Multi-Scale Investigation of Protein-Protein Interactions

Hou, Q.

2017

document version
Publisher's PDF, also known as Version of record

4

Sequence-based Homo- and Heteromeric Protein-protein Interaction Sites Prediction Using Random Forest

Hou, Q., De Geest, P. F. G., Vranken, W. F., Heringa, J., and Feenstra, K. A. (2016). Seeing the Trees through the Forest: Sequence-based Homo- and Heteromeric Protein-protein Interaction sites prediction using Random Forest. *Bioinformatics*, 2017, btx005.

Abstract

Genome sequencing is producing an ever-increasing amount of associated protein sequences. Few of these sequences have experimentally validated annotations, however, and computational predictions are becoming increasingly successful in producing such annotations. One key challenge remains the prediction of the amino acids in a given protein sequence that are involved in protein-protein interactions. Such predictions are typically based on machine learning methods that take advantage of the properties and sequence positions of amino acids that are known to be involved in interaction. In this paper, we evaluate the importance of various features using Random Forest (RF), and include as a novel feature backbone flexibility predicted from sequences to further optimise protein interface prediction.

Results: We observe that there is no single sequence feature that enables pinpointing interacting sites in our Random Forest models. However, combining different properties does increase the performance of interface prediction. Our homomeric-trained RF interface predictor is able to distinguish interface from non-interface residues with an area under the ROC curve of 0.72 in a homomeric test-set. The heteromeric-trained RF interface predictor performs better than existing predictors on a independent heteromeric test-set. We trained a more general predictor on the combined homomeric and heteromeric dataset, and show that in addition to predicting homomeric interfaces, it is also able to pinpoint interface residues in heterodimers. This suggests that our random forest model and the features included capture common properties of both homodimer and heterodimer interfaces.

Introduction

Many biological events rely on proteins and their interactions. These events include signal transduction, metabolic processes, RNA transcription, DNA replication and protein transport [e.g., Jones and Thornton, 1996, Shoemaker and Panchenko, 2007a, Valencia and Pazos, 2002]. Protein functions are defined by their interactions and therefore understanding the interactions between proteins is of great importance to reveal the mechanism of protein functions and all biological processes [e.g., Shoemaker and Panchenko, 2007a, Valencia and Pazos, 2002].

Ideally, 3D structures of interacting proteins can be elucidated through experiments that provide atomic details of protein-protein interactions. However, due to the high cost of the required time-consuming experiments, only few high-resolution structures are available compared to all PPIs that govern protein functions. To gain a better understanding of PPIs and help the experi-

mental approaches, large amounts of effort have been devoted to develop computational methods for PPI sites prediction; protein-protein docking and modelling, structure-based interface prediction and sequence-based interacting sites determination methods [e.g., Schueler-Furman et al., 2005, Shoemaker and Panchenko, 2007b]. Docking, molecular dynamic simulations and other structure-based methods usually need structural details [Hou et al., 2016], whereas sequence-based approaches employ information derived from sequences which is becoming increasingly necessary due to the increasing amount of unannotated sequence data that is available [e.g., Gallet et al., 2000, Tuncbag et al., 2009].

PPI sites prediction methods usually rely on physiochemical, biophysical and statistical features extracted from sequence and/or structure that differentiate between interface and non-interface residues. A large number of studies have been focused towards identifying and implementing these features [e.g., Glaser et al., 2001, Jones et al., 2000, Murakami and Mizuguchi, 2010, Ofran and Rost, 2007a, Porollo and Meller, 2007]. For sequence-based PPI site prediction, several of the commonly used features are:

- Conservation: This has been widely and successfully used for interface prediction, and is commonly extracted from multiple sequence alignments (MSA) via (PSI-) BLAST [Altschul et al., 1997]. From this alignment a conservation score or a Position Specific Scoring Matrix (PSSM) can be calculated [Bradford and Westhead, 2005, Murakami and Mizuguchi, 2010, Ofran and Rost, 2007a]. Conservation of certain residues as a characteristic of binding is mostly based on the fact that there is evolutionary pressure on these residues for structural or functional purposes and therefore they mutate with lower frequency [Choi et al., 2009]. Some studies have found that conservation levels are not distinctive between interface and other surface residues [Caffrey et al., 2004]. Others did find interface sites to be more conserved than the rest [Carl et al., 2008, Choi et al., 2009], or distinct between interaction-specific sub-groups [Hou et al., 2015].

- Secondary structure: Studies have described that β-strands and loops are more common at interface regions [de Vries and Bonvin, 2008, Guharoy and Chakrabarti, 2007]. Since secondary structure prediction from sequence has become quite reliable [Heringa, 2000, Petersen et al., 2009], it has also been used in sequence-based PPI predictors [Li et al., 2012, Ofran and Rost, 2007a].

- Accessible surface area: The accessible surface area (ASA) is defined as the surface area of a protein that is accessible to the surrounding solvent. In essence, residues with high solvent accessibility are more likely to be interacting [Chen and Zhou, 2005, de Vries and Bonvin, 2008, Hoskins et al., 2006]. A number of studies have successfully used the accessible surface area as a feature, both as a structural characteristic and as feature predicted from sequence [e.g., Li et al., 2012, Ofran and Rost, 2007a].

Despite the progress that has been made in this area, there are still issues remaining when trying to predict interface residues from sequence. It is clear that the characteristics used up until now are not sufficient for prediction and most of the prediction methods focus on certain types of interfaces, for example, heteromeric interfaces between two different proteins [Murakami and Mizuguchi, 2010, Ofran and Rost, 2007a]. To gain a better prediction performance and to build a more general interface predictor, we include in this study several features which are novel or which have been rarely used before.

- Protein size: The fraction of interface residues versus all interface residues generally decreases with protein size [Chen and Zhou, 2005, De Vries and Bonvin, 2006]. A recent study [Martin, 2014] has shown that interface predictors may be biased when not accounting for this.
- Backbone flexibility: Proteins are often flexible and have a range of motion, especially when looking at intrinsically disordered proteins, but also in loop regions of more rigid proteins. Previous studies have shown that this flexibility is of importance for protein functions, including binding [Faber and Matthews, 1990, Wright and Dyson, 1999]. Guharoy and Chakrabarti [2007] have found importance of non-regular structures in forming the protein-protein interactions in heterocomplexes. Several studies have already used backbone flexibility and protein dynamics in the creation of PPI predictors with success [Bendell et al., 2014, Hirose et al., 2010, Li et al., 2012]. However, the studies are either structure-based [Li et al., 2012], small scale predictions [Hirose et al., 2010] or concern a certain type of protein interface [Bendell et al., 2014]. In this study, we implement the backbone flexibility score predicted by DynaMine [Cilia et al., 2013, 2014], which is the first direct predictor of dynamics from sequence.
- Sequence Specificity: Specificity of interactions has been reported and used for interface prediction when different subfamilies show varied binding characteristics. For example, Pirovano et al. [2006] have found specificity of interaction sites when comparing homologous subgroups that bind to different interaction partners. Hou et al. [2015] implemented sequence specificity to identify interacting sites with some success for a homodimer and monomer use-case.

In this paper, we first investigate the importance of different features by comparing the prediction performance using Random Forest. We show that the new feature (predicted backbone flexibility) can be used for interface prediction. After combining it with other ‘old’ features, we develop a sequence-based interface predictor trained on the homomeric dataset which can gain a better performance than other methods on homodimer interface residues prediction. When trained on the heteromeric dataset using the same features, the resulting predictor is also superior to other methods for heteromeric interface prediction. Finally, a more general predictor

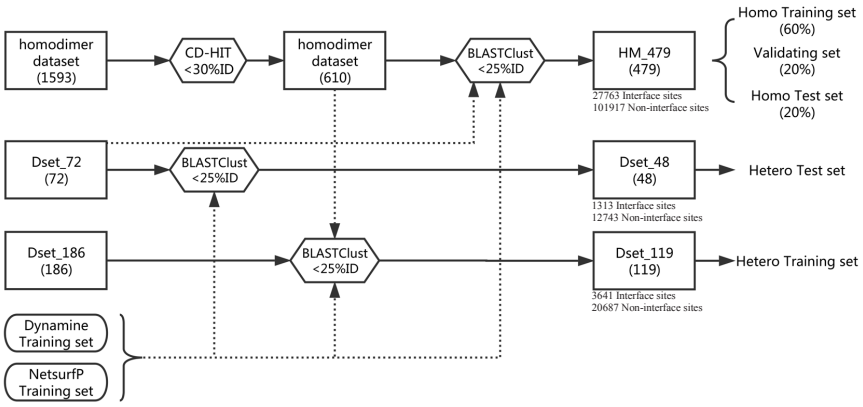

Figure 4.1: Process flow of constructing all data sets. Sequence Identify cut-off was used to remove the redundancy within the data sets. For example, starting with Dset_72 which is the test set of PSIVER, any sequences have more than 25% sequence identity over 90% overlap within Dset_72 and with the training sets of Dynamine and NetsurfP (the dotted arrow) are removed from Dset_72. Then, 48 sequences are retained and used as heteromeric test dataset.

trained on the combined heteromeric and homomeric dataset yields an even better performance on predicting both homomeric and heteromeric interfacers. This indicates that our predictor captures the common properties of both homodimer and heterodimer interfaces.

Methods

Datasets

To evaluate the importance of sequence features and develop an interface predictor, we took the homodimer interface dataset Test_set 1 (1593 homodimeric proteins) from Hou et al. (2015), and heterodimer datasets Dset_186 and Dset_72 from Murakami and Mizuguchi [2010] as our starting datasets. Figure 4.1 shows the process flow of constructing all data sets. CD-hit [Li and Godzik, 2006] was used to filter sequences in the homodimer dataset (1593 proteins) that have more than 30% sequence identity, retaining the set of 610 proteins forming heterodimers. Since we trained our models using homodimer and/or heterodimer proteins to predict both types of interfaces, redundant sequences among the training sets (both homodimer and heterodimer) and

test-sets should be removed. Since our predictors use features derived from predictions by NetsurfP and DynaMine, we also needed to remove redundancy with their training sets, as indicated in Figure 4.1. We collected the NetsurfP and DynaMine training sets from the respective authors. Any of the 610 homodimer proteins having more than 25% sequence identity over 90% of the sequence length with any protein in the training datasets of Netsurfp and DynaMine and the heteromeric test set Dset_72, were removed using BLASTClust [Altschul et al., 1990], retaining 479 sequences. This homomeric dataset is labeled HM_479. Dset_186 and Dset_72 were also filtered against the NetsurfP and DynaMine training datasets using BLASTClust with the same thresholds. In order to predict homomeric proteins from HM_479, Dset_186 was also filtered against the 610 homodimer proteins. We so obtained the heteromeric training dataset with 119 proteins labeled Dset_119 and heteromeric test dataset with 48 sequences with heteromeric interactions named Dset_48.

To validate the performance of each sequence feature to predict interacting sites, we split our HM_479 dataset into three parts: 60% for training, 20% for validating the models and 20% for testing the best model. The heteromeric training set Dset_119 was applied to evaluate the ability of our features on prediction of heteromeric interfaces and also combined with homomeric training sets from HM_479 to make a more general predictor which is expected to pinpoint both the homomeric and heteromeric interfaces. Dset_48 was employed as an independent heteromeric test set.

Alignments

For proteins in HM_479, the alignments built previously [Hou et al., 2015] were used, each containing one interacting (homodimer) subgroup and one non-interacting (monomer) subgroup. MUSCLE [Edgar, 2004], a fast alignment method, was used to make the multiple sequence alignments. For further details on the construction of this dataset, please refer to Hou et al. [2015]. For each sequence in Dset_119 and Dset_48, PSI-BLAST [Altschul et al., 1997] was used to obtain homologous sequence hits from the NCBI non-redundant protein sequence database with an e-value threshold 1E-3. Maximally 500 sequences for each query sequence were used to build sequence alignments using MUSCLE [Edgar, 2004] .

For each sequence in Dset_119 and Dset_48, PSI-BLAST [Altschul et al., 1997] was used to obtain homologous sequence hits from the NCBI non-redundant protein sequence database with an e-value threshold 1E-3. Maximally 500 sequences for each query sequence were used to build sequence alignments using MUSCLE[Edgar, 2004] .

Definition of Interfaces

Interface residues were defined by the criterium that both Accessible Surface Area (ASA) before association and Buried Surface Area (BSA) during the association are larger than 0 $Å^2$. The Accessible Surface Area and Buried Surface Area were obtained from the PISA database [Krissinel and Henrick, 2007].

Random Forest

Random Forest, an ensemble learning method, uses a tree based classification system where multiple classification trees form a 'forest'. Starting with a randomly selected subset from the original dataset, the concept of one classification tree is constructed by re-iteratively partitioning the data space into smaller sub-divisions until we come at a point where these sub-divisions are fit to the selected subsets. For each node within each tree, a randomly selected subset of the input variables is used. The number of variables randomly sampled is defined by one global parameter called 'mtry' [Liaw and Wiener, 2002]. We used a grid search over a set interval of values from 1 to 50 for 'mtry'. We then selected the optimal value of 'mtry' with the best Area Under the Curve (AUC) of the receiver operator characteristic (ROC) curve on the training set.

Classification is done for a new sample (a residues) by majority vote over all trees involved, is chosen over all trees involved. In this paper, 500 trees are used to obtain the classifier. The implementation from the Random Forest R-package was used [Liaw and Wiener, 2002]. We used several measures to evaluate the performance of our classification as follows:

- Area Under the Curve (AUC) of the receiver operator characteristic (ROC; true positive rate vs. false positive rate) plot
- Recall (True Positive Rate, Sensitivity or Coverage) = TP / (TP+FN)
- Precision (Positive Predictive Value) = TP / (TP+FP)
- Specificity = TN / (TN+FP)
- Accuracy = (TP+TN) / (TP+FN+TN+FP)
- F1 = 2 $\times$ (Precision $\times$ Recall) / (Precision + Recall)

The method described in DeLong et al. [1988] was used to compare two ROC plots for the heteromeric interface sites prediction. This was done using 'roc.test' function in R library 'pROC'[Robin et al., 2011]. True positives (TP) are correct prediction of interacting sites,

false positives (FP) are false prediction of interface residues, true negatives (TN) designate non-interacting sites that the method recognised, and false negatives (FN) are true binding sites which our approach missed.

The cross-validation and Random Forest parameter tuning were done by using with the caret R-package [Kuhn, 2015]. Feature importance was measured by using MeanDecreaseGini function in the R 'caret' package, which measures variable importance based on the Gini impurity index [Kuhn, 2015].

To obtain a reliable and stable prediction, we resampled the training and validating datasets using random seeds. To evaluate the importance of different features, the average performance of the ten models from ten-fold resampling on the validating set was used. For the RF_homo, RF_hetero and RF_combined predictor, five-fold resampling was used.

The features used for training RF models are all sequence-based features (details can be seen below).

Sequence Features

Conservation: Sequence entropy [Sander and Schneider, 1991, Shenkin et al., 1991] is used to describe the degree of conservation. For each column i in the alignment, the entropy is calculated as $S_i = -\sum_x p_{i,x} \log p_{i,x}$, where $p_{i,x}$ represents the fraction of amino acid x at the i-th position of the sequence; the sum is done over all 20 type of amino acids. Positions which have lower entropy scores indicate higher conservation rate. 'H-entropies' used in later analysis means the S_i calculated over the column of the homodimer subgroup.

Specificity: Sequence Specificity Score (SH) was calculated as:

$$SH_i = -\frac{1}{2}\sum_x P^H_{i,x} \log \frac{P^H_{i,x}}{P^H_{i,x} + P^M_{i,x}} - \frac{1}{2}\sum_x P^M_{i,x} \log \frac{P^M_{i,x}}{P^H_{i,x} + P^M_{i,x}} \tag{4.1}$$

where $P^H_{i,x}$ means the frequency of amino acid type x at position i in the homodimer group H, and analogously for monomer group M, and the sum is over all 20 amino acids [Pirovano et al., 2006]. The Sequence Specificity Scores were calculated between the interacting (Homodimer) and non-interacting (Monomer) sub group for each position in the alignment. Lower SH scores indicate positions with high specificity between homodimer and monomer groups.

Sequence length and HSP length: Sequence length is the number of residues in the query homo/hetero dimer protein whose structures will be used to evaluate prediction performance. High scoring Segment Pair (HSP) length is obtained from BLAST between the homodimer and

its first homologous monomer hit (details can be found in Hou et al. [2015]). Longer HSP length reflects stronger evidence for a homologous relation. HSP length is only used in combination with Sequence Specificity (SH score) to predict interacting sites.

Backbone Dynamics: DynaMine, the first direct predictor of dynamics from sequence, was used to obtain the backbone dynamics properties for each residue in the sequence [Cilia et al., 2013, 2014]. DynaMine uses a linear regresson model to predict backbone N-H S^2 order parameter values which were derived from chemical shift values. DynaMine needs a single sequence as input and gives each residue a predicted score (ranging from 0 to 1) reflecting the degree of backbone flexibility. Lower DynaMine scores indicate that the backbone is more flexible [Cilia et al., 2013, 2014]. For each column in the multiple sequence alignment, the average DynaMine score (DM) and the standard deviation of scores (dm_std) were calculated.

Accessibility and Secondary Structure: Absolute and Relative Surface Accessibility (ASA and RSA) describing the solvent accessibility of the amino acid were calculated by NetsurfP [Petersen et al., 2009]. The probability score of secondary structures was also predicted by NetsurfP. NetsurfP consists of two neural network for prediction. The primary network implements PSSM and predicted secondary structure as inputs to classify positions as buried or exposed. Then, the secondary networks use the classification predicted in the first step and PSSMs to predict RSA and ASA. Three vectors for RSA are used in this research: the RSA value for the query sequence (rsa_q), the mean of RSA values over the alignment (mean_rsa) and the standard deviation for each column (std_rsa). Similar to RSA, for ASA, the query (asa_q), the average (mean_asa) and the standard deviation (std_asa) are also used. Nine variables for predicted secondary structure are implemented: α helix (pa_q, mean_pa and pa_sd), β sheet (pb_q, mean_pb and pb_sd) and the coil (pc_q, mean_pc and pc_sd). Like before, q, mean and sd refer to the values for query, average and standard deviation respectively.

Sliding window approaches

As the propensity for protein interactions is not a local property of single amino acid positions, it is expected that using the characteristics and patterns of neighbouring residues could be helpful in the prediction of the interface residues. In this study, we implement a fixed 9 residue sliding window with all features (n) as a feature vector (9n). Thus, for each residue position i, values of features at position i-4, i-3, i-2, i-1, i, i+1, i+2, i+3, i+4 are considered and used as input features. A feature value is set to 'NA' when the positions in the windows are not available (i.e. at start and end of the sequence).

Table 4.1: Interface prediction performance using two new features

Feature(s)	Training set	Testing set	Accuracy	Sensitivity	Precision	Specificity	F1	AUCROC	AUCROC_std
En	HM_479 training	HM_479 validating	0.755	0.073	0.232	0.936	0.111	0.480	0.009
SH	HM_479 training	HM_479 validating	0.738	0.097	0.217	0.908	0.134	0.495	0.008
DM	HM_479 training	HM_479 validating	0.739	0.106	0.229	0.906	0.144	0.506	0.006
SH + len +hsp	HM_479 training	HM_479 validating	0.789	0.005	0.311	0.997	0.010	0.537	0.014
DM + len + win	HM_479 training	HM_479 validating	0.790	0.043	0.430	0.985	0.078	0.578	0.008
SH + len +hsp +win	HM_479 training	HM_479 validating	0.789	0.005	0.311	0.997	0.010	0.537	0.009
EN+ SH +DM +len +win	HM_479 training	HM_479 validating	0.794	0.012	0.603	0.998	0.023	0.587	0.012
EN+ DM +len +win	HM_479 training	HM_479 validating	0.797	0.037	0.647	0.995	0.070	0.616	0.015

'EN' means the conservation score over the homodimer sub-group; 'DM' represents the average DynaMine score over all positions in the homodimer sub-group; 'win' means all features are windowed (except the whole-group features length and hsp length). All results are the average over 10-fold re-sampling of the training and validating sets.

Comparison with other methods

For the homomeric interface prediction, we compare our method with PSIVER [Murakami and Mizuguchi, 2010] and SPPIDER [Porollo and Meller, 2007]. PSIVER, a Naïve Bayes classifier using PSSM and predicted RSA as features, was published as a sequence-based interface predictor. SPPIDER uses alternative machine learning methods and also implements predicted RSA as feature.

For heteromeric interface prediction, we compare with PSIVER, which was already shown to be the best predictor on its test set dset_72 [Murakami and Mizuguchi, 2010]. Recall, Precision, F1 score and AUC of ROC plot were calculated to compare between methods.

Results

We first test the ability of two new features, Sequence Specificity and DynaMine score, to predict protein interacting sites. We trained our model based on the homodimer training set. To evaluate the performance, we compare these two features with the most widely used property, Conservation (Sequence Entropy). All performances are measured on the validation dataset after acquiring the RF model from the training dataset.

Predicted backbone flexibility score can be used to predict interacting sites

Table 4.1 shows the performance of using single or combined features to predict interacting residues (more combinations can be seen in SI Table C.1). The Area Under the ROC curve (AUC) shows how well single and combined features work to predict interfaces. To confirm

the comparisons among the features are stable and non-random, all results are averaged over ten-fold resampling the training and validating datasets (see Methods). The standard deviations of all AUCs are also shown in the table.

The AUC is not significantly different than random (0.5) when using only one feature in the Random Forest model. This observation shows that none of these single features derived from protein sequence is sufficient to predict protein interface region correctly. In combination with sequence length and hsp length, we can get slightly better performance than a single feature, which is similar to what we showed previously for Sequence Specificity [Hou et al., 2015]. Interestingly, the AUC only using windowed DynaMine score together with sequence length can already reach around 0.578. The 'mean_dm' DynaMine score used here the averaged value per column in the alignments. Finally, when combining the new features (SH and DM) with the sequence length and conservation, the performance improves a bit more (AUC of 0.587). We obtained the highest performance (AUC 0.616) using windowed sequence entropy, windowed DynaMine score and length. From Table 4.1, we can conclude that applying a single sequence property is insufficient to predict interface positions. Of the two new features, DynaMine score can be used for interface prediction when windowed and combined with other simple features.

Constructing Random Forest Model Integrated all features

In the first part, we showed that combination of different features can improve the prediction performance. Now we further include several commonly used structural features predicted from sequence: secondary structure, Solvent Accessible Surface Accessibility (ASA) and Relative Surface Accessibility (RSA). These features are predicted using sequence information by the NetsurfP server [Petersen et al., 2009].

From Table 4.1 we observe that only using sequence specificity (SH score) together with sequence length and HSP length can help identify interacting sites. Sequence specificity is the only feature derived from both interacting and non-interacting sequences, while all other features are obtained from only the interacting group. To make a more general interface predictor without using non-interacting information, we therefore decided not to use sequence specificity for later feature combinations.

Figure 4.2 shows the ROC plots which measure the prediction performance for different combinations of features. As before, the plots are averaged over the ten resampled replicate training and validating datasets.

We start with the three features from the last section: entropy, DynaMine score and sequence

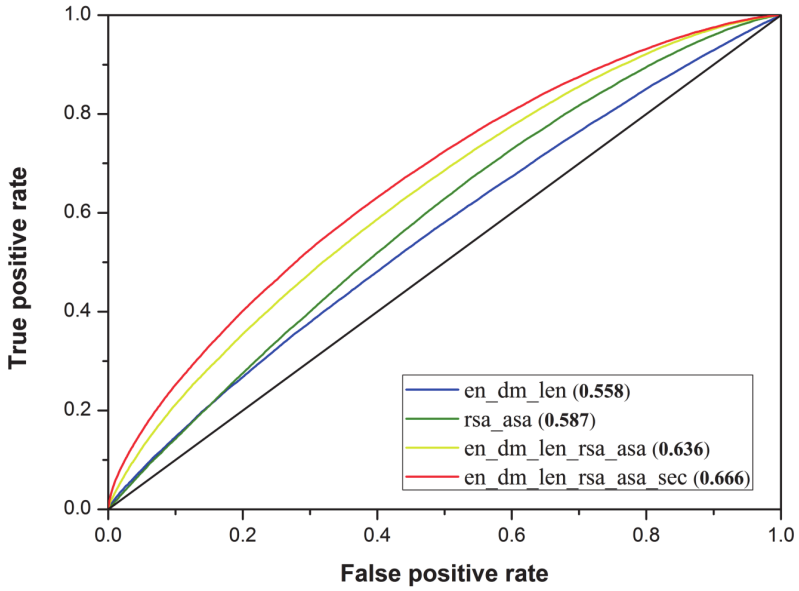

Figure 4.2: Performance of different combined features to predict interfaces ROC plots are used to compare the performance on the validating datasets. Each line is averaged by ten replicates simulations which the training and validating datasets are resampled. 'en', 'dm', 'len', 'rsa' and 'asa' stand for entropy, DynaMine score, length, relative surface accessibility and absolute surface accessibility, respectively.

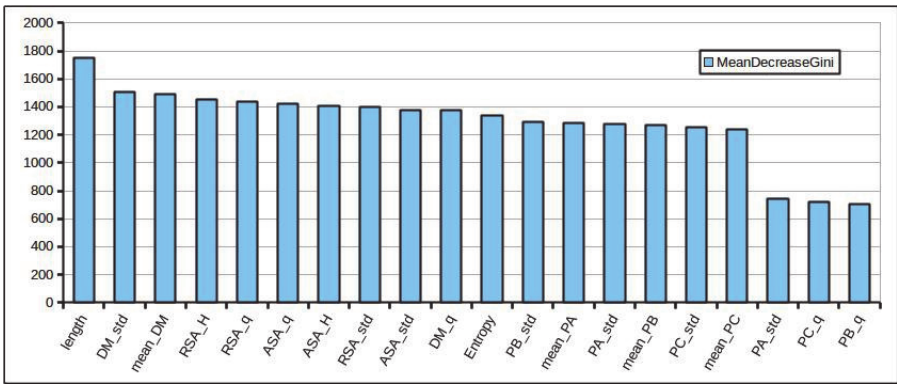

Figure 4.3: The ranking of feature importance for the model which integrate all features in Figure 4.2. The importance was measured by 'Mean Decrease Gini' in 'caret' package. The features are more important from the left to right. The most important feature is 'length' of query sequence when all features are used.

length (AUC 0.558). Before combining the features with ASA and RSA, we also showed the prediction performance only using ASA and RSA (AUC 0.587). The predictions becomes better when combined with ASA and RSA (AUC 0.636), and adding secondary structure even brings further improvement (AUC 0.666). It is clear that the prediction performance increases with more features combined. That also proves that all these features are useful for interface prediction. For the highest performing model (the one including all features), the 20 features that were used are: Dynamine score: dm_q, mean_dm and dm_sd; RSA: rsa_q, mean_rsa and rsa_sd; ASA: asa_q, mean_asa, and asa_sd; secondary structure: α helix (mean_pa, pa_q and pa_sd), β sheet (pb_q, mean_pb and pb_sd) and coil (pc_q, mean_pc and pc_sd); Entropy (en); length of query sequence (len). By including the average value and standard deviation of features at each column, we introduce the evolutionary and protein family information into our prediction model.

We also evaluate the contribution of each single feature to the best model without windowing approach. Figure 4.3 shows the importance of each feature for training the random forest model. Interestingly, the length of the sequence is the leading contribution to the interface prediction when combining all features. The backbone flexibility predicted by DynaMine, both the average and standard deviation, are two of the most important features.

Table 4.2: Performance comparison among different features

Features	Training set	Testing set	Accuracy	Sensitivity	Precision	Specificity	F1	AUCROC	AUCROC_std
all features	HM_479 training	HM_479 validating	0.790	0.025	0.487	0.992	0.047	0.666	0.008
all features + win	HM_479 training	HM_479 validating	0.795	0.016	0.896	0.999	0.032	0.710	0.011
all features + win	balanced HM_479 training	HM_479 validating	0.688	0.614	0.355	0.707	0.450	0.728	0.008
all features + win	balanced HM_479 training	HM_479 testing	0.695	0.581	0.373	0.722	0.454	0.720	0.007

'win' indicates windowing approach is used; 'balanced' means balanced training dataset (I:NI=1:1).

Predicting homomeric interfaces

After obtaining the model which simply combines all features (AUC 0.666 in Figure 4.2), we include the neighbour information in a window to further improve the prediction. We windowed all the features using 9-residue windows (except sequence length which is a global feature). Then, we trained our Random Forest model using the windowed features. All results are averaged over five-fold resampled training and validation sets. Table 4.2 shows that the AUC with windowed data reaches 0.710, a major performance increase compared to non-windowed data (0.666).

Our 'all features+win' model can separate interaction sites and non-interacting sites very well, with a particularly high Precision (0.896). However, these results show a very low sensitivity of 0.016 (Table 4.2), which translates into about one residue ($27763 \div 479 \times 0.016 = 0.928$, see number in Figure 4.1) predicted on average per protein. Furthermore, high specificity arises from a high imbalance between classes [Murakami and Mizuguchi, 2010, Šikić et al., 2009]. In our dataset, the ratio between interface and non-interface residues is 0.21. In order to counter this problem concerning imbalanced data we downsampled our training dataset to decrease the specificity, as suggested by Chen et al. [2004] and Lin and Chen [2013]. We randomly sampled the majority class (non-interacting positions) to the same frequency (1:1) as the minority class (interacting positions). As a double check, we also confirmed that upsampling, i.e. replicating the minority class to the same frequency as the majority class, did not lead to better classification (data not shown). Table 4.2 shows that after balancing, the sensitivity increases considerably from 0.025 to 0.614, while the specificity moderately decreases from 0.999 to 0.707, which indicates downsampling the training set helps to overcome the imbalance. This finding is consistent with other studies [Dhole et al., 2014, Šikić et al., 2009]. At the same time, the overall prediction performance as represented by the AUC-ROC increased to 0.72. Trained on balanced data, our model now obtains an AUC of 0.728 for the validation dataset which is an excellent prediction performance in this field. The performance remains stable when measured on the separate test

set, showing performance to be rather stable (AUC 0.720 vs. 0.728). From these results, we can conclude that our Random Forest model (named RF_homo) is a well-performing homomeric interface predictor.

We compare RF_homo with two sequence based methods, SPPIDER and PSIVER, using ROC curve and Precision-Recall plots, shown in Figure 4.4. Both SPPIDER and PSIVER are sequence-based interface prediction methods using machine learning approaches. The RF_homo predictor outperforms the other two methods in both ROC plots (Figure 4.4A) and Precision-Recall plots (Figure 4.4B). The asterisk in the plots indicates the data point using the predictors' default thresholds (the values can be seen in Table 4.3). Our random forest models implement the probability score 0.5 as a cut-off and positions with scores higher than 0.5 will be identified as interacting sites. For SPPIDER and PSIVER, we use the default cut-offs provided by their websites. For homodimer interface prediction, the RF_homo predictor obtains a higher precision than SPPIDER and PSIVER at any coverage. In addition, RF_homo also obtains higer MCC, F1 and AUC scores than the other two methods (Table 4.3). We can conclude that when predicting *homodimeric interfaces*, our predictor trained on the homomeric dataset is much better than other predictors using sequence information alone.

Predicting heteromeric interfaces

To evaluate our Random Forest model as a more general predictor other than only predicting one certain type of interface, we apply RF_homo predictor to the independent extended dataset dset_48, consisting solely out of *heteromeric complexes*. Figure 4.5 shows that PSIVER obtains a higher precision (0.2) at very low recall (0.02) than RF_homo predictor. But for most of the recall range (0.06-0.76), RF_homo has a better precision than PSIVER particularly for a recall up to 0.4. Moreover, our homomeric predictor also obtains an AUC in ROC similar to PSIVER (0.619 vs 0.613, p-value 0.323). Based on these results, the RF_homo predictor performs similar to PSIVER, which is surprising given that our homomeric predictor is trained on *homomeric* interactions while PSIVER was trained on *heteromeric* data.

To further investigate our approach to predict heteromeric interfaces, we also trained our model on the heteromeric training set Dset_119. Downsampling and windowing approaches are applied to Dset_119 in the same way as for our homomeric training dataset. Figure 4.5 shows the comparison of precision-recall and ROC curves between our heteromeric predictor (RF_hetero) and PSIVER (the blue line) on this dataset. Our heteromeric predictor (the red line) obtain higher AUC (0.652 vs 0.613) than PSIVER, which indicates our RF_hetero predictor is better

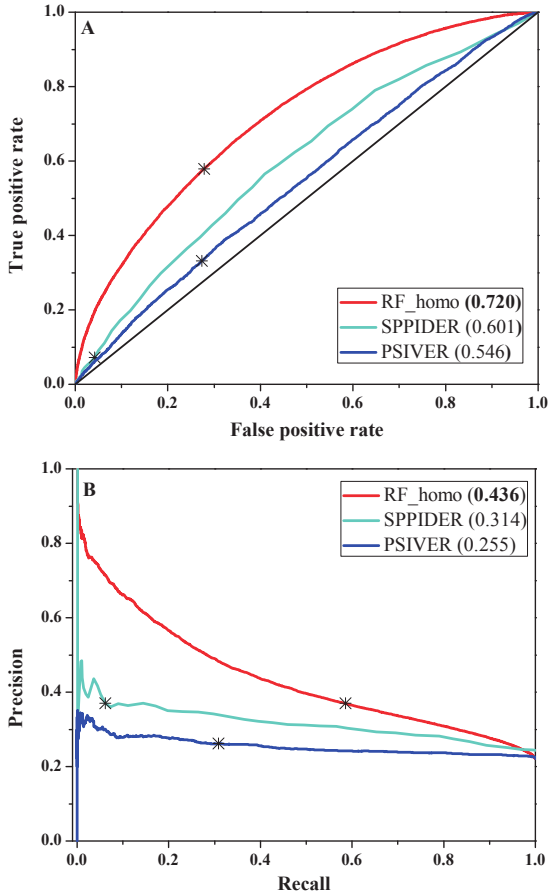

Figure 4.4: Comparison of performance between our Homo predictor, PSIVER and SPPIDER on homodimer dataset. ROC plots and Precison-Recall plots are used to measure performance of interface prediction. Obviously, our predictor performs much better than other approaches when used to predict homodimer interfaces. Star '*' in the figures means the data point for each methods using their own default thresholds.

Table 4.3: Performance comparison amog predictors using different training datasets

Predictor	Training dataset	Test dataset	Recall	Precision	Specificity	MCC	F1	AUCROC	
RF_homo	HM_479 training	HM_479 testing	0.581	0.373	0.722	0.265	0.454	0.720	
RF_hetero	Dset_119	HM_479 testing	0.343	0.263	0.727	0.064	0.297	0.552	
RF_combined	HM_479 training+Dset_119	HM_479 testing	0.581	0.383	0.734	0.277	0.462	**0.724**	
SPPIDER	homo+hetero†	HM_479 testing	0.073	0.361	0.958	0.062	0.121	0.601	
PSIVER	Dset_186	HM_479 testing	0.315	0.262	0.743	0.054	0.286	0.546	
RF_homo	HM_479 training	Dset_48	0.446	0.140	0.716	0.103	0.213	0.619	
RF_hetero	Dset_119	Dset_48	0.547	0.146	0.667	0.131	0.230	**0.652**	* **, *
RF_combined	HM_479 training+Dset_119	Dset_48	0.500	0.146	0.695	0.122	0.226	0.636	** *
PSIVER	Dset_186	Dset_48	0.668	0.119	0.493	0.094	0.203	0.614	

† SPPIDER uses its own training dataset; The training datasets for our predictors are balanced and windowed datasets; Highest AUCROC per test set is indicated in bold. ROC plots significantly different at *p<0.01 or at ** p<0.001, details can be seen in SI Table C.3.

on predicting heteromeric interfaces than PSIVER. Moreover, it shows that the features selected from our homomeric dataset can be also applied to train a heteromeric interface predictor.

Meanwhile, when predicting heteromeric interfaces, our heteromeric predictor obtains a better performance than the homomeric predictor and the homomeric model is superior on the homomeric interfaces. It seems that the predictors perform better on the same types of interfaces.

To develop a more general predictor which is able to predict both types of interfaces, we combine the homomeric HM_479 and heteromeric Dset_119 training datasets to train a new predictor RF_combined. Table 4.3 shows the performance of the new predictor trained on the combined dataset and compared to other methods. On the heteromeric interface test set, the RF_combined predictor performs slightly lower than the RF_hetero, as may be expected, but still better than PSIVER (AUC of ROC 0.634 vs 0.613, p-value 0.01, details in Table 4.3). Interestingly, on the homomeric interface test set the RF_combined prediction even outperforms the RF_homo predictor. Our RF_combined predictor therefore has the best overall performance, and constitutes a significant improvement over the current state of the art.

Discussion

Sequence-based interface prediction remains a challenging problem. There is no single property from sequence that can predict interacting sites correctly. In this paper, we show that two new features (Sequence Specificity and backbone flexibility predicted from sequence) can be used for interface prediction. When combining the new feature (backbone flexibility) with several 'old' features, we get an attractive prediction performance on both homomeric and heteromeric interfaces. Interestingly, sequence length is one of our most important features. Although a

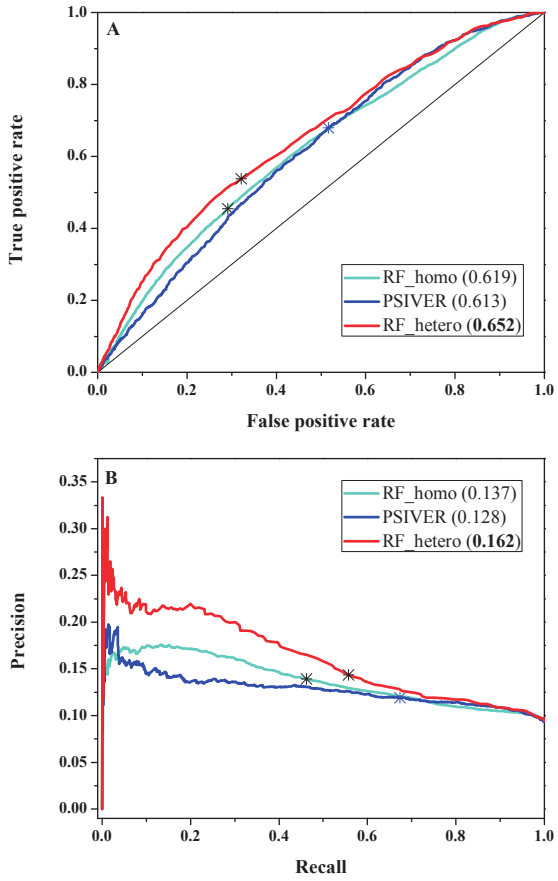

Figure 4.5: Comparison of performance between our methods and PSIVER on the heterodimer testset. ROC plots and Precison-Recall plots are used to measure performance of interface prediction. Star '*' in the figures means the data point for each methods using their own default thresholds.

global feature by itself obviously can not pinpoint local sites, the RF decision points may nevertheless benefit from this 'global' information, as indicated by the feature importance in Figure 4.3 and from comparing feature sets with and without 'length' in SI Table C.1. The global feature might allow the predictor to do background normalization for local features.

One may assume therefore that we only predict well for proteins of a certain length. To test this, we checked whether there is a protein size bias towards our prediction. This was done on our RF_homo predictor and HM_479 dataset. We mapped the performance (AUC of ROC) to the query sequence length for each group for our five models. The datasets used here are the validation and test sets of five models (SI Figure C.1 a-e). SI Figure C.1 shows that there is no detectable correlation between our prediction and sequence length which indicates that we do not add any bias to our prediction when using sequence length as a feature. Interestingly, in recent work (Abeln, Van Dijk, Bouwmeester, Heringa; personal communication), it was shown that using sequence length can also help improve the prediction performance of the hydrophobic surface area.

The definition of interface in PSIVER and our paper are slightly different. In PSIVER, a threshold for Buried Surface Area (BSA) during the association of $> 1\text{Å}^2$ is used. Here, we use the lower cut-off ($> 0\text{Å}^2$) from the PISA database. There is barely a difference when we compared the interacting sites of Dset_72 and Dset_186 defined by PSIVER and by PISA, so we expect this does not influence our performance assessment.

Our homodimer dataset is extraced from PISA and PDB and might contain bias from the PDB. To check this is not an issue, we mapped the sequences of the validating and testing set in HM_479 onto CATH superfamilies. Per superfamily we calculated the AUC of ROC. The datasets are the validation and test sets, as used in SI Figure C.1. SI Figure C.2 shows the average of the AUC per CATH superfamily for all five models. For example, in model 1, our 191 sequences belong to 105 superfamiles which cover all four main classes, 52% (21/40) architectures and 5.9% (81/1375) topologies. That indicates our method does not only predict superfamiles that are enriched in PDB but cover the full fold known space. From SI Figure C.2, it is clear that the superfamily with most proteins predicted is not even our best-performing one. On the contrary, the average AUC varies and has no correlation with the size of the superfamiles.

Conclusion

In this paper, we show that predicted backbone flexibility can be used for interacting sites prediction. After integrating with other features, our predictor shows an excellent performance on

the prediction of both homomeric and heteromeric interfaces which is better than other methods. That indicates our predictor captures the common properties of interfaces in both homodimers and heterodimers.

Acknowledgements

The authors declare that there is no conflict of interest. Qingzhen Hou is supported by Chinese Scholarship Council (No.2011627127). The authors thank Bent Peterson for sharing the training dataset of NetSurfP and Ted Meeds for constructive discussions. The authors thank the anonymous reviewers for their constructive remarks, and in particular for their suggestion to use the heteromeric training set, which greatly contributed to this paper.