BIOINFORMATICS

Vol. 00 no. 00 201X
Pages 1–7

Contact map prediction using a large-scale ensemble of rule sets and the fusion of multiple predicted structural features

Jaume Bacardit [1], Paweł Widera [1], Alfonso Márquez-Chamorro [2], Federico Divina [2], Jesús S. Aguilar-Ruiz [2] and Natalio Krasnogor [1*]

[1]ICOS research group, School of Computer Science, University of Nottingham, [2]School of Engineering, Pablo de Olavide University of Sevilla, Spain

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

ABSTRACT

Motivation: The prediction of a protein's contact map has become in recent years a crucial stepping stone for the prediction of the complete 3D structure of a protein. In this paper we describe a methodology for this problem that was shown to be successful in CASP8 and CASP9. The methodology is based on (1) the fusion of the prediction of a variety of structural aspects of protein residues, (2) an ensemble strategy used to facilitate the training process and (3) a rule-based machine learning system from which we can extract human-readable explanations of the predictor and derive useful information about the contact map representation.

Results: The main part of the evaluation is the comparison against the sequence-based contact prediction from CASP9, where our method presented the best rank in five out of the six evaluated metrics. We also assess the impact of the size of the ensemble used in our predictor to show the trade-off between performance and training time of our method. Finally, we also study the rule-sets generated by our machine learning system. From this analysis we are able to estimate the contribution of the attributes in our representation and how these interact to derive contact predictions.

Availability: http://icos.cs.nott.ac.uk/servers/psp.html

Contact: natalio.krasnogor@nottingham.ac.uk

1 INTRODUCTION

Contact Map (CM) prediction is one of the most challenging problems within the field of protein structure prediction (PSP). This is due to the sparseness of the contacts (i.e. the positive examples) and the large training sets (millions of instances, GBs of disk space) that are generated by using just a few thousands of proteins. CM can provide crucial information for improving PSP methods in a variety of ways: providing restraints candidate conformations (Zhang, 2009), reconstructing approximate 3D structures from the CM (Vassura *et al.*, 2008) or selecting good models (Tress and Valencia, 2010).

Most CM prediction methods use a sequence-based approach using machine learning methods. Through the years many techniques have been applied to CM prediction, such as neural networks (Shackelford and Karplus, 2007; Punta and Rost, 2005), support vector machines (Cheng and Baldi, 2007), genetic programming (MacCallum, 2004) or random forests (Li *et al.*, 2011). Moreover, many sources of information can be used for CM prediction. Beside evolutionary information, used by all methods, some use predicted secondary structure (SS), predicted solvent accessibility (SA), correlated mutations, contact propensity, statistics over the connecting segment between the pair of target residues or global protein information. The diversity of information sources as well as the fact that CM datasets can easily reach millions of residue pairs requires the use of methods that can cope with both large instance sets and high dimensionality spaces.

This paper introduces the prediction methodology with which we have participated in the last two editions of CASP under the name *Infobiotics*. The main characteristics of this predictor are (a) an ensemble architecture designed to alleviate the sparseness and large training set sizes of the CM problem, (b) the fusion of several predicted 1D structural features. Beside the usual SS and SA we also use the less frequently used Coordination Number (CN) (Kinjo *et al.*, 2005) and our own 1D metric called Recursive Convex Hull (RCH)(Stout *et al.*, 2008), which models the degree of burial of an amino-acid within a protein by modelling a protein's structure as a series of nested convex hulls and assigning each residue to a certain hull, and (c) a robust genetic algorithms-based rule learning system called BioHEL(Bacardit *et al.*, 2009b) (http://icos.cs.nott.ac.uk/software/biohel.html) that has been designed to cope with both large numbers of instances and large dimensionality spaces and has been successfully applied across a broad range of bioinformatics problems (Stout *et al.*, 2008; Bacardit *et al.*, 2009a; Stout *et al.*, 2009; Bassel *et al.*, 2011).

We assess the prediction capacity of our method firstly on a set of 3262 non-redundant protein chains and afterwards on the CASP8 and CASP9 free modelling targets. Finally we compare the performance of our method against the top sequence-based methods in CASP9, showing that our method is very competitive, being the top ranked sequence-based method in most metrics. We also assessed the influence of the size of the ensemble architecture on its performance. Finally, the added benefit of using a rule-based machine learning system such as BioHEL is that we can study the human-readable solutions it produces to understand *how* the system predicts, identifying which attributes are the most useful and how they interact.

*to whom correspondence should be addressed

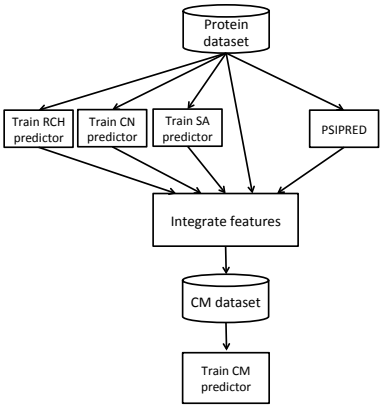

Fig. 1. General architecture of the contact map predictor

2 MATERIALS AND METHODS

Our CM prediction architecture integrates four types of complementary 1D predictions of structural aspects of protein residues: SS, SA, CN and RCH. These predictions together with information derived from the primary sequence are integrated to create the full CM dataset from where our prediction model is trained. This architecture is represented in figure 1.

Protein chains were selected from PDB-REPRDB, a non-redundant curated subset of the Protein Data Bank (PDB) (Noguchi *et al.*, 2001), covering the space of possible folds. Chains were selected using the following criteria: less than 30% sequence identity, sequence length greater than 50 residues, no non-standard residues, no chain breaks, resolution smaller than 2Å and crystallographic R factor smaller than 20%. PDB entries that had been used in the CASP8 Free Modelling category were removed from the training set as these will be used for the evaluation of the CM predictor. 3262 protein chains were selected with a total of 637494 residues. 90% of the set was used for training and 10% for test. For clarity we will refer to this protein set as *CM-3262* in the rest of the manuscript. The lists of proteins used for training and test is available in the supplementary material of the paper.

The complete training set was used to generate the predictors of CN, SA and RCH. For efficiency reasons, we thinned out the sets of proteins used to train and test the CM predictor. We kept all proteins with less than 250 residues and a randomly selected 20% of larger proteins, resulting in a training set of almost 32M pairs of amino acids (using a minimal chain separation of 6; small separation, to generate a large number of residue pairs) and a test set of 2.8M pairs of amino acids (using a minimal chain separation of 24; as used in CASP to assess CM prediction). Overall less than 2% of all amino acids pairs were real contacts at the usual distance threshold of 8Å.

2.1 Prediction of 1D structural features

For the prediction of SS we have used PSIPRED (Jones, 1999) and hence its 3-state representation of SS (helix, strand or coil). We have generated predictors for the other three metrics using the same system (BioHEL) as for the CM predictor. We describe the three metrics and how these are predicted.

2.1.1 Coordination Number The CN of a certain amino acid is the number of spatial neighbours of the residue within a specified distance threshold. We have used the CN definition proposed by Kinjo et al. (Kinjo *et al.*, 2005). It is defined using the C_β atom (C_α for glycine) of each residue. The boundary of the sphere around a residue, defined by the distance cutoff $d_c \in \Re^+$, is made smooth by using a sigmoid function. A minimum chain separation of two residues is required. Formally, the CN, N_i^p, of residue i in protein chain p is computed as:

$$N_i^p = \sum_{j:|j-i|>2} \frac{1}{1 + exp(w(r_{ij} - d_c))} \quad (1)$$

where r_{ij} is the Euclidean distance between the C_β atoms of the ith and jth residues. The constant w determines the sharpness of the boundary of the sphere. In this paper we used a distance cutoff d_c of 10Å (which gives higher predictability than 8Å (Bacardit *et al.*, 2006)) and a w of 3.

2.1.2 Solvent Accessibility Following (Rost and Sander, 1994) we predict the *relative* SA, where the SA of a residue is divided by the maximum accessible surface in the extended conformation of its amino-acid (AA) type. DSSP (Kabsch and Sander, 1983) is used to obtain the absolute SA of each residue in the dataset. Next, to obtain the relative SA values we divide the absolute values by the maximum SA values specified for each AA type in (Rost and Sander, 1994).

2.1.3 Recursive Convex Hull RCH (Stout *et al.*, 2008) is a metric that aims at assessing the degree of burial of a residue within the core of a protein. This is achieved by modelling the topology of a protein structure using the well defined geometry concept of convex hull. The convex hull (Preparata and Shamos, 1985) of a set of points X is the minimal convex set containing X where a set is said to be convex if, for every pair of points within the set, all points on the line segment joining these two points are also within the set. As in CN, residues are represented by the position of their C_β atoms (C_α for glycine). Convex hulls for each chain were identified from the residue C_β atom coordinate point sets using the QHull package (Barber *et al.*, 1996). Hulls were iteratively identified, surface residues were assigned a hull number and then removed from the point set. This was repeated until all residues had been assigned a hull number. Hulls were numbered outmost inward. Software to compute the RCH of the residues of a protein is available at http://icos.cs.nott.ac.uk/resources/RCH.

2.1.4 Representation and training process We used the same representation and training process for CN, SA and RCH. We predicted all three metrics as a five-state problem by binning the range of values of the metric into five intervals of approximately the same number of data points. The boundaries of the states were computed using the training set and applied to the test set. The cut points for each metric are reported in the supplementary material. The representation for the predictor contains information of a window of ± 4 residues around the target amino acid. Evolutionary information in the form of position-specific scoring matrices (PSSM) (generated using PSI-BLAST (Altschul *et al.*, 1997) using the non-redundant protein sequences database) has been used to represent each residue in the window. Hence, the representation of the CN, SA and RCH predictors consists of a vector of 180 continuous attributes.

2.2 CM representation

Three types of information sources were used for the representation of our CM predictor:

1. Detailed local sequence information from three selected regions (windows) around specific residues
2. Information about the segment connecting the target pair of residues
3. Global sequence information and other attributes.

Two windows of ± 4 amino acids are centered around the two target residues and a third window of ± 2 residues is centered around the middle point in the chain between the two target residues (Punta and Rost, 2005). Each residue in all three windows is characterised by the PSSM profile and the predictions of SS, SA, CN and RCH. The two windows around the targets are represented using 216 attributes each and the central window using 120 attributes. The connecting segment is represented by the frequencies of the amino acids types (20 attributes), predicted SS states (3 attributes), predicted SA (5 attributes) (Punta and Rost, 2005), predicted CN (5 attributes) and predicted RCH (5 attributes). In total 33 attributes are used for this connecting segment. The global sequence information uses the same representation as the connecting segment with the addition of an extra attribute representing the sequence length (34 attributes in total). Finally, two extra attributes are included: the chain separation of the target residues (Punta and Rost, 2005) and the contact propensity between the amino acid types of the two target

residues (Shackelford and Karplus, 2007). In total, 631 attributes are used to represent a given pair of residues for which we are predicting whether they are in contact or not. While this is a very large number of attributes, it is relatively small compared to other recent predictors (Li *et al.*, 2011).

2.3 Training process of the CM prediction

The training process for CM prediction is challenging for two reasons: (1) the relatively large size of the training set (32M pairs of residues and 56.7GB of disk space) which is impossible to load and hold in memory all together and (2) the low ratio (less than 2%) of true contacts, which makes the training set unbalanced and hence extremely difficult to learn from. We have used ensemble learning to deal with both challenges simultaneously.

First, to create smaller and more balanced (in terms of contacts/non contacts) training sets we generated 50 random samples from the complete set. Each sample contained around 660000 residue pairs with a fixed 2:1 proportion of non-contacts to real contacts (re-balancing the original 50:1 proportion). The ratio of contacts/non-contacts has an influence in the rate of predicted contacts produced by the system. Preliminary experiments (see supplementary material) showed that using a 1:1 ratio lead to a very high false positives rate. This was due to the fact that a 1:1 sampling induced classifiers that predicted too many spurious contacts. Hence, our strategy of resampling with a more conservative 2:1 ratio. The sampling was performed separately for each protein in the training set in order to sample residue pairs from all proteins. Afterwards, we run BioHEL 25 times for each sample with different initial random seeds. BioHEL is a stochastic algorithm (based on genetic algorithms), so each run generated a different rule set. Thus, in total we generated 1250 rule sets (50 training samples x 25 seeds). Finally, the contact predictions were performed as a simple majority vote of all rule sets in the ensemble.

The ensemble was also used to estimate the confidence of the predictions (as required by CASP). It was estimated from the margin of victory of the vote. If all rule sets agreed the confidence was 1, if the vote was split 50:50, the confidence was 0. Specifically, the confidence was defined as $conf = (2 \cdot V - T)/T$, where V is the number of votes casted for the winning outcome and T is the total number of votes casted.

2.4 Improvements since CASP8

Our CASP8 and CASP9 predictors used exactly the same representation. The only differences were in the selection of protein dataset and the sizes of the samples fed into BioHEL. The complete protein set had 2811 proteins in CASP8 (3262 in CASP9) which is a small difference. The major difference was the "thinning" process performed to select the proteins used for the CM dataset. In CASP8 we discarded all proteins larger than 350 residues and selected only a random half of the smaller ones. This resulted in a set of 15.2M pair of residues (32M in CASP9). Finally, the sizes of the 50 samples fed to BioHEL was 300K in CASP8 (660K in CASP9). These changes meant that the computational resources required for the CASP9 dataset were larger than before (25000 CPU hours were used for the training process of the CM predictor). Nevertheless, as the result section will show, the larger and more representative dataset used in CASP9 managed to boost the performance of our predictor consistently across most evaluation metrics.

3 RESULTS

We have evaluated our CM predictor according to the CASP evaluation rules (Monastyrskyy *et al.*, 2011), where (1) only long range contacts (at least 24 residues apart) are considered, (2) the predicted contacts are ranked by confidence, (3) only the top ranked 5, L/10 and L/5 predicted contacts (L=chain length) are considered and (4) the performance metrics considered are accuracy (defined as TP/(TP+FP)) and Xd, defined as:

$$Xd = \sum_{i=1}^{15} \frac{Pp_i - Pa_i}{15d_i} \quad (2)$$

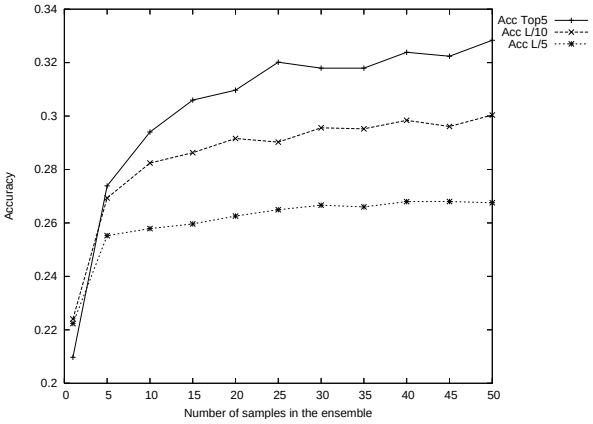

Fig. 2. Accuracy vs number of samples in the ensemble, CM-3262 test set

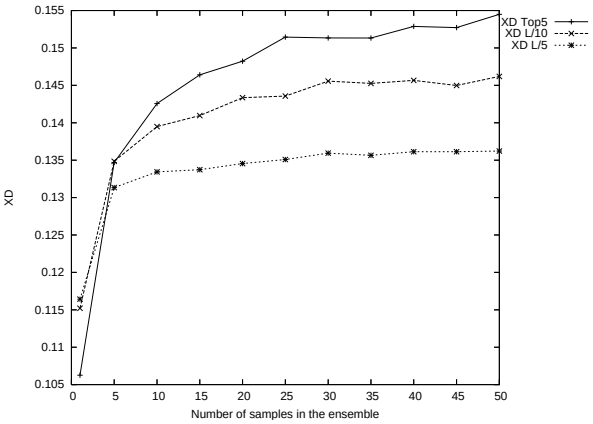

Fig. 3. Xd vs number of samples in the ensemble, CM-3262 test set

where Pp_i is the percentage of predicted pairs with a distance between $4(i-1)$ and $4i$, Pa_i is the percentage of all pairs with a distance between $4(i-1)$ and $4i$ and d_i is the upper limit of the ith bin normalised to 60.

3.1 Influence of the size of the ensemble

Firstly we assess the impact of the number of predictors in our ensemble architecture. To this aim we apply our predictor to the test partition of our CM-3262 dataset using an ensemble including the rule sets generated from only one sample (25 rule sets) and then a number of predictors ranging from 125 (using 5 training samples) to 1250 (using 50 training samples) rule sets in increments of 25. Figures 2 and 3 show the results of this experiment for accuracy and Xd, respectively, showing that the increase in predictors is beneficial to the predictive power of our method for both metrics, although the slope of the plots suggests that the influence of the ensemble size is stronger in the top predicted contacts (Top 5 and L/10) and less in L/5. Of course, increased number of predictors means increased computational cost. In our case, training the 25 rule sets derived from each sample took approx. 500 CPU hours. Finally we can also see that it is not clear that adding samples beyond 50 will contribute to a large performance increase except in the Top 5 contacts.

Table 1. Comparison of our CASP8 and CASP9 predictors on the CASP8 and CASP9 CM assessment datasets

Predictor	Metric	CASP8 dataset	CASP9 dataset
CASP8 predictor	Acc Top5	21.7±19.1	26.4±26.2
	Acc L/10	26.4±18.6	23.8±17.7
	Acc L/5	23.7±11.5	19.6±13.6
	Xd Top5	10.7±4.3	12.1±7.5
	Xd L/10	11.7±4.4	11.3±6.1
	Xd L/5	10.6±3.1	10.2±5.3
CASP9 predictor	Acc Top5	28.3±22.3	25.7±23.2
	Acc L/10	27.3±16.8	24.1±16.4
	Acc L/5	28.9±12.9	21.1±13.3
	Xd Top5	13.2±6.3	11.8±9.0
	Xd L/10	12.7±4.9	11.7±7.1
	Xd L/5	12.8±3.4	10.6±5.3

3.2 Comparing our CASP8 and CASP9 predictors

Table 6 in the supplementary material shows the performance of our CM predictor on the 28 targets used in CASP9 for the assessment of CM prediction. In most domains, and for both accuracy and Xd, we observed that the highest performance was achieved under the Top5 metric, then came the L/10 metric and finally the L/5 metric had the worst performance. These results indicate that our prediction confidence estimator procedure is sound, because the best performance is obtained when using only the predictions at the top of the rank, and it degrades when more predictions are included (first L/10, then L/5) in the metric computation. This trend, however, was not observed in all domains. Thus, the confidence procedure can still be improved further and is the subject of future research.

Next, we compare the performance of our CASP8 and CASP9 predictors (as these are trained slightly different, as detailed above) on the targets used in CASP8 (T0397-D1, T0405-D1, T0405-D2, T0416-D2, T0443-D1, T0443-D2, T0465-D1, T0476-D1, T0482-D1, T0496-D1, T0510-D3 and T0513-D2) and CASP9 (detailed in table 6 of the supplementary material) for the assessment of CM prediction. The aim of this experiment is to test the consistency of the predictor, that is, to check if it manages to maintain stable prediction capacity across CASP editions. The results of this experiment are reported in table 1 and show that the CASP9 predictor is slightly better than its CAPS8 counterpart for almost all scenarios (the only two exceptions are Acc Top5 and Xd Top5 in the CASP9 dataset) although the difference is minor. Hence, the consistency of the predictions across CASP editions is confirmed. Moreover, the CASP9 CM prediction assessors observed that the average performance of the CASP9 predictors was lower than in CASP8, indicating that the CASP9 targets were more difficult (Monastyrskyy *et al.*, 2011). The results in table 1 for the two versions of our predictor are consistent with this observation (in all metrics except Acc Top5 and Xd Top5 the predictors obtained lower average performance in CASP8 targets than in CASP9 targets), although the difference is difficult to statistically measure due to the low number of targets used in CASP8 for the assessment of CM prediction.

3.3 Comparison with the top methods in CASP9

Finally, the last stage of the evaluation is the comparison against the top methods that participated in CASP9 on the 28 domains used for the assessment of CM prediction. The predictions from all methods have been extracted from `http://www.predictioncenter.org/download_area/CASP9/predictions/RR.tar.gz`. We include in this comparison the top 10 sequence-based methods [1] (including ours) that the CASP9 assessors highlighted in their report (Monastyrskyy *et al.*, 2011). Given that not all methods managed to submit enough predictions for all targets, we will focus on a subset of 23 domains for which all methods generated enough predictions. The supplementary material reports, for each predictor, average results across all targets for which each method generated enough predictions.

Furthermore, we have analyzed these results using the recommendations proposed by Demšar (Demšar, 2006) for comparing multiple methods over multiple datasets (domains in this case). This procedure takes into account that, when comparing multiple methods, corrections need to be applied to the pair-wise comparisons in order to make sure that all of them hold simultaneously. Moreover, Demšar recommends to perform the comparison based on averaging the ranks of performance of the methods for each domain rather than on the average of a given performance metric across datasets. Furthermore, the Friedman statistical test (a non-parametric test that makes no assumptions about the distribution of the data) is used to determine if there are statistically significant differences within the methods included in the comparison. If the Friedman test detects significant differences, a post-hoc test is applied to identify them. For this paper we have used the Holm post-hoc test which compares a control method with the rest of methods to determine if there is any significant performance difference between any of them. We have used, for each of the six metrics, the method with the best average rank as control. All tests were applied with 95% confidence level.

Table 2 contains the results of this analysis. Each row shows the p-value of the Friedman test and afterwards the methods sorted by their average rank across protein domains. Bold cells indicate methods that are significantly worse than the top ranked method.

Our method presented the best average rank in five out of the six metrics. It should be mentioned, though, that the best method was shown to be statistically indistinguishable from the other nine methods in table 2 according to the Acc measure, and showed statistical superiority over the methods ranked as 8-10 according to the Xd score. Using the average rank of a metric instead of the average value of the metric reveals some difference in the ranking, favoring methods that regularly perform well. A single large performance difference in a specific protein can distort the average accuracy computation, but it will not distort the average rank.

4 MINING BIOHEL'S RULE SETS

One issue that affects most CM predictors (and many other subproblems of PSP) is that it is extremely difficult to explain the predictions performed by the system, quantify the contribution that the different parts of the representation give to the predictive power of the method or identify the interactions between the different parts of the representation. Modern CM prediction methods generally use hundreds (or even thousands) of attributes in their representation, so

[1] The two top groups in the CASP9 contact map assessment, 391 and 490 are not included in this comparison as these groups derived contact predictions from 3D models, instead of directly from sequence.

Table 2. Statistical comparison of CASP9 methods on the common set of 23 domains using the Friedman and Holm statistical tests.

Acc Top5	p-value	Method	103	51	138	375	422	244	119	2	214	80
	0.1199	Rank	4.63	4.78	4.80	4.85	5.57	5.65	5.72	5.89	6.48	6.63
Acc L/10	p-value	Method	51	103	138	2	375	244	422	119	214	80
	0.2482	Rank	3.98	4.63	4.80	4.93	5.43	5.65	5.78	6.35	6.61	6.83
Acc L/5	p-value	Method	51	2	138	103	244	375	422	214	119	80
	0.0916	Rank	4.63	4.78	4.80	4.85	5.57	5.65	5.72	5.89	6.48	6.63
Xd Top5	p-value	Method	51	103	375	138	2	119	422	244	**80**	**214**
	0.0142	Rank	3.98	4.63	4.80	4.93	5.43	5.65	5.78	6.35	**6.61**	**6.83**
Xd L/10	p-value	Method	51	2	103	138	375	422	244	119	**80**	**214**
	0.0025	Rank	4.15	4.63	4.76	4.83	4.93	5.63	5.67	6.04	**6.78**	**7.57**
Xd L/5	p-value	Method	51	2	138	375	103	422	244	**119**	**80**	**214**
	0.0227	Rank	4.43	4.74	4.74	4.85	4.93	5.52	5.76	**5.96**	**6.74**	**7.33**

Each row sorts the 12 methods by their average rank for the row's metric. p-value = result of the Friedman test. Bold cells indicate statistically worse methods than the top ranked method at 95%. Methods are identified by their CASP9 ID. Our method = 51.

this issue becomes even more important. Our BioHEL machine learning system generates human-readable sets of production rules, and we can exploit this characteristic to extract knowledge from the rules that can help address these challenges. Figure 4 contains one of the 1250 rule sets that form our CM predictor. A full description of the meaning of the attributes that appear in the rules is available in the supplementary material. On average a rule set contains 152.5±7.1 rules, and each rule uses 8.4±2.9 attributes. Given the large volume of rules it would be very difficult to inspect them manually, but we can extract global statistics from the complete set of rules.

4.1 Most frequently used attributes

Table 3 lists the top 20 attributes most frequently used in the rules. The complete ranking is available in the supplementary material of the paper and contains all 631 attributes of our representation. Thus, all of them were used, although some of them rarely.

All four types of 1D predictions for both target residues (_r1/_r2) were within the top 20 most frequently used attributes which indicates that, despite expressing similar structural properties (especially SA and RCH), **all of them contributed complementary information to the predictor**. The static AA-wise contact propensity metric (Shackelford and Karplus, 2007), a simplistic predictor on its own, was the second most used attribute when combined with others in a rule. Properties about window positions other than those of the target pair of amino acids start appearing at position 8 of the ranking, and the evolutionary information (the PSSM attributes) at position 11. The PSSM attributes appearing in the top 20 were all polar (D, E, N and K), and most of them charged. At positions 18-19 we found two summary statistics for the chain segment connecting the target pair of residues: the proportion of AAs that belong to the outer hull and the proportion of amino acids in coil state.

Table 3. Top 20 most frequent attributes used in BioHEL's rules

Rank	Attribute	Ratio
1	PredSA_r1	22.3
2	propensity	20.1
3	PredSA_r2	18.4
4	PredSS_r1	17.4
5	PredSS_r2	15.7
6	PredRCH_r1	15.6
7	PredRCH_r2	13.9
8	PredSS_r1_1	13.7
9	PredSS_r2_-1	13.2
10	PredCN_r1	12.3
11	PSSM_r2_0_E	11.6
12	PSSM_r2_0_D	10.9
13	PredCN_r2	10.1
14	PredSS_r1_-1	10.0
15	PSSM_r1_0_D	10.0
16	PSSM_r1_0_E	9.9
17	PSSM_r2_0_N	9.4
18	PredRCH_freq_connecting_0	9.4
19	PredSS_freq_connecting_C	9.0
20	PSSM_r2_0_K	9.0

Ratio = percentage of rules where the attribute appears

4.2 Contribution of the information sources

To measure the contribution of different information sources in our rule sets we aggregated the ranks of all the attributes (window positions/frequency counts) belonging to each source. The results of this analysis are reported in table 4. We can observe that, while PredSA_r1 was the most frequently used attribute, the corresponding attributes for other positions in the windows are much less used, and the average rank of that type of information is only 10. On the other hand, the predicted SS attributes for most of the window positions around the targets are useful, as their average rank is 5th and 6th, for the 1st and the 2nd residue in the pair, respectively. Even higher is the average rank of the frequency of predicted SS elements across the connecting segment between the target pair (PredSS_freq_connecting), which is the first actual average rank (unlike propensity, separation and length that are individual attributes). The average ranks for RCH, SA and CN are relatively similar, and much lower than the SS one. At the bottom of the average ranks we find all the attributes related to the central window, clearly indicating that these are the least useful information sources.

4.3 Contribution of the PSSM profile's columns

Table 4 also shows a big disparity between the best and average rank of the evolutionary information (PSSM_r1 and PSSM_r2). To analyse this in more detail we have computed the average rank of the PSSM elements corresponding to each amino acid type. Table 5 contains the results of this analysis, reporting the average rank for the positions of the windows associated to the target residues or for the complete windows. Only the two windows around the target pair have been included in this analysis, ignoring the central window, and the rank is sorted by the central positions rank. As we observed in the top 20 rank, the top AA types are all polar and most of them charged (except H which is lower in the rank). Next we find two aliphatic AAs (I and V). Aromatic and tiny AAs are in general low in the ranking. There are small differences between

1: If $propensity \in [0.53, 1.51]$, $PredSA\_freq\_connecting\_0 \leq 0.00$, $PSSM\_r1\_0\_E \in [-10.06, -4.78]$, $PSSM\_r2\_0\_Q \leq -3.57$, $PSSM\_central\_-2\_Q \in [-12.98, 6.42]$, $PSSM\_central\_-1\_R \in [-2.96, 7.34]$ → **predict** contact
2: If $PredSS\_r1 \notin \{C\}$, $PredSS\_r1\_2 \in \{E\}$, $PredSS\_r2\_2 \in \{E, X\}$, $PredSA\_freq\_connecting\_2 \leq 0.52$, $PredRCH\_r2\_1 \in \{2, 3, 4\}$, $PredSA\_r1\_-2 \in \{0, 1, X\}$, $PredSA\_r2\_3 \in \{0, 1, 3\}$, $AA\_freq\_connecting\_Y \leq 0.00$, $PSSM\_r1\_0\_K \in [-9.97, -2.03]$, $PSSM\_r2\_-1\_N \geq -10.69$, $PSSM\_r2\_0\_I \in [2.24, 8.16]$, $PSSM\_central\_1\_N \geq -7.67$ → **predict** contact
.
.
161: Everything else → **predict** non-contact

Fig. 4. Rule set generated by BioHEL. Attributes with _r1, _r2 or _central belong to the corresponding three windows of AAs. A suffix (-4 .. 4) after _r1/_r2/_central gives the relative window position. The _connecting suffix is used for frequency statistics computed over the segment connecting the target pair. X is the end of chain symbol.

Table 4. Best and average rank of the information sources in our CM representation sorted by average rank

Type	Best rank	Average rank
propensity	2	2.0±0.0
separation	24	24.0±0.0
PredSS_freq_connecting	19	32.0±9.9
length	42	42.0±0.0
PredSS_r1	4	43.0±31.4
PredSS_r2	5	47.8±38.9
PredSS_freq_global	30	75.3±58.5
PredCN_r1	10	81.6±40.2
PredRCH_r1	6	82.7±40.1
PredSA_r1	1	82.8±42.6
PredRCH_r2	7	85.6±37.8
PredCN_r2	13	89.2±37.7
AA_freq_connecting	45	91.8±38.4
PredSA_freq_connecting	27	92.4±56.2
PredSA_r2	3	93.6±44.3
PredRCH_freq_connecting	18	114.2±48.9
PredCN_freq_connecting	63	118.2±49.2
PredCN_freq_global	31	133.2±66.6
PredRCH_freq_global	62	134.2±37.2
PredSA_freq_global	65	171.8±81.4
PredSS_central	232	282.2±39.5
AA_freq_global	181	290.9±63.7
PSSM_r1	15	322.6±130.8
PredCN_central	301	329.4±18.0
PSSM_r2	11	334.6±144.8
PredRCH_central	305	366.4±33.6
PredSA_central	331	408.8±41.4
PSSM_central	390	568.6±50.4

Table 5. Rank of the evolutionary information attributes aggregated by their AA type

AA Type	Target residues rank	Whole window rank
D	13.5±1.5	184.5±84.4
E	13.5±2.5	189.7±82.8
N	19.5±2.5	197.3±79.8
K	21.5±1.5	221.1±93.5
Q	27.0±2.0	236.4±91.4
R	34.0±1.0	271.2±109.7
I	57.0±11.0	340.5±148.8
V	66.0±16.0	359.7±151.8
S	73.0±2.0	303.9±108.4
G	85.0±15.0	226.6±80.1
H	92.5±2.5	357.9±107.5
L	111.0±23.0	384.3±140.7
P	136.0±12.0	257.7±96.3
M	209.0±7.0	374.0±76.2
F	227.5±13.5	439.7±92.4
C	252.5±85.5	441.4±90.5
T	260.5±24.5	413.1±74.2
Y	270.5±7.5	467.2±73.6
W	292.0±19.0	437.6±67.6
A	490.5±22.5	467.4±60.5

the average rank for the target residues and for the whole window for most AA types except for G (which raises 5 positions in the whole window rank), P (which raises 6 positions) and V (which falls 4 positions). Interestingly, these three AA types are among the most frequent of the AA_freq_connecting attributes (as shown in the complete attribute ranking in the supplementary material).

4.4 Interactions between attributes

The analysis of BioHEL's rules performed so far has revealed useful information about the contribution of the different information sources into the predictor. However, it is a limited analysis. A rule is activated when all of its attributes are activated together. Therefore, it is also important to look at which pairs of attributes appear together frequently in rules. Table 6 reports the top 20 pairs of attributes. In this case we do not report the complete ranking as there were almost 200K pairs of attributes identified in BioHEL's rules (which is roughly half of the total possible pairs of attributes). We can observe a very clear trend in the most frequent pairs: a pair includes one attribute associated to each of the two resides in the pair. This was expected as the rules try to predict if the two residues are in contact. Interestingly, the most frequent pair (PredSS_r1 & PredSS_r2) does not include the most frequent attribute (PredSA_r1). We can observe that PredSS (both r1/r2) and propensity appear very often across the top 20 pairs rank.

4.5 Lessons learnt from the rule analysis

This section has provided a thorough analysis of the rules generated by our BioHEL system. We have been able to quantify the contribution of each of the sources of information as well as individual attributes, thus providing useful information to the designers of CM prediction methods about which information sources to choose. Moreover, this information can be applied in specific ways to refine the representation of the predictor: indicating which parts may be candidates to be discarded all together (e.g. the central window) or, in a more fine-grained strategy, the relevance of window positions, PSSM columns and, in general, individual attributes. Thus, we can avoid a blind feature selection process which, considering the size of the training set (in both attributes and instances) could be very

Table 6. Top 20 most frequent pairs of attributes used together in BioHEL's rules

Rank	Attribute 1	Attribute 2	Ratio
1	PredSS_r1	PredSS_r2	5.4037
2	PredSA_r1	PredSA_r2	4.7722
3	PredSA_r1	propensity	4.7537
4	PredSS_r1_1	PredSS_r2	4.1945
5	PredSS_r1	PredSS_r2_-1	3.9722
6	PredSS_r1_1	PredSS_r2_-1	3.7199
7	PredSA_r2	propensity	3.6903
8	PSSM_r2_0_E	PredSA_r1	3.4991
9	PSSM_r2_0_E	propensity	3.3260
10	PredSA_r1	PredSS_r2	3.2753
11	PredSS_r1	PredSS_r2_1	3.1855
12	PredRCH_r1	PredRCH_r2	3.1802
13	PSSM_r2_0_D	PredSA_r1	2.9923
14	PSSM_r1_0_E	propensity	2.9389
15	PredRCH_r2	PredSA_r1	2.9363
16	PredRCH_r1	PredSA_r2	2.8951
17	PredSA_r2	PredSS_r1	2.8735
18	PredSS_r1_-1	PredSS_r2	2.8676
19	PredSA_r1	PredSS_r2_-1	2.7636
20	PSSM_r2_0_K	PredSA_r1	2.7557

Ratio = percentage of rules using the attributes

computationally demanding. Finally, the analysis of the frequent pairs of attributes provides useful information to understand how the prediction is performed.

5 CONCLUSIONS

This paper has described our CM prediction methodology that participated in CASP9 (under the name Infobiotics). Our method is based on (a) the integration of several information sources including the prediction of four types of 1D structural features and (b) an ensemble architecture that allows the use of very large training sets via a distributed training process. Our experiments show that both aspects are crucial for the predictor's performance: On one hand larger ensembles obtain better performance. On the other hand the analysis of the rule sets generated by our BioHEL machine learning system has identified the important parts of the representation (placing all the 1D features among the top ranked attributes) and their interactions. The comparison against the top sequence-based methods in CASP9 showed that our predictor is very competitive, ranking first on five out of the six metrics. In future work we would like to bring this analysis of rules much further with the objective of refining our predictor and possibly tailoring its representation for varying scenarios. Also, there are many aspects of our prediction architecture that can be adjusted in different ways (e.g. the size and ratio of contacts/non-contacts of the samples, size of the windows) which could improve its performance. Finally, we would like to study how to improve our prediction confidence estimation.

ACKNOWLEDGEMENT

We would like to acknowledge Jonathan D. Hirst and Michael Stout for their collaboration on this work. We are also grateful for the use of the HPC facility at the University of Nottingham.

Funding: UK Engineering and Physical Sciences Research Council (EPSRC) under grants EP/H016597/1, GR/T07534/01 and EP/J004111/1

REFERENCES

Altschul, S. F., Madden, T. L., Scher, A. A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. J. (1997). Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic Acids Res*, **25**, 3389–3402.

Bacardit, J., Stout, M., Krasnogor, N., Hirst, J. D., and Blazewicz, J. (2006). Coordination number prediction using learning classifier systems: performance and interpretability. In *GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pages 247–254. ACM Press.

Bacardit, J., Stout, M., Hirst, J. D., Valencia, A., Smith, R. E., and Krasnogor, N. (2009a). Automated alphabet reduction for protein datasets. *BMC Bioinformatics*, **10**, 6.

Bacardit, J., Burke, E. K., and Krasnogor, N. (2009b). Improving the scalability of rule-based evolutionary learning. *Memetic Computing*, **1**, 55–67.

Barber, C. B., Dobkin, D. P., and Huhdanpaa, H. (1996). The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, **22**(4), 469–483.

Bassel, G. W., Glaab, E., Marquez, J., Holdsworth, M. J., and Bacardit, J. (2011). Functional network construction in arabidopsis using rule-based machine learning on large-scale data sets. *The Plant Cell*, **23**(9), 3101–3116.

Cheng, J. and Baldi, P. (2007). Improved residue contact prediction using support vector machines and a large feature set. *BMC Bioinformatics*, **8**(113).

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, **7**, 1–30.

Jones, D. (1999). Protein secondary structure prediction based on position-specific scoring matrices. *J Mol Biol*, **292**, 195–202.

Kabsch, W. and Sander, C. (1983). Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, **22**, 2577–2637.

Kinjo, A. R., Horimoto, K., and Nishikawa, K. (2005). Predicting absolute contact numbers of native protein structure from amino acid sequence. *Proteins*, **58**, 158–165.

Li, Y., Fang, Y., and Fang, J. (2011). Predicting residue-residue contacts using random forest models. *Bioinformatics*.

MacCallum, R. (2004). Striped sheets and protein contact prediction. *Bioinformatics*, **20**, I224–I231.

Monastyrskyy, B., Fidelis, K., Tramontano, A., and Kryshtafovych, A. (2011). Evaluation of residue-residue contact predictions in CASP9. *Proteins: Structure, Function, and Bioinformatics*, **79**(S10), 119–125.

Noguchi, T., Matsuda, H., and Akiyama, Y. (2001). Pdb-reprdb: a database of representative protein chains from the protein data bank (pdb). *Nucleic Acids Res*, **29**(1), 219–20.

Preparata, F. P. and Shamos, M. I. (1985). *Computational Geometry : An Introduction*. Texts and monographs in computer science. Springer-Verlag.

Punta, M. and Rost, B. (2005). Profcon: novel prediction of long-range contacts. *Bioinformatics*, **21**, 2960?2968.

Rost, B. and Sander, C. (1994). Conservation and prediction of solvent accessibility in protein families. *Proteins*, **20**, 216–226.

Shackelford, G. and Karplus, K. (2007). Contact prediction using mutual information and neural nets. *Proteins: Structure, Function, and Bioinformatics*, **69**(S8), 159–164.

Stout, M., Bacardit, J., Hirst, J. D., and Krasnogor, N. (2008). Prediction of recursive convex hull class assignments for protein residues. *Bioinformatics*, **24**(7), 916–923.

Stout, M., Bacardit, J., Hirst, J. D., Smith, R. E., and Krasnogor, N. (2009). Prediction of topological contacts in proteins using learning classifier systems. *Soft Computing*, **13**, 245–258.

Tress, M. L. and Valencia, A. (2010). Predicted residue-residue contacts can help the scoring of 3D models. *Proteins: Structure, Function, and Bioinformatics*, **78**(8), 1980–1991.

Vassura, M., Margara, L., Di Lena, P., Medri, F., Fariselli, P., and Casadio, R. (2008). Ft-comar: fault tolerant three-dimensional structure reconstruction from protein contact maps. *Bioinformatics*, **24**(10), 1313–1315.

Zhang, Y. (2009). I-tasser: Fully automated protein structure prediction in CASP8. *Proteins: Structure, Function, and Bioinformatics*, **77**(S9), 100–113.