

# Ranking Protein-Protein Binding Using Evolutionary Information and Machine Learning

Roshanak Farhoodi  
University of Massachusetts Boston  
100 Morrissey Blvd.  
Boston, Massachusetts 02125  
rfarhoodi@gmail.com

Bahar Akbal-Delibas  
Kadir Has University  
Kadir Has Caddesi, Cibali  
Istanbul, Turkey  
bahar.delibas@khas.edu.tr

Nurit Haspel\*  
University of Massachusetts Boston  
100 Morrissey Blvd.  
Boston, Massachusetts 02125  
nurit.haspel@umb.edu

## ABSTRACT

Discriminating native-like complexes from false-positives with high accuracy is one of the biggest challenges in protein-protein docking. The relationship between various favorable intermolecular interactions (e.g., Van der Waals, electrostatic, desolvation forces, etc.) and the similarity of a conformation to its native structure is commonly agreed, though the precise nature of this relationship is not known very well. Existing protein-protein docking methods typically formulate this relationship as a weighted sum of selected terms and tune their weights by introducing a training set with which they evaluate and rank candidate complexes. Despite improvements in recent docking methods, they are still producing a large number of false positives, which often leads to incorrect prediction of complex binding. Using machine learning, we implemented an approach that not only ranks candidate complexes relative to each other, but also predicts how similar each candidate is to the native conformation. We built a Support Vector Regressor (SVR) using physico-chemical features and evolutionary conservation. We trained and tested the model on extensive datasets of complexes generated by three state-of-the-art docking methods. The set of docked complexes was generated from 79 different protein-protein complexes in both the rigid and medium categories of the Protein-Protein Docking Benchmark v.5. We were able to generally outperform the built-in scoring functions of the docking programs we used to generate the complexes, attesting to the potential of our approach in predicting the correct binding of protein-protein complexes.

## CCS CONCEPTS

• **Computing methodologies** → **Machine Learning; Supervised learning; Support vector machines; Neural networks**; • **Applied computing** → **Computational biology; Molecular structural biology; Bioinformatics**;

## KEYWORDS

Protein-Protein docking, machine learning, evolutionary conservation, SVR

\*Corresponding author

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).  
ACM-BCB'17, August 20–23, 2017, Boston, MA, USA.  
© 2017 ACM. ISBN 978-1-4503-4722-8/17/08...\$15.00  
DOI: <http://dx.doi.org/10.1145/3107411.3107497>

## ACM Reference format:

Roshanak Farhoodi, Bahar Akbal-Delibas, and Nurit Haspel. 2017. Ranking Protein-Protein Binding Using Evolutionary Information and Machine Learning. In *Proceedings of ACM-BCB'17, August 20–23, 2017, Boston, MA, USA.*, 6 pages.

DOI: <http://dx.doi.org/10.1145/3107411.3107497>

## 1 INTRODUCTION

Proteins play a major role in nearly any vital biological function [14, 21]. Proteins often bind to other proteins as part of their function, to form *protein complexes* [13]. In order to understand many of the important roles proteins play, we must have good understanding of their structure and function [14, 17, 24].

Computational docking methods aim to compute the correct bound form of two or more molecules. Protein-protein docking methods take two (or more) protein structures and try to predict the structure of the complex formed by them. This is a highly challenging task because even for rigid body docking, the search space spans the three translational and three rotational degrees of freedom for the other protein. Therefore, the search space grows exponentially with the size of the input proteins [25]. To make the problem more tractable, some docking programs allow the user to add a-priori knowledge like interacting residues [9, 11] or an initial conformation which is the basis for a local search [22].

*Protein-Protein Scoring Functions:* Most docking algorithms apply a geometric search for the correctly bound complex, followed by a ranking/scoring stage where a scoring function aims to distinguish native-like candidates from false positives. Scoring functions are designed to favor conformations with low binding energy, good geometric fit, clusters of conserved residues on the binding interfaces, and more. Over the past 20 years several scoring functions have been developed for ranking putative docked complexes. These scoring functions combine geometric complementarity with physical and chemical interactions [9, 11, 15, 20, 22]. These functions often use a combination of Van der Waals (VdW) energy, electrostatic interactions and desolvation terms. The combination and weighing of the terms differs among different methods. A recent docking refinement method by our group [2, 3] uses a scoring function that also includes an evolutionary traces (ET) term [30]. The assumption is that binding interfaces tend to be conserved due to their evolutionary importance.

Modern docking algorithms are often successful in predicting the correctly bound complex of their input proteins but many times the highest ranking docking candidates are still often false positives [15, 16]. A recent large-scale benchmarking of many current

docking methods revealed that most physics-based scoring functions still fail to accurately predict the binding affinity of complexes [17]. In other words, the top scoring candidates are not always the closest ones to the native complex. Furthermore, even the most accurate scoring functions cannot always accurately estimate the least RMSD (IRMSD) of a docked structure with respect to its native conformation, since their aim is to provide relative ranking of a set of docked structures. Therefore, more work is needed to improve the existing scoring functions or design new methods.

It is generally agreed that there is a relationship between various scoring terms (e.g., Van der Waals, electrostatic, desolvation forces, etc.) and the similarity of a docked complex to its native structure [4]. However, the exact form of this relationship is unknown. Therefore, docking algorithms often formulate this relationship as a weighted sum of selected energetic, biochemical or geometric terms and adjust their weights against a training set [24]. Yet, the general inaccuracy of the rankings may suggest that the relationship between the scoring function terms and IRMSD of a conformation may be more complex than a weighted sum. For this reason, many docking algorithms provide an additional, sometimes optional, refinement stage, where selected putative complexes are being refined and re-ranked in order to improve the geometric fit and the binding energy of the candidate complexes.

*Machine Learning for Docking:* In recent years there has been an increasing use of machine learning based methods in bioinformatics. Support Vector Machines (SVM) are one of supervised learning methods that are widely used in solving classification problems. An SVM model is a representation of the training samples as points in space that are mapped so that different classes are separated by a gap that is as wide as possible. To classify new samples, features of the new data is used to map the samples into the same space and their class label is determined based on which side of the gap they fall on. SVMs can also be used to do nonlinear classification by using a kernel, where the samples are mapped into high-dimensional feature spaces to achieve separation of the target classes [10]. SVMs can be modified to suit regression problems, where they are referred to as Support Vector Regression (SVR) methods. Inspired by statistical learning theory, support vector methods aim at minimizing the training error while trying to keep the complexity of the function to be learned under control. SVR learns the nonlinear mapping between the feature values and the output values of the given training set in the form of a function, and this function can be later used to do prediction of target values. SVMs, SVRs and other similar kernel methods have been successfully used in bioinformatics applications including protein interaction prediction in the context of interaction networks [6], ranking of predicted protein structures [28], protein function classification [8] and protein ligand docking [18]. To the best of our knowledge, there are not many machine learning methods applied to protein-protein docking. In this work we use an SVR model as our scoring function in protein complex similarity prediction.

*This Contribution:* The objective of this paper is two-fold: first, based on our previous work [1, 5, 12], we describe a new SVR based machine learning approach, to predict the IRMSD of a set of candidate complexes with respect to their native conformation. Our method includes evolutionary conservation information in addition

to physico-chemical interactions. Through a set of cross validation experiments, the SVR model showed comparable performance with the best performing method from our previous work, a multilayer neural network, while being much faster (The comparisons are not shown in this paper). Additionally, our SVR prediction model shows comparable and often better performance than the built-in scoring functions of three well-known docking methods, which were used to produce the protein complexes.

Second, in our previous work we used smaller sets of protein complexes including mostly near-native complexes of IRMSD range (0-7Å) [12]. That set provided only a partial sampling of the conformational space of docking candidates, which often span a very wide range of RMSDs from the native complex. In this paper, in order to conduct a more extensive analysis, we used complexes with much wider RMSD ranges. We also used a larger and more diverse set of 79 complexes from both the easy and medium categories of the Protein-Protein Docking Benchmark v.5 [29]. Our experiments can be used as a guiding tool for building the right training dataset and employing an accurate model in the studies that rely on identifying the best docking candidate complexes.

## 2 METHODS

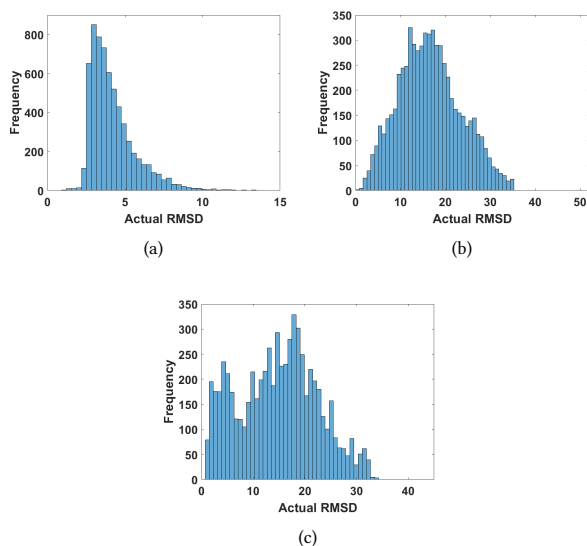
### 2.1 Generating the Complexes

We initially selected 81 protein-protein dimers from the easy (rigid) and medium categories in the Protein-Protein Docking Benchmark v.5 [29] for which the corresponding evolutionary trace files were available in the ET Server [23]. Then, we generated docking results for each of these input complexes with PyDock [9], coarse RosettaDock [22] (without refinement) and a version of ClusPro [20] which generates the candidate complexes using the PIPER scoring function [19], without the clustering phase (S. Vajda, personal communication). For each protein, we retained the top-ranking 100 complexes generated by each one of the three docking algorithms, as ranked by that docking program's built-in scoring function. The IRMSD distribution of the generated candidates with respect to the native complex is shown in Table 1 and Figures 1-2.

To generate VdW and electrostatic values for our scoring function, we added hydrogen atoms using CHARMM [7] followed by 500 steps of energy minimization using NAMD [27] to resolve collisions without creating large changes to the complexes. Following these stages, two complexes were excluded due to problems in the calculation of the evolutionary trace values. The remaining 79 complexes were used to produce the results below.

### 2.2 Training Dataset

The training datasets contain 6,400 complexes (100 complexes for each protein) generated for the following 44 proteins from the easy category and 20 proteins from the medium category of the Protein-Protein Docking Benchmark v.5 [29]. The easy complexes are: 1Z5Y, 2AJF, 1GLA, 1JTD, 1YVB, 2GTP, 1EWY, 3A4S, 1J2J, 1T6B, 1US7, 1OC0, 1ZHI, 1OYV, 1H9D, 2I25, 2VDB, 1ZHH, 2HLE, 1EFN, 1B6C, 2O0B, 2O8V, 1Z0K, 1PVH, 4H03, 3BIW, 3VLB, 1GL1, 2YVJ, 2A9K, 2AYO, 2FJU, 2G77, 2J0T, 2SNI, 3PC8, 1R0R, 4M76, 7CEI, 2GAF, 2B42, 1GXD, 2A5T



**Figure 1: RMSD (Å) distribution with respect to the native complex of training datasets generated by (a) RosettaDock, (b) pyDock and (c) ClusPro.**

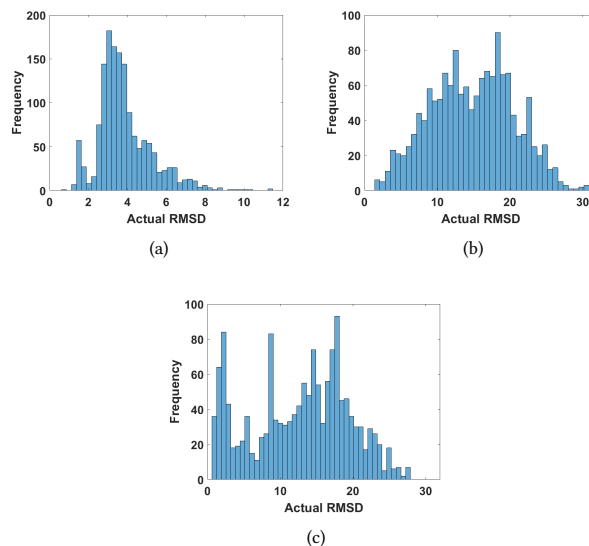
The complexes from the medium category are: 1GRN, 2HRK, 1LFD, 3CPH, 2Z0E, 1XQS, 1R6Q, 3DAW, 4IZ7, 1WQ1, 2CFH, 1CGI, 1I2M, 1ZM4, 1NW9, 1HE8, 1MQ8, 2OZA, 3S9D, 4FZA.

### 2.3 Test Dataset

The test set includes 1,000 complexes (100 complexes for each protein) from the following 10 proteins in the easy category: 3D5S, 3K75, 2HQ5, 1JTG, 1GPW, 1XD3, 2A1A, 4CPA, 1FFW, 1S1Q, and 500 complexes from the following 5 proteins in the medium category: 1SYX, 1JIW, 1M10, 3BX7, 3AAD.

**Table 1: Training and test datasets statistics summary: minimum, mean, maximum and standard deviation of the least RMSD (lRMSD) values of the samples in each dataset and the methods used to generate the samples (Tr=Training, Te=Testing, N=Number of proteins in each set)**

Set	Range	Mean	Std	Method	N
Tr 1	1.1–14.41	4.2	1.49	Rosetta	64
Tr 2	0.74–51.78	17.05	7.59	pyDock	64
Tr 3	0.75–44.37	14.91	7.79	Cluspro	64
Tr 4	0.74–51.78	15.98	7.77	pyDock-Cluspro	64
Te 1	0.77–11.44	3.84	1.38	Rosetta	15
Te 2	1.47–31.55	14.85	5.71	pyDock	15
Te 3	0.66–27.67	12.71	6.76	Cluspro	15
Te 4	0.66–31.55	13.78	6.25	pyDock-Cluspro	15



**Figure 2: RMSD (Å) distribution with respect to the native complex of the test datasets generated by (a) RosettaDock, (b) pyDock, and (c) ClusPro.**

### 2.4 Features

Our prediction methods approximate the relationship between 16 different features and the lRMSD of a protein complex with respect to its native structure. The majority of these features are used as scoring function terms by multiple docking and refinement methods [9, 19]. Additionally, we have an evolutionary conservation based feature (ICAR) [4]. The features are as follows:

- Van der Waals (VdW): The VdW force for interface atoms (defined as the atoms within at most 6Å to the adjacent chain atoms) is computed using a soft Lennard-Jones potential [2].
- Electrostatic: Computed for interface atoms, based on Coulomb's law as explained in [2].
- Interface Conserved Atom Ratio (ICAR): the ratio of the evolutionarily conserved interface atoms to the total interface size, see [4].
- Complex Category: The numeric representation of the protein category, as defined in the Protein-Protein Docking Benchmark v.5 [29].
- The fraction of interface atoms belonging to a residue type: Hydrophobic (A, C, G, I, L, M, P, V); Positively Charged (H, K, R); Negatively Charged (D, E); Polar (N, Q, S, T); Aromatic (F, H, W, Y).

### 2.5 Prediction Method: SVR

We used the training complexes represented with the above 16 features to train the SVR model. Eight of these features consist of continuous values and were initially scaled to the range of [0..1]. The remaining eight features, that are used to represent the eight different protein categories, have been used as binary categorical features. The lRMSD values of the training samples have been scaled

to the range of [0..1] as well. After several exploratory rounds of parameter tuning and cross validation (exhaustive grid search), we chose RBF (Radial Basis Function) as the kernel for the SVR model with kernel coefficient (gamma) equal to 0.01. The penalty term for the error was chosen to be 0.9. After the model was implemented using the module provided by Scikit-learn [26] and trained using the training sets, the models were then used to predict the output value of the given test structures, where the resulting values were re-scaled to represent the final predicted IRMSD value.

### 3 RESULTS AND DISCUSSIONS

In this section, we discuss the prediction accuracy of the models by comparing predicted and actual IRMSDs of the samples in our test datasets, as well as describing our cross-validation experiments for comparing different datasets that we used to train the SVR models. Last but not least, we compared the performance of SVR with the scoring functions of Rosetta, pyDock and ClusPro using two objective metrics that will be described in next subsections. We mainly aimed at comparing the predictive power of SVR in protein complex ranking with other well-known methods. We first conducted a set of experiments to compare the prediction accuracy of the models when trained using each of our datasets. In these experiments we measured the errors and Pearson correlations of the actual and predicted RMSD values. Then, a set of 5-fold cross validation tests were conducted to unbiasedly examine the accuracy of the models using randomly selected training and testing sets. Finally, we examined how SVR stands in ranking the top 100 structures for each of the candidate complexes generated by Rosetta, pyDock and ClusPro when compared to those methods' rankings.

#### 3.1 Performance Testing

We tested the performance of our method against complexes from the easy and medium category of the protein-protein benchmark [29]. The medium difficulty complexes are harder to predict, since they model possible conformational changes upon binding. Each of the test sets had a total of 1,500 candidates from 15 proteins that we had randomly selected for testing (see Test Datasets above). The Pearson correlation coefficients of the predicted and actual IRMSD values and the error of these experiments for each test protein is listed in Table 2. The lowest average prediction error of 1.15Å was observed with the structures generated by RosettaDock and the highest average error of 6.09Å was returned by the ClusPro dataset. Despite having proteins with medium difficulty, we were able to obtain prediction errors within less than one standard deviation of the IRMSD distribution, and the correlation coefficients were all above 0.33.

Finally, our goal is to make our method agnostic to the docking method used to generate the complexes. We trained the model using the dataset that we built by combining the structures generated by pyDock and ClusPro. We kept the RosettaDock out of this experiment since it is a local search method as opposed to the FFT-based search used by both ClusPro and PyDock, and hence the IRMSD range is much lower, which would result in a vastly different conformational space. The test results using this 4th dataset are shown in Table 2. The average prediction error of 5.82Å was

returned, which not surprisingly is approximately midway between the ClusPro and pyDock separate predictions.

Looking at the prediction results, it is worth highlighting that the correlation coefficients of the actual and predicted RMSD values as well as the prediction accuracy of models vary from case to case for a given protein due to the diversity of the distribution of features and RMSD values for each protein. Also, a general observation worth noting is that the prediction errors of the models trained with samples generated using RosettaDock were on average smaller compared to the models trained with samples generated by pyDock and ClusPro. We attribute this to having considerably more samples with lower IRMSD values in training and test datasets generated by RosettaDock due to its local sampling nature, while the complexes obtained by pyDock and ClusPro had a much wider spread over the higher ranges of IRMSD values.

**Table 2: SVR Prediction errors and Pearson correlations using complexes from the rigid+medium category generated by Rosetta, pyDock and ClusPro and a combined set of pyDock and ClusPro (Err=Error, Co=Pearson correlation, C. and D.=ClusPro and pyDock)**

PDB	Rosetta		pyDock		ClusPro		C. and D.	
	Err	Co	Err	Co	Err	Co	Err	Co
3D5S	0.87	0.58	6.53	0.33	13.07	0.3	10.43	0.08
3K75	0.79	0.25	5.9	0.34	2.87	0.15	4.68	0.04
2HQS	1.47	0.64	6.68	0.17	5.99	0.23	6.11	0.17
1JTG	1.91	0.01	5.7	0.23	8.35	0.08	7.11	0.27
1GPW	0.99	0.44	4.65	0.57	8.92	0.2	6.86	0.6
1XD3	0.71	0.38	7.45	0.63	12.43	0.19	10.28	0.6
2A1A	1.42	0.4	6.16	0.67	5.35	0.74	5.73	0.76
4CPA	0.7	0.37	4.8	0.77	2.57	0.49	3.01	0.65
1FFW	0.64	0.16	4.71	0.34	3.98	0.13	4.19	0.3
1S1Q	0.96	0.21	2.82	0.53	2.58	0.64	2.82	0.67
1SYX	0.51	0.4	3.67	0.07	2.71	0.63	3.46	0.21
1JIW	0.9	0.2	5.57	0.16	6.62	0.07	6.23	0.07
1M10	2.46	0.36	6.27	0.18	4.21	0.3	5.13	0.41
3BX7	1.42	0.64	6.01	0.26	6.18	0.91	6.11	0.49
3AAD	1.44	0.13	4.64	0.51	5.53	0.73	5.17	0.49
Avg.	1.15	0.34	5.44	0.38	6.09	0.37	5.82	0.39

#### 3.2 Model Comparison by Cross Validation

In order to analyze the performance of the models in an unbiased fashion and to demonstrate that characteristics of individual protein complexes in our test sets is not in any way favoring the prediction, we conducted 5-fold cross validation experiments. We combined the training/test datasets and randomly divided the samples into training and testing sets, where 80 percent of total samples were used for training and the remaining samples were set aside for testing. This was repeated in an iterative manner for 5 times such that no samples generated for a particular protein could fall in both training and testing sets.

The prediction errors and correlation coefficients for the four datasets are summarized in Table 3. Similarly, the lowest average

error of 1.34Å and highest average correlation of 0.47 were generated by RosettaDock complexes. Average prediction errors of 6.57Å 7.6Å and 7.17Å were produced by pyDock, ClusPro and the combined dataset respectively. The prediction error was similar to the test sets reported above, with a slightly lower but still positive correlation.

**Table 3: 5-fold cross validation SVR results using the structures in medium category generated by Rosetta, pyDock and ClusPro.**

Fold	Rosetta		pyDock		ClusPro		C. and D.	
	Err	Co	Err	Co	Err	Co	Err	Co
1	1.25	0.33	6.41	0.29	6.52	0.42	6.69	0.33
2	1.22	0.47	6.83	0.34	7.98	0.18	7.4	0.27
3	1.25	0.43	6.33	0.28	6.21	0.39	6.67	0.22
4	1.79	0.36	7.24	0.4	9.26	0.08	8.3	0.21
5	1.21	0.59	6.05	0.29	8.03	0.15	6.75	0.27
Avg.	1.34	0.47	6.57	0.32	7.6	0.24	7.17	0.26

### 3.3 Comparing SVR with the Scoring Functions

Finally, we compared the predictive ability of our model with the built-in scoring functions used by the docking methods to rank the candidate complexes. For each complex, we compared the relative ranking of the 100 docking candidates based on the docking method's scoring function against the predicted IRMSD produced by our model. As a ground truth we ranked the candidate complexes by their IRMSD from their native complex. In order to conduct an objective comparison for the ranking performance of the SVR and the other 3 docking methods, two measurements are used: (1) the number of correctly identified top ten structures for each complex, and (2) each method's ranking of a structure against its real ranking is compared and the root mean square error in ranking is calculated.

Tables 4, 5 and 6 present how RosettaDock, pyDock and ClusPro ranked the 100 solutions for each protein. In these set of experiments, SVR performed better than pyDock and ClusPro in identifying top-10 complexes, and had lower average ranking error. Our model performs slightly worse than RosettaDock in average ranking error. RosettaDock achieved lower ranking error in 73 percent of the cases, but the difference is rather small (25.57 for RosettaDock vs. 26.98 for SVR). Finally, SVR and RosettaDock identified, on average, the same number of top-10 complexes, but our model was able to identify more top-10 complexes in six out of the 15 test cases, whereas RosettaDock identified more top-10 complexes in four out of the 15 cases.

## 4 CONCLUSIONS

We presented a machine learning approach to predict and rank protein-protein docking candidates. A major challenge in protein-protein docking is that existing scoring functions still produce a large number of false positive candidates, which are high-ranking complexes with high RMSD with respect to the native complex.

**Table 4: Comparison of Rosetta and SVR by the number of best structures included in their top-10, and by their error in ranking, medium category.**

PDB	Error in ranking		# of detected top-10	
	Rosetta	SVR	Rosetta	SVR
3D5S	20.68	20.78	3	5
3K75	30.06	30.46	0	1
2HQS	11.06	17.87	6	4
1JTG	32.46	34.34	2	1
1GPW	23.5	24.42	3	3
1XD3	26.26	<b>24.66</b>	3	1
2A1A	20.68	24.54	1	2
4CPA	19.2	24.38	0	1
1FFW	40.04	<b>37.06</b>	0	0
1S1Q	23.44	27.38	1	1
1SYX	29.92	<b>26.58</b>	0	0
1JIW	24.5	30.52	0	0
1M10	24.34	25.76	3	1
3BX7	18.78	19.68	2	3
3AAD	38.7	<b>36.24</b>	0	1
Avg.	25.57	26.98	1.6	1.6

**Table 5: Comparison of pyDock and SVR by the number of best structures included in their top-10, and by their error in ranking, medium category.**

PDB	Error in ranking		# of detected top-10	
	pyDock	SVR	pyDock	SVR
3D5S	33.6	40.68	3	0
3K75	31.9	<b>28.36</b>	3	6
2HQS	35.72	23	1	3
1JTG	31.46	<b>27.88</b>	4	1
1GPW	28.2	<b>20.6</b>	4	3
1XD3	32.22	<b>18.42</b>	1	3
2A1A	30.4	<b>22.12</b>	3	5
4CPA	30.6	<b>15.54</b>	0	2
1FFW	33.52	39.04	2	0
1S1Q	38.5	<b>19.42</b>	0	4
1SYX	35.54	37.68	0	0
1JIW	30.58	32.16	1	2
1M10	36.36	37.34	0	1
3BX7	24.24	25.62	2	1
3AAD	29.36	<b>23.78</b>	1	2
Avg.	32.15	27.44	1.67	2.2

We trained our prediction model on a large number of protein-protein docking candidates with a wide range of RMSDs and complex types. Our features include amino acid type, physico-chemical interactions and evolutionary conservation on the binding interface. We showed that our ranking and predictive ability was comparable to, and in most cases better than, existing scoring functions. Initial results (not shown) demonstrate that the addition of evolutionary

**Table 6: Comparison of ClusPro and SVR by the number of best structures included in their top-10, and by their error in ranking, medium category.**

PDB	Error in ranking		# of detected top-10	
	ClusPro	SVR	ClusPro	SVR
3D5S	36.5	<b>26.66</b>	0	<b>1</b>
3K75	37.74	<b>34.04</b>	0	<b>1</b>
2HQS	34.28	<b>26.7</b>	1	<b>6</b>
1JTG	33.78	34.7	0	1
1GPW	36.02	<b>29.96</b>	1	0
1XD3	29.32	33.46	1	0
2A1A	30.32	<b>14.92</b>	2	<b>5</b>
4CPA	34.76	44.92	1	1
1FFW	34.76	<b>30.22</b>	1	0
1S1Q	35.84	<b>16.76</b>	0	<b>3</b>
1SYX	37.6	<b>22.7</b>	2	<b>3</b>
1JIW	32.08	33.72	2	0
1M10	33	39.56	1	0
3BX7	33.6	<b>15.18</b>	2	<b>4</b>
3AAD	34.54	<b>18.94</b>	3	2
Avg.	33.59	29.75	0.87	1.67

conservation contributes to the better performance of the model in all of our test cases, and especially when the binding interface is not known and the full conformational space is explored. This is the subject of current work. Future work includes incorporating the ranking function into a docking scheme. Most docking programs first perform geometric search, followed by a ranking stage. Incorporating a scoring phase in the search will allow us to filter out implausible candidates and make the search more effective.

## Acknowledgements

The work was funded in part by NSF grant no. CCF-1421871 (NH). Some of the computations were carried out on the UMB research cluster and on the Massachusetts Green High Performance Computing Center (MGHPCC) cluster.

## REFERENCES

- [1] B. Akbal-Delibas, R. Farhoodi, M. Pomplun, and N. Haspel. 2016. Accurate refinement of docked protein complexes using evolutionary information and deep learning. *Journal of Bioinformatics and Computational Biology* 14, 3 (2016), 1642002.
- [2] B. Akbal-Delibas, I. Hashmi, A. Shehu, and N. Haspel. 2012. An evolutionary conservation-based method for refining and reranking protein complex structures. *J Bioinform Comput Biol* 10, 3 (2012).
- [3] B. Akbal-Delibas and N. Haspel. 2013. A conservation and biophysics guided stochastic approach to refining docked multimeric proteins. *BMC Structural Biology* 13, Suppl 1 (2013), S7.
- [4] B. Akbal-Delibas, M. Pomplun, and N. Haspel. 2014. AccuRMSD: a machine learning approach to predicting structure similarity of docked protein complexes. In *Proceedings of the 5th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics*. ACM, 289–296.
- [5] B. Akbal-Delibas, M. Pomplun, and N. Haspel. 2015. Accurate Prediction of Docked Protein Structure Similarity. *J. Comp. Biol.* 22, 9 (2015), 892–904.
- [6] Asa Ben-Hur and William Stafford Noble. 2005. Kernel methods for predicting protein-protein interactions. *Bioinformatics* 21, suppl 1 (2005), i38–i46.
- [7] B Brooks, R.E. Bruccoleri, B.D. Olafson, D.J. States, S. Swaminathan, and M. Karplus. 1983. CHARMM: A program for macromolecular energy, minimization, and dynamics calculations. *J Comput Chem* 4, 2 (1983), 187 – 217.
- [8] C.Z Cai, W.L Wang, L.Z Sun, and Y.Z Chen. 2003. Protein function classification via support vector machine approach. *Mathematical Biosciences* 185, 2 (2003), 111 – 122. DOI: [https://doi.org/10.1016/S0025-5564\(03\)00096-8](https://doi.org/10.1016/S0025-5564(03)00096-8)
- [9] T.M-K. Cheng, T.L. Blundell, and J. Fernandez-Recio. 2007. pyDock: Electrostatics and desolvation for effective scoring of rigid-body protein-protein docking. *Proteins: Structure, Function, and Bioinformatics* 68, 2 (2007), 503–515.
- [10] N. Cristianini and J. Shawe-Taylor. 2000. *An introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press.
- [11] C. Dominguez, R. Boelens, and A. Bonvin. 2003. Haddock: A protein-protein docking approach based on biochemical or biophysical information. *J. Am. Chem. Soc.* 125, 1 (2003), 1731–1737.
- [12] R. Farhoodi, B. Akbal-Delibas, and N. Haspel. 2015. Accurate Prediction of Docked Protein Structure Similarity Using Neural Networks and Restricted Boltzmann Machines. In *CSBW (Computational Structural Bioinformatics Workshop), in conjunction with IEEE-BIBM 2015*.
- [13] D.S. Goodsell and A.J. Olson. 2000. Structural symmetry and protein function. *Annu. Rev. Biophys. Biomol. Struct.* 29, 1 (2000), 105–153.
- [14] J.J. Gray. 2006. High-resolution protein-protein docking. *Current opinion in structural biology* 16, 2 (2006), 183–193.
- [15] I. Halperin, B. Ma, H. Wolfson, and R. Nussinov. 2002. Principles of docking: an overview of search algorithms and a guide to scoring functions. *Proteins: Structure, Function, and Bioinformatics* 47, 4 (2002), 409–443.
- [16] J. Janin. 2010. Protein-protein docking tested in blind predictions: the CAPRI experiment. *Molecular BioSystems* 6, 12 (2010), 2351–2362.
- [17] P. L. Kastiris and A. M. Bonvin. 2010. Are Scoring Functions in Protein-Protein Docking Ready To Predict Interactomes? Clues from a Novel Binding Affinity Benchmark. *J. Proteome Res.* 9, 5 (2010), 2216–2225.
- [18] Mohamed A Khamis, Walid Goma, and Wala F Ahmed. 2015. Machine learning in computational docking. *Artificial intelligence in medicine* 63, 3 (2015), 135–152.
- [19] D. Kozakov, R. Brenke, S.R. Comeau, and S. Vajda. 2006. PIPER: An FFT-based protein docking program with pairwise potentials. *Proteins: Structure, Function, and Bioinformatics* 65, 2 (2006).
- [20] D. Kozakov, D.R. Hall, B. Xia, K.A. Porter, D. Padhorny, C. Yueh, D. Beglov, and S. Vajda. 2017. The ClusPro web server for protein-protein docking. *Nature Protocols* 12 (2017), 275–288.
- [21] A.M. Lesk. 2008. *Introduction to Bioinformatics* (3rd ed.). Oxford University Press.
- [22] S. Lyskov and J. J. Gray. 2008. The RosettaDock server for local protein-protein docking. *Nucleic Acids Res.* 36, S2 (2008), W233–W238.
- [23] I. Mihalik, I. Res, and O. Lichtarge. 2006. Evolutionary Trace Report Maker: a new type of service for comparative analysis of proteins. *Bioinformatics* 22, 13 (2006), 1656–7.
- [24] Iain H Moal, Mieczyslaw Torchala, Paul A Bates, and Juan Fernández-Recio. 2013. The scoring of poses in protein-protein docking: current capabilities and future directions. *BMC bioinformatics* 14, 1 (2013), 286.
- [25] I.S. Moreira, P.A. Fernandes, and M.J. Ramos. 2010. Protein-protein docking dealing with the unknown. *J. Comput. Chem.* 31, 2 (2010), 317–342.
- [26] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Courville, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.
- [27] J.C. Phillips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R.D. Skeel, L. Kale, and K. Schulten. 2005. Scalable molecular dynamics with NAMD. *Journal of computational chemistry* 26, 16 (2005), 1781–1802.
- [28] J. Qiu, W. Sheffler, D. Baker, and W.S. Noble. 2008. Ranking predicted protein structures with support vector regression. *Proteins: Structure, Function, and Bioinformatics* 71, 3 (2008), 1175–1182. DOI: <https://doi.org/10.1002/prot.21809>
- [29] T. Vreven, I.H. Moal, A. Vangone, B.G. Pierce, P.L. Kastiris, M. Torchala, R. Chaleil, B. Jiménez-García, P.A. Bates, J. Fernandez-Recio, and others. 2015. Updates to the Integrated Protein-Protein Interaction Benchmarks: Docking Benchmark Version 5 and Affinity Benchmark Version 2. *Journal of molecular biology* 427, 19 (2015), 3031–3041.
- [30] A. Wilkins, S. Erdin, R. Lua, and O. Lichtarge. 2012. Evolutionary trace for prediction and redesign of protein functional sites. *Methods Mol Biol.* 819 (2012), 29–42.