-Native Ligand Poses and Better Affinity Prediction. *J. Med. Chem.* **2005**, *48*, 6296−6303.
(30) Neudert, G.; Klebe, G. DSX: A Knowledge-Based Scoring Function for the Assessment of Protein-Ligand Complexes. *J. Chem. Inf. Model.* **2011**, *51*, 2731−2745.
(31) Huang, S.-Y.; Zou, X. An iterative knowledge-based scoring function to predict protein-ligand interactions: I. Derivation of interaction potentials. *J. Comput. Chem.* **2006**, *27*, 1865−1875.
(32) Huang, S. Y.; Zou, X. An iterative knowledge-based scoring function to predict protein-ligand interactions: II. Validation of the scoring function. *J. Comput. Chem.* **2006**, *27*, 1876−1882.
(33) Huang, S. Y.; Zou, X. Inclusion of solvation and entropy in the knowledge-based scoring function for protein-ligand interactions. *J. Chem. Inf. Model.* **2010**, *50*, 262−273.
(34) Zheng, Z.; Merz, K. M. Development of the Knowledge-Based and Empirical Combined Scoring Algorithm (KECSA) To Score Protein−Ligand Interactions. *J. Chem. Inf. Model.* **2013**, *53*, 1073− 1083.
(35) Bohm, H. J. The development of a simple empirical scoring function to estimate the binding constant for a protein-ligand complex of known three-dimensional structure. *J. Comput. Aided. Mol. Des.* **1994**, *8*, 243−256.
(36) Verkhivker, G.; Appelt, K.; Freer, S. T.; Villafranca, J. E. Empirical free energy calculations of ligand-protein crystallographic complexes. I. Knowledge-based ligand-protein interaction potentials applied to the prediction of human immunodeficiency virus 1 protease binding affinity. *Protein Eng.* **1995**, *8*, 677−691.
(37) Eldridge, M. D.; Murray, C. W.; Auton, T. R.; Paolini, G. V.; Mee, R. P. Empirical scoring functions: I. The development of a fast empirical scoring function to estimate the binding affinity of ligands in receptor complexes. *J. Comput. Aided. Mol. Des.* **1997**, *11*, 425−445.
(38) Murray, C. W.; Auton, T. R.; Eldridge, M. D. Empirical scoring functions. II. The testing of an empirical scoring function for the prediction of ligand-receptor binding affinities and the use of Bayesian regression to improve the quality of the model. *J. Comput. Aided. Mol. Des.* **1998**, *12*, 503−519.
(39) Friesner, R. A.; Banks, J. L.; Murphy, R. B.; Halgren, T. A.; Klicic, J. J.; Mainz, D. T.; Repasky, M. P.; Knoll, E. H.; Shelley, M.; Perry, J. K.; Shaw, D. E.; Francis, P.; Shenkin, P. S. Glide: A New Approach for Rapid, Accurate Docking and Scoring. 1. Method and Assessment of Docking Accuracy. *J. Med. Chem.* **2004**, *47 (7)*, 1739− 1749.
(40) Friesner, R. A.; Murphy, R. B.; Repasky, M. P.; Frye, L. L.; Greenwood, J. R.; Halgren, T. A.; Sanschagrin, P. C.; Mainz, D. T. Extra Precision Glide: Docking and Scoring Incorporating a Model of Hydrophobic Enclosure for Protein-Ligand Complexes. *J. Med. Chem.* **2006**, *49*, 6177−6196.
(41) Deng, W.; Breneman, C.; Embrechts, M. J. Predicting Protein- Ligand Binding Affinities Using Novel Geometrical Descriptors and Machine-Learning Methods. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 699− 703.

(42) Zhang, S.; Golbraikh, A.; Tropsha, A. Development of Quantitative Structure-Binding Affinity Relationship Models Based on Novel Geometrical Chemical Descriptors of the Protein-Ligand Interfaces. *J. Med. Chem.* **2006**, *49*, 2713−2724.

(43) Durrant, J. D.; McCammon, J. A. NNScore: A Neural-Network- Based Scoring Function for the Characterization of Protein-Ligand Complexes. *J. Chem. Inf. Model.* **2010**, *50*, 1865−1871.

(44) Durrant, J. D.; McCammon, J. A. NNScore 2.0: A Neural- Network Receptor-Ligand Scoring Function. *J. Chem. Inf. Model.* **2011**, *51*, 2897−2903.

(45) Ballester, P. J.; Mitchell, J. B. O. A machine learning approach to predicting protein−ligand binding affinity with applications to molecular docking. *Bioinformatics* **2010**, *26*, 1169−1175.

(46) Ballester, P. J.; Schreyer, A.; Blundell, T. L. Does a More Precise Chemical Description of Protein−Ligand Complexes Lead to More Accurate Prediction of Binding Affinity*? J. Chem. Inf. Model.* **2014**, *54*, 944−955.

(47) Zilian, D.; Sotriffer, C. A. SFCscore^{RF}: A Random Forest-Based Scoring Function for Improved Affinity Prediction of Protein-Ligand Complexes. *J. Chem. Inf. Model.* **2013**, *53*, 1923−1933.

(48) Li, G. B.; Yang, L. L.; Wang, W. J.; Li, L. L.; Yang, S. Y. ID- Score: A New Empirical Scoring Function Based on a Comprehensive Set of Descriptors Related to Protein−Ligand Interactions. *J. Chem. Inf. Model.* **2013**, *53*, 592−600.

(49) Deng, Z.; Chuaqui, C.; Singh, J. Structural Interaction Fingerprint (SIFt): A Novel Method for Analyzing Three-Dimensional Protein-Ligand Binding Interactions. *J. Med. Chem.* **2004**, *47*, 337− 344.

(50) Dangetti, P. Journey from Statistics to Machine Learning. In *Statistical for Machine Learning*, Editing, S., Pagare, V., Singh, A., Pawanikar, M., Pawar, D. Ltd: Packt Publishing, Birmingham, United Kingdom, 2017; pp 9.

(51) Pei, J.; Zheng, Z.; Merz, K. M. Random Forest Refinement of the KECSA2 Knowledge-based Scoring Function for Protein Decoy Detection. *J. Chem. Inf. Model.* **2019**