Bioinformatics, 33(17), 2017, 2675–2683
doi: 10.1093/bioinformatics/btx296
Advance Access Publication Date: 3 May 2017
Original Paper

Structural bioinformatics

A deep learning framework for improving long-range residue–residue contact prediction using a hierarchical strategy

Dapeng Xiong[1,2], Jianyang Zeng[2,3,*] and Haipeng Gong[1,2,*]

[1]MOE Key Laboratory of Bioinformatics, School of Life Sciences, [2]Beijing Innovation Center of Structural Biology and [3]Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing 100084, China

*To whom correspondence should be addressed.

Associate Editor: Alfonso Valencia

Received on October 18, 2016; revised on April 9, 2017; editorial decision on April 29, 2017; accepted on May 2, 2017

Abstract

Motivation: Residue–residue contacts are of great value for protein structure prediction, since contact information, especially from those long-range residue pairs, can significantly reduce the complexity of conformational sampling for protein structure prediction in practice. Despite progresses in the past decade on protein targets with abundant homologous sequences, accurate contact prediction for proteins with limited sequence information is still far from satisfaction. Methodologies for these hard targets still need further improvement.

Results: We presented a computational program DeepConPred, which includes a pipeline of two novel deep-learning-based methods (DeepCCon and DeepRCon) as well as a contact refinement step, to improve the prediction of long-range residue contacts from primary sequences. When compared with previous prediction approaches, our framework employed an effective scheme to identify optimal and important features for contact prediction, and was only trained with coevolutionary information derived from a limited number of homologous sequences to ensure robustness and usefulness for hard targets. Independent tests showed that 59.33%/49.97%, 64.39%/54.01% and 70.00%/59.81% of the top L/5, top L/10 and top 5 predictions were correct for CASP10/CASP11 proteins, respectively. In general, our algorithm ranked as one of the best methods for CASP targets.

Availability and implementation: All source data and codes are available at http://166.111.152.91/Downloads.html.

Contact: hgong@tsinghua.edu.cn or zengjy321@tsinghua.edu.cn

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Native residue–residue contacts provide essential information to facilitate the challenging task of protein structure prediction (Eickholt and Cheng, 2012; Schneider and Brock, 2014; Tress and Valencia, 2010; Vassura *et al.*, 2008). In practice, residue contact information could be integrated into the scoring function to reduce the space of conformational sampling (Li *et al.*, 2011; Wu and Zhang, 2008) or to improve the selection of template models (Eickholt and Cheng, 2012; Miller and Eisenberg, 2008). Recently, its application has been expanded to rational drug design (Kliger *et al.*, 2009).

Contemporary methods for protein residue contact prediction can be categorized as template- and sequence-based (Eickholt and Cheng, 2012; Li *et al.*, 2011). The former makes prediction based on homologous templates (Misura *et al.*, 2006; Skolnick *et al.*, 2004; Wu and Zhang, 2008) and is thus limited in usefulness. Conversely, the latter that only requires the amino acid sequence for prediction has been investigated more enthusiastically.

The sequence-based methods were first developed by retrieving statistical information from the structural database to train various machine learning models, including artificial neural network (Jones et al., 2015; Kosciolek and Jones, 2015; Punta and Rost, 2005; Tegge et al., 2009; Xue et al., 2009; Zhang and Huang, 2004), support vector machine (Cheng and Baldi, 2007; Wu and Zhang, 2008; Zhao and Karypis, 2005), random forest (Li et al., 2011; Wang and Xu, 2013), hidden Markov model (Björkholm et al., 2009; Shao and Bystroff, 2003) and deep architectures (Di Lena et al., 2012; Eickholt and Cheng, 2012). As one popular representative, CMAPpro (Di Lena et al., 2012) was designed in a multi-stage strategy, utilizing predicted coarse contacts between secondary structure elements (SSEs) to further improve the accuracy of residue contact prediction.

With the rapid expansion of sequence database, another kind of sequence-based methods were proposed, under the assumption that contacting residues should present correlated mutations in the multiple sequence alignment (MSA). Residue coevolution could be identified by metrics including mutual information (Dunn et al., 2008; Lee and Kim, 2009; Little and Chen, 2009), direct coupling analysis (Ekeberg et al., 2013, 2014; Morcos et al., 2011; Weigt et al., 2009) and sparse inverse covariance (Jones et al., 2012). Despite the exceptional predictive power for targets with plenty of homologous sequences, performance of coevolution-based methods severely relies on the availability and quality of MSA, which hinders their success on protein targets of small families (Kamisetty et al., 2013).

In principle, the two kinds of sequence-based methods could be combined to further improve prediction, because of the complementarity between information retrieved from structure and sequence databases. In this respect, some preliminary trials have been made (Jones et al., 2015; Kamisetty et al., 2013; Skwark et al., 2014; Wang and Xu, 2013). As a famous example, GREMLIN simply took the results of a machine-learning-based predictor as the input of a coevolution-based statistical model (Kamisetty et al., 2013). PconsC2 incorporated predicted coevolutionary information and a few basic structural features into a deep learning model (Skwark et al., 2014). Most recently, Yang et al. presented a novel method R_2C, which evaluates residue contacts using the linear combination of scores from a machine-learning-based and a coevolution-based predictors together with a 2D Gaussian noise filter (Yang et al., 2016). Nevertheless, the two kinds of information were integrated in crude manners in all previous explorations, which therefore still await further refinement. Furthermore, most methods relying on coevolutionary information require sufficient sequence abundance to guarantee accuracy and thus fail on targets with limited sequence information. Special care therefore should be taken for these hard targets in model construction. In addition, after including coevolutionary information, the effective combination of feature descriptors for machine learning models should be further optimized.

In this work, we developed a package DeepConPred, which includes a pipeline of two deep-learning-based models (DeepCCon and DeepRCon) as well as a refinement step, to effectively combine statistical information retrieved from the structure database and coevolutionary information extracted from the sequence database for long-range residue–residue contact prediction. Using a hierarchical approach similar to CMAPpro, the coarse contacts between SSEs predicted by DeepCCon in the first stage can facilitate the prediction of residue contacts by DeepRCon in the second stage. Deep learning technologies have generally exhibited better predictive power than conventional methods, especially in the present era of massive data (Najafabadi et al., 2015). Previous applications of deep architectures (Di Lena et al., 2012; Eickholt and Cheng, 2012; Skwark et al., 2014), however, did not achieve the expected improvement on residue contact prediction in the latest Critical Assessment of protein Structure Prediction (CASP) competitions (Monastyrskyy et al., 2015), possibly due to the lack of extraction of representative and predictive features. Here, for both DeepCCon and DeepRCon, we proposed a number of novel features and incorporated them with good known features for more comprehensive description of protein structural properties. Moreover, for the first time to our knowledge, we employed feature selection to eliminate feature redundancy in contact prediction. Specifically, DeepCCon and DeepRCon were trained using coevolutionary information derived from a reduced number of homologous sequences to ensure robustness for small-family proteins, while the subsequent refinement step was designed to integrate full coevolutionary information to improve predictions of large-family proteins. Notably, these unique protocols have not been adopted in previous successful predictors including MetaPSICOV (Jones et al., 2015), CoinDCA-NN (Ma et al., 2015) and CONSIP2 (Kosciolek and Jones, 2015). According to performance evaluation on CASP proteins, our method reaches a high level of prediction accuracy in general and ranks as one of the best methods on CASP targets.

2 Materials and methods

Our algorithm integrates two deep learning models (DeepCCon and DeepRCon) in a hierarchical strategy (Fig. 1). In specific, DeepCCon predicts the coarse contact information between two SSEs in the first stage and this information is fed to DeepRCon to facilitate the prediction of residue contacts. For model training at each stage, a complete feature set was first constructed, which was utilized for the optimization of all tunable parameters. Feature selection was then conducted to find the optimal feature subset, which could not only reduce computational complexity but also further improve performance. Predicted residue contacts were finally refined by a deep learning model, which combines the prediction scores of DeepRCon and a coevolution-based predictor CCMpred (Seemayer et al., 2014).

2.1 Datasets

The datasets in this study were derived from the database of Structure Classification of Proteins—extended (SCOPe) (Fox et al., 2014). In specific, we extracted the initial dataset from SCOPe release 2.05 and removed domains that had <50 residues, multiple structures or missing backbone atoms. Redundancy was removed using BLASTCLUST (Altschul et al., 1997), by clustering the domains with a cutoff of 20% sequence identity and choosing one

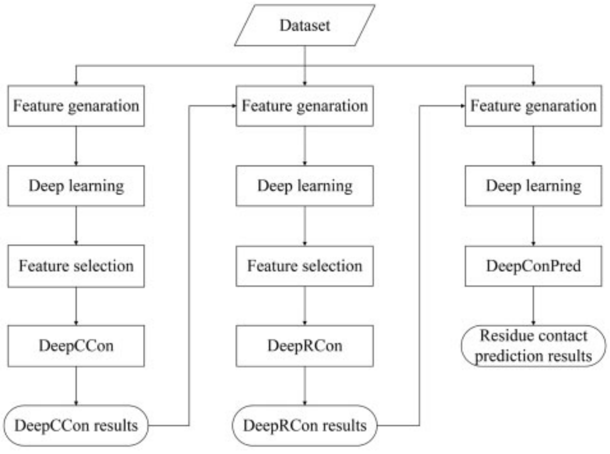

Fig. 1. Flowchart overview of the pipeline of DeepConPred

representative (the shortest one) from each acquired cluster. Similarly, the dataset was further filtered to retain only one representative per SCOPe family. The complete dataset contained 3443 protein domains.

For an objective performance evaluation, the complete dataset was divided into two mutually exclusive parts: a training set for model optimization/cross-validation and a testing set for independent benchmark test. Considering that an effective and robust method should show good predicting power for the presently unknown and undiscovered targets, 2898 domains released in SCOPe version 1.75 were assigned to the training dataset, while the remaining 545 domains, namely the novel folds with respect to the training dataset, were assigned to the testing dataset.

To compare with other state-of-the-art methods, we also evaluated the performance on all valid targets in the CASP10 (Monastyrskyy *et al.*, 2014) and CASP11 (Monastyrskyy *et al.*, 2015) competition datasets, following the CASP definition and classification of residue contacts. In specific, a pair of residues is considered in contact if the distance between their C_β atoms (C_α in case of glycine) is <8Å. Non-local residue contacts were classified into three types according to sequence separation: long-range contacts (separation $\geq$ 24), medium-range contacts (12 $\leq$ separation < 24) and short-range contacts (6 $\leq$ separation < 12). According to this definition, with respect to the long-range contacts, our training dataset includes 647718 contact and 47355205 non-contact residue pairs, while our testing dataset includes 109189 contact and 6432869 non-contact residue pairs. In this study, we only focused on the prediction of long-range contacts, which are generally considered as the most valuable and informative in structure modeling and also the most difficult to predict (Di Lena *et al.*, 2012; Schneider and Brock, 2014; Yang *et al.*, 2016).

2.2 Deep learning framework

We adopted the deep belief network (DBN) (Hinton *et al.*, 2006; Hinton and Salakhutdinov, 2006) to build our models (see Supplementary Fig. S1 for the schematic architecture and Supplementary Material for detailed introduction). The model architecture as well as all tunable hyper-parameters were optimized by 5-fold cross validation on the training dataset using the complete feature set. According to the cross-validation results, our final models consist of one input layer, three hidden layers and one output layer, with the overall architectures of *d*–700–200–700–2 for DeepCCon and DeepRCon, and *d*–300–200–150–2 for the refinement step, where *d* is the dimension of the feature vectors. In each building block (called restricted Bolzmann machine) of our DBN models, the learning rates of weights and biases were set to 0.01 and 0.1, respectively, with the weight cost of 0.0002. The momentum was initially set to 0.5 and then increased to 0.9 after 5 epochs. The number of training samples in mini-batches and epoch number were both set to 100.

2.3 Feature selection

We used the group minimax concave penalty (MCP) (Huang *et al.*, 2012) to find the optimal feature subset within the complete feature set of our models (see Supplementary Material for detailed introduction). The algorithm was implemented using the grMCP R package (Breheny and Huang, 2009; Huang *et al.*, 2012), with weight of regularization parameters of the group and L2 penalties as well as the maximum number of iterations optimized to 0.5 and 100 000, respectively by 5-fold cross validation on the training dataset. Final models were constructed using the optimal feature subsets.

2.4 Coarse contact prediction

We first built the training and testing datasets of SSE pairs from the corresponding protein datasets respectively. In specific, all SSEs were extracted by removing coil residues as well as short strands ($\leq$3 residues) and short helices ($\leq$6 residues) following the DSSP definition (Kabsch and Sander, 1983). Thus, the combination of all SSE pairs within each protein jointly composed the dataset. The SSE pairs separated by $\leq$1 residue were ignored to avoid ambiguity. A pair of SSEs was defined as contacting if there were $\geq$2 inter-SSE residue–residue contacts with the distal contacting points separated by $\geq$1 residue in both SSEs, and as no-contact otherwise. Contacting SSE pairs were subsequently categorized as parallel and anti-parallel contacts if the scalar angle between their orientation vectors was $\leq$90° or >90°, respectively, where the orientation vector of each SSE was computed from the centers of mass of C_α atoms in the first and second halves.

We then constructed a deep learning model DeepCCon to predict the probabilities of parallel contact, anti-parallel contact and no-contact for a pair of SSEs denoted as S_m and S_n, based on their amino acid sequences. Here, we designed several new features for an SSE pair:

1. Coevolutionary information: The residue coevolutionary information was first predicted by plmDCA (Ekeberg *et al.*, 2013). In order to minimize the reliance on sequence abundance, the number of effective homologous sequences was restricted to be $\leq 0.7L$ (L denotes the chain length of target protein) in plmDCA calculation. That is, if the number exceeded $0.7L$, effective homologous sequences were grouped into $0.7L$ clusters using BLASTCLUST and one representative sequence was chosen from each group to compose the final sequence set. The choice of $0.7L$ as the cutoff was made by statistical analysis over SCOPe protein families (see Section 3.1 for details). Procedure for plmDCA calculation is described in details in the Supplementary Material. Given the plmDCA scores, each SSE was evenly divided into five sub-regions and residues were assigned to these sub-regions based on percentiles of their sequence orders within the SSE. The score for every pair of inter-SSE subregions was then defined as the maximum plmDCA score among all inter-SSE residue pairs within the sub-regions. This feature vector contained 25 entries in total.
2. Contact propensity: Again, both SSEs were divided into five subregions as aforementioned. Propensities of inter-SSE residue contacts were first estimated from proteins in the training dataset by differentiating sub-regions as well as parallel/anti-parallel SSE contacts. In specific, the propensity score P_{ij} for each residue pair was derived as,

$$P_{ij} = \frac{N_{\mathrm{con}}(i,j)}{N_{\mathrm{con}}(i,j) + N_{\mathrm{no-con}}(i,j)}, \tag{1}$$

where $N_{\mathrm{con}}(i,j)$ and $N_{\text{no-con}}(i,j)$ are the numbers of contact and no-contact pairs, respectively. Thus, 50 contact-propensity matrices were obtained in total, each of which had the size of 20 × 20 to enumerate all 20 amino acids. Scores in each propensity matrix were rescaled with the minimum and maximum scores set to 0 and 1 respectively. The contact propensity for a pair of sub-regions within two parallel/anti-parallel SSEs was then represented by the maximum propensity score of all inter-SSE residue pairs. This feature vector contained 50 entries in total.
3. Natural vector of intervening sequence: Theoretically, each protein sequence could be equivalently represented by a 60D natural

vector that describes the positional distribution of 20 amino acids (Yu *et al.*, 2013). Here, we presented the amino acid sequence between S_m and S_n using natural vectors.

4. Length of intervening sequence: This feature was represented as a binary vector for states in eight intervals (2–5, 6–9, 10–12, 13–15, 16–23, 24–27, 28–65 and ≥66).
5. Lengths of sequences connecting neighboring SSEs (for $\{S_{k-1}, S_k\}$ and $\{S_k, S_{k+1}\}$, where $k = m, n$): Similarly, the length of intervening sequence was represented by a vector of binary states in seven categories (0, 1, 2–7, 8–22, 23–37, 38–52 and ≥53, where 0 means S_k is the first or last one). This feature contained 28 entries (two intervening sequences for each SSE).
6. Number of intervening SSEs: Similarly, the number of SSEs between S_m and S_n was represented by a vector of binary states in eight intervals (0, 1, 2–5, 6–13, 14–22, 23–31, 32–40, ≥41).

Features (1–3) are uniquely proposed by us, while features (4–6) were adopted by CMAPpro but using original values curtly. Notably, we made abundant designs on the boundary of intervals in features (4–6) to allow roughly even distributions of corresponding properties for proteins in the training dataset. Besides these features, we also incorporated known features reported by CMAPpro, assuming that a combination of good features can better reflect the encoded information and thus improve the predictive power. These features are listed in detail as below:

1. Amino acid compositions of the SSEs (for $\{S_{k-1}, S_k, S_{k+1}\}$, where $k = m, n$): The amino acid composition records the frequencies of all 20 amino acids within each SSE. This feature thus contained 120 entries (20 entries for each SSE).
2. Numbers of residues of the SSEs (for $\{S_{k-1}, S_k, S_{k+1}\}$, where $k = m, n$): This feature was extracted as a 6D vector (one entry for each SSE).
3. A vector of flags to identify the first, second, second-to-last and last SSEs in the sequence: Only 6 flags were needed here, because the SSE on the N-/C-terminal side within a pair is impossible to be the last/first one in the sequence.
4. Amino acid compositions for even- and odd-numbered positions of the SSEs: This feature was specially designed for strands that have the periodicity of 2, and contained 80 entries (20 entries for even-numbered and 20 entries for odd-numbered positions in each SSE).

Combining all features, we constructed a 391D complete feature vector for each SSE pair. Hyper-parameters of the DBN model (see Section 2.2) were optimized using this complete feature set, by 5-fold cross validation on the training dataset. The complete feature set was then optimized to the best feature subset by the group MCP with tunable parameters optimized simultaneously (see Section 2.3). The final model was built using the optimal feature subset. Notably, although secondary structure information was extracted using DSSP (Kabsch and Sander, 1983) in model construction, predicted secondary structure information by SSpro (Magnan and Baldi, 2014) was used for feature extraction in the practical applications of DeepCCon, e.g. providing the coarse contact information for the subsequent residue–residue contact prediction.

The performance of models was evaluated using three measures, including positive predictive value (PPV, or precision), true positive rate (TPR, or recall) and *F*-measure:

$$\mathrm{PPV} = \frac{XX}{AX + PX + NX}, \tag{2}$$

$$\mathrm{TPR} = \frac{XX}{XA + XP + XN}, \tag{3}$$

$$F - \mathrm{measure} = \frac{2 \times \mathrm{PPV} \times \mathrm{TPR}}{\mathrm{PPV} + \mathrm{TPR}}, \tag{4}$$

where the labels of P, A and N refer to parallel contact, anti-parallel contact and no-contact, respectively. The format of XY refers the number of times when SSE pairs in class of $X \in \{P,A,N\}$ are predicted to be in class of $Y \in \{P,A,N\}$. As a comprehensive evaluator combining both PPV and TPR, *F*-measure was chosen as the primary evaluation criterion for parameter optimization.

2.5 Residue–residue contact prediction

We developed a deep-learning-based model DeepRCon to predict residue–residue contacts, by combining multiple features including the coarse contact information provided by DeepCCon. The input features could be categorized as those of residue pairs, of intervening sequences and of entire protein. For the target residue pair, each residue was represented by a window of odd size (w) centered at it to consider the effect of local environment. For the intervening sequence, residue at the center as well as those located at the centers of the first and second halves were chosen, and three windows of identical odd sizes (cw) centered at these residues were employed to reflect property of the sequence. DeepRCon was trained in a protocol similar to DeepCCon (Fig. 1). We designed the following new features for DeepRCon:

1. Coarse contact information: If the target residues were located in SSEs, this feature described the probabilities of the corresponding SSE pair in various categories (parallel contact, anti-parallel contact and no-contact), as predicted by DeepCCon.
2. Smoothed position specific scoring matrix (PSSM) (Cheng *et al.*, 2008) of residue pair and intervening sequence: The PSSM was obtained using PSI-BLAST (Altschul *et al.*, 1997) search against the non-redundant protein sequence database at NCBI (released until December 26, 2014), with the substitution matrix, round of iteration and *E*-value chosen as BLOSUM62, 10 and 0.001, respectively. The smoothed PSSM uses the sum of PSSM score within a window of odd size (sw) to include local environmental effect in the description of evolutionary conservation of the central residue. Here, we introduced smoothed PSSM into residue contact prediction, considering its superior performance over the traditional PSSM in the prediction of RNA-binding residues of proteins (Cheng *et al.*, 2008; Xiong *et al.*, 2015).
3. Natural vector of intervening sequence: The 60D natural vector was extracted in the same way as Section 2.4.
4. Contact propensity of residue pair: Similar to Section 2.4, contact propensity for each residue pair was calculated by Equation (1), using proteins in the training dataset. In order to include the influence of secondary structures, residues were differentiated by both identities and secondary structure classification. Therefore, a total of nine propensity matrices were constructed.
5. Coevolutionary information of residue pair: This information was directly obtained from plmDCA scores that were extracted from ≤0.7*L* homologous sequences (see Section 2.4).

In addition, we incorporated popular features from previous studies:

1. Secondary structure information of residue pair and intervening sequence: obtained from prediction by SSpro (Magnan and Baldi, 2014).

2. Solvent accessibility of residue pair and intervening sequence: obtained from prediction by ACCpro (Magnan and Baldi, 2014).
3. Amino acid groups of residue pair and intervening sequence: based on the study of ProC_S3 (Li *et al.*, 2011), the 20 amino acids were divided into seven groups: (a) Arg and Lys; (b) Asp and Glu; (c) Ala and Met; (d) Val, Ile and Leu; (e) Phe, Tyr and Trp; (f) Cys; (g) Gly, Ser, Asn, His, Pro, Thr and Gln. A binary vector was used to represent the state of each individual residue.
4. Length of intervening sequence: This feature was represented by a vector to describe the binary states within eight intervals (24–28, 29–33, 34–38, 39–43, 44–48, 49–58, 59–68, 69–78, 79–88 and $\geq$89). Again, the intervals were adjusted based on the distribution of proteins in the training dataset.
5. Secondary structure composition of entire protein: calculated as the percentage of each type in the predicted secondary structure sequence.
6. Solvent accessibility composition of entire protein: calculated as the percentage of each type (buried or exposed) in predicted profile of solvent accessibility.
7. Amino acid composition of entire protein: calculated as the occurring frequencies of all 20 amino acids.
8. Length of entire protein: Again, this feature was represented by a vector of binary states in four intervals ($\leq$80, 81–160, 161–240 and $\geq$241).

Combining all features, we eventually constructed a 1224D complete feature vector to describe the property of each residue pair. Here, besides the DBN parameters (see Section 2.2), values of three window sizes (*w*, *cw* and *sw*) also need optimization. Their optimal values were chosen as 13, 3 and 5, respectively (see Section 3.3 for details). Subsequently, the complete feature set was optimized to a 758D optimal feature subset using the group MCP. The final model of DeepRCon was built using the optimal feature subset.

In the performance evaluation, we adopted the conventional measures in CASP competition: accuracy (Acc) and distance distribution (X_d). Acc refers to the fraction of correct predictions within all predicted contacts, and X_d quantifies the difference between the distance distribution of predicted contacting residue pairs and that of all residue pairs in native protein structure:

$$\mathrm{Acc} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \tag{5}$$

$$X_\mathrm{d} = \sum\nolimits_{i=1}^{15} \frac{Pp_i - Pa_i}{i}, \tag{6}$$

where *TP* and *FP* refer to the correctly and incorrectly predicted contacts, respectively. Pp_i is the percentage of predicted contact pairs within a distance interval of [4(i−1), 4i], and Pa_i is corresponding metric for all residue pairs in the native structure. Large positive X_d usually reflects good performance (a random prediction corresponds to $X_\mathrm{d} = 0$). The measures were computed for the top *L*/5, top *L*/10 and best five predicted contacts. As the most widely used evaluator for residue contact prediction, Acc of the top *L*/5 predicted contacts was chosen as our primary evaluation criterion for parameter optimization.

2.6 Refinement of contact map

As described earlier, both DeepCCon and DeepRCon were deliberately constructed to reduce the reliance on sequence abundance. Accordingly, the performance of DeepRCon is robust upon sequence abundance (see Section 4 for details). In practical contact prediction, however, sequence information should be maximally utilized, especially for protein targets of large-families. Therefore, we constructed a DBN model to effectively combine the prediction results of DeepRCon and a pure coevolution-based predictor CCMpred (Seemayer *et al.*, 2014). In specific, this refinement step predicts the contact probability of a target residue pair (*i*, *j*) based on the number of available homologous sequences in MSA as well as the *rw* × *rw* matrices of output contact maps from both DeepRCon and CCMpred that were centered at position (*i*, *j*). The window size *rw* was optimized to 9. All parameters of this model were optimized based on the training dataset.

3 Results

3.1 Database analysis on sequence abundance

Despite the success in the past years, methods that predict residue contacts based on coevolutionary information invariantly require the presence of abundant homologous sequences in MSA to effectively remove the noises caused by information transitivity, and thus may be useful only for the structure prediction of a small fraction of proteins (Ekeberg *et al.*, 2014; Jones *et al.*, 2012; Kamisetty *et al.*, 2013). Here, we investigated the sequence abundance in our training dataset that contained one representative domain from each SCOPe family. The cumulative distribution function (CDF) indicates that ~60% protein families have < 1*L* homologous sequences (Supplementary Fig. S2), a level at which coevolution-based methods become powerless (Kamisetty *et al.*, 2013). In order to expand the usefulness of our method, we elevated the noise level of coevolutionary information in our model training, by artificially reducing the number of homologous sequences in plmDCA calculation. In specific, the number of effective sequences was strictly limited to $\leq$0.7*L* in the extraction of coevolutionary-information-related features of both training and testing datasets. The cutoff of 0.7*L* was carefully chosen as the value at which CDF equals to 0.5 (i.e. the median number of sequences among all protein families) to favor proteins of small families.

3.2 Coarse contact prediction

In our datasets, the contacting SSE pairs (parallel and anti-parallel) are greatly outnumbered by no-contact ones. Because the former provides more useful information for the subsequent prediction of residue contacts, we ignored the latter in the performance evaluation of coarse contact prediction, in order to avoid the unwanted biases in evaluation.

Using the complete feature set, DeepCCon achieves an *F*-measure of 49.33%, a PPV of 55.88% and a TPR of 44.58% using 5-fold cross validation on the training dataset, when comprehensively considering the parallel and anti-parallel SSE pairs (Supplementary Table S1). In contrast, the models constructed with the same feature set but using two popular conventional learning technologies (random forest and back-propagation neural network as tested here) are less powerful (Supplementary Table S1), thereby reinforcing the great advance brought by deep learning technique. The complete feature set was designed by introducing a number of new features into those adopted by CCMAPpro (the coarse contact predictor of CMAPpro). To evaluate the effect of new features, we re-trained the model using features from CCMAPpro only for comparison. Results of 5-fold cross validation (Supplementary Table S2) suggest that our newly proposed features make a significant contribution in *F*-measure (by 4.50%).

Despite the improvement in performance, incorporation of new features raises the feature dimension from 271 to 391. To reduce computational complexity, we carried out feature selection to find the optimal feature subset, which effectively reduced the feature dimension (from 391 to 133) and mildly improved the model performance at the same time (Supplementary Table S2). In addition, three out of the six features that were retained in the optimal feature set are new features (Supplementary Table S3). The final DeepCCon model was thus constructed using the optimal feature subset.

The performance was further evaluated on the independent testing dataset. Considering both parallel and anti-parallel contacts, DeepCCon achieves an *F*-measure of 51.75% with a PPV of 57.37% and a TPR of 47.14%. The steady performance of DeepCCon on the training and testing datasets for all tested feature sets excludes the presence of over-training (compare Table 1 and Supplementary Table S2).

In the above model, secondary structure information was obtained unambiguously from DSSP (Kabsch and Sander, 1983) calculation to simplify parameter optimization. In practice, this information has to be predicted from primary sequence. To evaluate this effect, DeepCCon was rebuilt using secondary structures predicted by SSpro (Magnan and Baldi, 2014) with all tunable parameters unchanged. The new model only shows slightly weakened performance (~2% reduction in *F*-measure) on both training (Supplementary Table S4) and testing datasets (Supplementary Table S5), and therefore can still provide comparably useful information for the subsequent residue contact prediction.

3.3 Residue–residue contact prediction

In the process of model training, since the non-contact pairs are considerably more abundant than the contact pairs (positive case: ~1.35%), we randomly selected 3% of all samples, which included all positive samples, to deal with the seriously unbalanced training set. With the complete feature set, three window sizes (*w*, *cw* and *sw*) of DeepRCon were first optimized from 1 to 19, 1 to 11 and 1 to 19, respectively, by 5-fold cross validation on the training dataset, with Acc of the top *L*/5 predicted contacts chosen as the main evaluator. Considering the high computational complexity, optimization was carried out sequentially, in the order of *w*, *cw* and *sw*, with the un-optimized parameters temporarily set to 1. Finally, the optimal values of *w*, *cw* and *sw* were set to 13, 3 and 5, respectively (Supplementary Fig. S3). The performance of DeepRCon at the same tested combinations of parameter values was also evaluated on the benchmark testing dataset (Supplementary Fig. S4). The consistence between profiles of the training and testing datasets indicate the robustness of our model. As shown in Supplementary Table S6, after parameter optimization, DeepRCon achieves an Acc of 35.32% and an X_{d} of 15.77% for the top *L*/5 predicted contacts using the complete feature set.

The complete feature set of DeepRCon has a feature dimension of 1224, contributed by 17 features. Although high dimensionality guarantees sufficient coverage on the encoded information, the noise and redundancy among features may impair the predictive power and raise computational complexity. Therefore, similar to the coarse contact prediction, we conducted feature selection on DeepRCon to obtain the optimal feature subset. As shown in Supplementary Tables S7, 11 out of 17 features were retained after this process. Not unexpectedly, 5 out of 6 newly proposed features, including the coarse contact information, remained in the optimal feature subset, which reinforces their strong discriminating power in the residue–residue contact prediction. As shown in Supplementary Table S6, feature selection successfully reduced the feature dimension from 1224 to 758, and more importantly, significantly improved the prediction performance of DeepRCon in respect of all evaluators using 5-fold cross validation on the training dataset. Particularly, the Acc and X_{d} of the top *L*/5 predicted contacts were enhanced to 38.65 and 17.07%, respectively. To quantify the contribution of coarse contact information and coevolutionary information in the performance of DeepRCon, we removed each feature from the optimal feature subset respectively and reevaluated the model performance. The results show that coarse contact information and coevolutionary information are both essential features, contributing to Acc of the top *L*/5 predicted contacts by 2.42 and 3.93%, respectively (Supplementary Table S6). We also evaluated the relative contributions of the other features using the same method, and the detailed information is shown in Supplementary Table S8.

The performance of DeepRCon was further evaluated on the independent testing dataset (Table 2), with all tunable parameters fixed at their optimal values. Again, the steady performance of DeepRCon on the training and testing datasets supports robustness of our model (compare Table 2 and Supplementary Table S6). Specifically, the model built with the optimal feature subset achieves an Acc of 39.12% and an X_{d} of 16.64% for the top *L*/5 predicted contacts. On the testing dataset, the contributions of coarse contact information (3D) and coevolutionary information (1D) to Acc of the top *L*/5 predicted contacts were evaluated as 3.51 and 4.63%, respectively, an amazing level considering the total 758 dimensions in the optimal feature subset.

3.4 Refinement of contact prediction

The refinement step integrates more thoroughly derived coevolutionary information from CCMpred as well as predicted contact information of nearby residue pairs. As shown in Table 3, due to the inclusion of many large-family proteins in the test dataset, CCMpred that made use of all available homologous sequences remarkably outperforms DeepRCon that was trained using limited sequence information. After the refinement step, however, the final

Table 1. Performance of DeepCCon built with different feature sets on the benchmark testing dataset

Feature set	FD	Parallel and Anti-parallel			Parallel		Anti-parallel	
		F-measure	PPV	TPR	PPV	TPR	PPV	TPR
CCMAPpro	217	46.01	58.66	37.85	44.77	18.22	62.58	48.41
Complete	391	50.09	54.98	46.00	41.64	29.75	60.67	54.74
Optimal	133	51.75	57.37	47.14	42.43	32.95	64.76	54.76

Note: 'FD' represents 'Feature dimension'. The evaluating measures are represented as percentages.

Table 2. Performance of DeepRCon built with different feature sets on the benchmark testing dataset

Feature set	FD	Acc (%)			X_{d} (%)		
		L/5	*L*/10	5	*L*/5	*L*/10	5
Complete	1224	36.16	40.73	45.37	15.80	17.16	18.36
Optimal	758	39.12	43.63	49.01	16.64	17.99	19.66
Optimal (-CCI)	755	35.61	39.61	45.44	15.33	16.47	18.18
Optimal (-CI)	757	34.49	38.47	41.50	15.38	16.38	17.28

Note: 'FD' represents 'Feature dimension'. CCI and CI stand for coarse contact information and coevolutionary information respectively. Here, '(-CCI)' and '(-CI)' mean that the corresponding features are removed from the optimal feature subset.

program DeepConPred shows significant improvement over both DeepRCon and CCMpred, which thus indicates the effectiveness of information integration by this step.

3.5 Evaluation on the CASP protein sets

The performance of our method was then evaluated on the latest two CASP datasets (CASP10 and CASP11). To prevent the involvement of information from homologous templates in model building, we removed all proteins in the training dataset that had ≥20% sequence identity to any CASP targets tested here and reconstructed the model using the updated training dataset. On the CASP10 set, DeepConPred reaches an Acc of 59.33, 64.39 and 70.00% for the top *L*/5, top *L*/10 and top 5 predictions, respectively (Table 4). On the CASP11 set, the corresponding prediction accuracies drop to 49.97, 54.01 and 59.81%, respectively (Table 4). In comparison with the top 20 predictors reported at the CASP website (http://predictioncenter.org/) that were selected by the CASP official rank based on the results of top *L*/5 long-range contact predictions, DeepConPred exhibits significant improvement on both CASP10 and CASP11 sets, in respect of all evaluators (Supplementary Tables S9 and S10). The conclusion generally holds for free modeling (FM) targets (Supplementary Tables S11 and S12), despite the comparable performance of DeepConPred and CONSIP2 in the CASP11 set.

Most recently, Yang *et al.* (2016) presented a novel method R_2C and claimed an overwhelming improvement in prediction performance. According to the published numbers, DeepConPred outperforms R_2C by 14.53, 14.19 and 11.00% for the top *L*/5, top *L*/10 and top 5 predictions on CASP10 set, and 12.37, 12.01 and 10.91% for the top *L*/5, top *L*/10 and top 5 predictions on CASP11 set, respectively.

We also made an elaborate evaluation of DeepConPred (against R_2C) on two sets of hard CASP targets. The first set is composed of all FM and template-based modeling-hard (TBM-hard) targets according to the official definition of CASP, while the second set is collected from the R_2C paper, in the definition that the first models predicted by the best half of participating servers should have an average TM-score <0.5 for a hard CASP target. The performance of R_2C on the first set was evaluated through the R_2C web server, while the results on the second set were directly obtained from the published paper.

On the first set of hard targets, DeepConPred outperforms R_2C in every evaluating category. In specific, as compared with R_2C, DeepConPred improves the Acc and X_d of the top *L*/5 predicted contacts by 13.67 and 6.43% on CASP10 set, and by 7.58 and 5.01% on CASP11 set, respectively (Table 5). Similar results were obtained on the second set of hard targets (Table 6), where DeepConPred improves over R_2C by 11.00, 13.37 and 9.74% for the top *L*/5, top *L*/10 and top 5 predictions on CASP10 set, and 4.22, 5.36 and 6.67% for the top *L*/5, top *L*/10 and top 5 predictions on CASP11 set. These results thus strongly demonstrate the great advantage of DeepConPred in predicting the long-range residue–residue contacts of hard targets.

4 Discussion

DeepCCon and DeepRCon were constructed using the ≤0.7*L* homologous sequences to ensure robustness for small-family proteins. Here, we validated the model robustness upon sequence abundance on the CASP10 and CASP11 sets, by feeding the models with coevolutionary information calculated from all versus ≤0.7*L* homologous sequences. As shown in Table 7, changes caused by the number of effective homologous sequences are tiny and insignificant for each evaluator, which thus confirms the robustness of DeepCCon and DeepRCon. We speculate that the robustness may arise from the small proportion of coevolutionary-information-related features in the high dimensionality of the feature space (see CI and CCI in Table 2). Notably, it is the model robustness of DeepCCon and DeepRCon that lays foundation for the great improvement in the subsequent refinement step, because sufficient information complementarity is present between the contact maps predicted by DeepRCon and by the pure coevolution-based CCMpred. In this

Table 3. Comparison of the long-range residue–residue contact prediction obtained through different modules of DeepConPred on the benchmark testing dataset

Module	Acc (%)			X_d (%)		
	L/5	*L*/10	5	*L*/5	*L*/10	5
DeepRCon	39.12	43.63	49.01	16.64	17.99	19.66
CCMpred	61.81	66.13	69.42	22.02	23.42	24.72
DeepConPred	69.59	74.28	77.71	24.39	25.52	26.55

Table 4. Comparison of the long-range residue–residue contact prediction on the 123 CASP10 and 105 CASP11 targets

Dataset	Method	Acc (%)			X_d (%)		
		L/5	*L*/10	5	*L*/5	*L*/10	5
CASP10	R_2C	44.8	50.2	59.0	NA	NA	NA
	DeepConPred	59.33	64.39	70.00	22.47	23.39	24.94
CASP11	R_2C	37.6	42.0	48.9	NA	NA	NA
	DeepConPred	49.97	54.01	59.81	19.72	20.64	22.04

Note: The results of R_2C are obtained from the R_2C paper. 'NA' represents the lack of data in the R_2C paper.

Table 5. Comparison of the long-range residue–residue contact prediction on the 23 CASP10 and 38 CASP11 hard targets as defined by the composition of FM and TBM-hard targets in CASP

Dataset	Method	Acc (%)			X_d (%)		
		L/5	*L*/10	5	*L*/5	*L*/10	5
CASP10	R_2C	24.50	25.05	33.04	10.85	12.78	15.05
	DeepConPred	38.17	41.49	46.09	17.28	18.30	19.83
CASP11	R_2C	25.53	29.59	37.89	10.00	10.12	13.03
	DeepConPred	33.11	36.69	42.11	15.01	15.62	16.88

Note: The results of R_2C were obtained from the R_2C online server.

Table 6. Comparison of the long-range residue–residue contact prediction on the 35 CASP10 and 48 CASP11 hard targets defined in R_2C paper

Dataset	Method	Acc (%)			X_d (%)		
		L/5	*L*/10	5	*L*/5	*L*/10	5
CASP10	R_2C	29.2	30.4	39.4	NA	NA	NA
	DeepConPred	40.20	43.77	49.14	16.48	17.58	19.09
CASP11	R_2C	25.6	27.3	30.0	NA	NA	NA
	DeepConPred	29.82	32.66	36.67	13.48	14.04	15.05

Note: The results of R_2C are obtained from the R_2C paper. 'NA' represents the lack of data in the R_2C paper.

Table 7. Performance evaluation of DeepRCon using different input sources of coevolutionary information on the 123 CASP10 and 105 CASP11 targets

Dataset	Number of homologous sequences used	Acc (%)			X_d (%)		
		L/5	*L*/10	5	*L*/5	*L*/10	5
CASP10	0.7*L*	39.58	45.15	55.41	17.51	19.12	21.85
	All	40.35	46.48	55.92	18.13	19.79	22.23
CASP11	0.7*L*	32.90	37.22	44.52	15.39	16.91	19.41
	All	33.29	37.42	45.27	15.49	17.26	19.72

Table 8. Comparison of precision (%) for long-range contact predictions with the top 20 official results on the CASP12 set

Method	L/5	L/2	10
Deepfold-Contact	59.25	51.09	63.37
naive	59.21	50.91	63.80
RaptorX-Contact	58.66	49.87	64.47
MetaPSICOV	56.51	47.58	60.98
DeepConPred	54.84	43.78	58.89
iFold_1	54.35	45.54	58.33
PconsC31	53.27	43.09	59.68
FLOUDAS_SERVER	51.94	48.27	53.33
raghavagps	51.91	44.18	54.55
Pcons-net	51.43	42.27	57.45
MULTICOM-CONSTRUCT	50.31	41.18	54.89
MULTICOM-CLUSTER	50.14	40.87	55.11
Yang-Server	48.40	40.46	54.57
RBO-Epsilon	47.02	36.74	55.43
IGBteam	46.23	37.41	52.02
Zhang_Contact	45.19	37.58	48.91
FALCON_COLORS	44.62	36.19	50.96
Distill	44.58	40.59	46.49
PconsC2	44.49	35.95	51.31
Myprotein-me	43.14	36.00	49.42
AkbAR	39.99	31.22	46.94

Note: The top 20 official CASP12 results are sorted by the official CASP ranks based on results of the top *L*/5 long-range contact predictions.

respect, the special model design and hierarchical architecture allows our program DeepConPred to take into account proteins from both small and large families.

During the peer review, 54 CASP12 targets were released with full native contact information, which allows preliminary performance evaluation (Supplementary Table S13). On this subset of CASP12 targets, DeepConPred achieves comparable performance to the best models reported by CASP12 official website (Table 8 for succinct results and Supplementary Table S14 for full details). In an in-depth analysis, we compared our algorithm against MetaPSICOV, one of the most successful and most famous contact predictors in the past CASPs. Despite the similar levels of average prediction accuracy, DeepConPred shows advantages on a number of CASP12 targets, which are α or $\alpha\beta$ proteins with relatively short chain lengths (~100 amino acids). Notably, these small-sized proteins are hard targets in long-range contact prediction, due to their limited numbers of native contacts. We found that DeepConPred can outperform most of other methods on these proteins. For example, for two hard targets, T0862-D1 and T0943-D1, DeepConPred ranks as the best and second-to-the-best respectively among all CASP12 participating groups in terms of the accuracy of the top *L*/5 predicted long-range contacts. As shown by the side-by-side comparisons of DeepConPred and MetaPSICOV on the native long-range contact maps of these two proteins, DeepConPred includes more native contacts of the helical or strand regions within the top *L*/5 predictions with fewer false positives (Supplementary Fig. S5). Therefore, despite the comparable performance in general, DeepConPred can provide contact information that cannot be captured by other state-of-the-art methods, especially for those small-sized hard targets. Moreover, Zhang et al. demonstrated the positive effects of predicted long-range contact information in practical protein structure prediction when the precision of contact prediction exceeded 22% (Zhang *et al.*, 2003). Similarly, a number of previous researches (e.g. CONFOLD (Adhikari *et al.*, 2015), PconsFold (Michel *et al.*, 2014) and EVfold (Marks *et al.*, 2011)) reported that contact predictors with similar levels of prediction accuracy to DeepConPred could facilitate the prediction of native protein conformation. Therefore, we believe that our contact prediction algorithm DeepConPred will benefit the field of protein structure prediction.

5 Conclusion

In this work, we proposed two novel models DeepCCon and DeepRCon, which could be utilized in a pipeline to improve the prediction of long-range residue–residue contacts in combination with a refinement model. Besides the introduction of powerful deep learning technique, we improved the model performance by proposing novel features and by identifying the optimal feature subset. Moreover, the overall architecture was designed to consider protein targets of both small and large families. Evaluation on CASP proteins showed usefulness of our method in protein structure prediction.

Acknowledgements

We thank Mr Sai Zhang for the assistance in the DBN model design and Dr. Hongbin Shen for kindly providing the list of hard CASP10 and CASP11 targets defined in the R_2C article.

Funding

This work has been supported by the funds from the National Natural Science Foundation of China (Nos. 31670723, 31470033 and 61472205) and from the Beijing Innovation Center of Structural Biology.

Conflict of Interest: none declared.

References

Adhikari,B. *et al.* (2015) CONFOLD: residue-residue contact-guided ab initio protein folding. *Proteins*, **83**, 1436–1449.

Altschul,S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Björkholm,P. *et al.* (2009) Using multi-data hidden Markov models trained on local neighborhoods of protein structure to predict residue–residue contacts. *Bioinformatics*, **25**, 1264–1270.

Breheny,P. and Huang,J. (2009) Penalized methods for bi-level variable selection. *Stat. Interface*, **2**, 369–380.

Cheng,C.-W. *et al.* (2008) Predicting RNA-binding sites of proteins using support vector machines and evolutionary information. *BMC Bioinformatics*, **9**, S6.

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

Di Lena,P. *et al.* (2012) Deep architectures for protein contact map prediction. *Bioinformatics*, **28**, 2449–2457.

Dunn,S.D. *et al.* (2008) Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*, **24**, 333–340.

Eickholt,J. and Cheng,J. (2012) Predicting protein residue–residue contacts using deep networks and boosting. *Bioinformatics*, **28**, 3066–3072.

Ekeberg,M. *et al.* (2014) Fast pseudolikelihood maximization for direct-coupling analysis of protein structure from many homologous amino-acid sequences. *J. Comput. Phys.*, **276**, 341–356.

Ekeberg,M. *et al.* (2013) Improved contact prediction in proteins: using pseudolikelihoods to infer Potts models. *Phys. Rev. E*, **87**, 012707.

Fox,N.K. *et al.* (2014) SCOPe: Structural Classification of Proteins—extended, integrating SCOP and ASTRAL data and classification of new structures. *Nucleic Acids Res.*, **42**, D304–D309.

Hinton,G.E. *et al.* (2006) A fast learning algorithm for deep belief nets. *Neural Comput*, **18**, 1527–1554.

Hinton,G.E. and Salakhutdinov,R.R. (2006) Reducing the dimensionality of data with neural networks. *Science*, **313**, 504–507.

Huang,J. *et al.* (2012) A selective review of group selection in high-dimensional models. *Stat. Sci.*, **27**, 481–499.

Jones,D.T. *et al.* (2012) PSICOV: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinformatics*, **28**, 184–190.

Jones,D.T. *et al.* (2015) MetaPSICOV: combining coevolution methods for accurate prediction of contacts and long range hydrogen bonding in proteins. *Bioinformatics*, **31**, 999–1006.

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

Kamisetty,H. *et al.* (2013) Assessing the utility of coevolution-based residue-residue contact predictions in a sequence- and structure-rich era. *Proc. Natl. Acad. Sci. USA*, **110**, 15674–15679.

Kliger,Y. *et al.* (2009) Peptides modulating conformational changes in secreted chaperones: from in silico design to preclinical proof of concept. *Proc. Natl. Acad. Sci. USA*, **106**, 13797–13801.

Kosciolek,T. and Jones,D.T. (2015) Accurate contact predictions using covariation techniques and machine learning. *Proteins*, **84**, 145–151.

Lee,B.-C. and Kim,D. (2009) A new method for revealing correlated mutations under the structural and functional constraints in proteins. *Bioinformatics*, **25**, 2506–2513.

Li,Y. *et al.* (2011) Predicting residue–residue contacts using random forest models. *Bioinformatics*, **27**, 3379–3384.

Little,D.Y. and Chen,L. (2009) Identification of coevolving residues and coevolution potentials emphasizing structure, bond formation and catalytic coordination in protein evolution. *PLoS One*, **4**, e4762.

Ma,J. *et al.* (2015) Protein contact prediction by integrating joint evolutionary coupling analysis and supervised learning. *Bioinformatics*, **31**, 3506–3513.

Magnan,C.N. and Baldi,P. (2014) SSpro/ACCpro 5: almost perfect prediction of protein secondary structure and relative solvent accessibility using profiles, machine learning and structural similarity. *Bioinformatics*, **30**, 2592–2597.

Marks,D.S. *et al.* (2011) Protein 3D structure computed from evolutionary sequence variation. *PLoS ONE*, **6**, e28766.

Michel,M. *et al.* (2014) PconsFold: improved contact predictions improve protein models. *Bioinformatics*, **30**, i482–i488.

Miller,C.S. and Eisenberg,D. (2008) Using inferred residue contacts to distinguish between correct and incorrect protein models. *Bioinformatics*, **24**, 1575–1582.

Misura,K.M.S. *et al.* (2006) Physically realistic homology models built with rosetta can be more accurate than their templates. *Proc. Natl. Acad. Sci. USA*, **103**, 5361–5366.

Monastyrskyy,B. *et al.* (2014) Evaluation of residue–residue contact prediction in CASP10. *Proteins*, **82**, 138–153.

Monastyrskyy,B. *et al.* (2015) New encouraging developments in contact prediction: Assessment of the CASP11 results. *Proteins*, **84**, 1–14.

Morcos,F. *et al.* (2011) Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proc. Natl. Acad. Sci. USA*, **108**, E1293–E1301.

Najafabadi,M.M. *et al.* (2015) Deep learning applications and challenges in big data analytics. *J. Big Data*, **2**, 1–21.

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

Schneider,M. and Brock,O. (2014) Combining physicochemical and evolutionary information for protein contact prediction. *PLoS One*, **9**, e108438.

Seemayer,S. *et al.* (2014) CCMpred—fast and precise prediction of protein residue–residue contacts from correlated mutations. *Bioinformatics*, **30**, 3128–3130.

Shao,Y. and Bystroff,C. (2003) Predicting interresidue contacts using templates and pathways. *Proteins*, **53**, 497–502.

Skolnick,J. *et al.* (2004) Development and large scale benchmark testing of the PROSPECTOR_3 threading algorithm. *Proteins*, **56**, 502–518.

Skwark,M.J. *et al.* (2014) Improved contact predictions using the recognition of protein like contact patterns. *PLoS Comp. Biol.*, **10**, e1003889.

Tegge,A.N. *et al.* (2009) NNcon: improved protein contact map prediction using 2D-recursive neural networks. *Nucleic Acids Res.*, **37**, W515–W518.

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

Vassura,M. *et al.* (2008) Reconstruction of 3D structures from protein contact maps. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, **5**, 357–367.

Wang,Z. and Xu,J. (2013) Predicting protein contact map using evolutionary and physical constraints by integer programming. *Bioinformatics*, **29**, i266–i273.

Weigt,M. *et al.* (2009) Identification of direct residue contacts in protein–protein interaction by message passing. *Proc. Natl. Acad. Sci. USA*, **106**, 67–72.

Wu,S. and Zhang,Y. (2008) A comprehensive assessment of sequence-based and template-based methods for protein contact prediction. *Bioinformatics*, **24**, 924–931.

Xiong,D. *et al.* (2015) RBRIdent: an algorithm for improved identification of RNA-binding residues in proteins from primary sequences. *Proteins*, **83**, 1068–1077.

Xue,B. *et al.* (2009) Predicting residue–residue contact maps by a two-layer, integrated neural-network method. *Proteins*, **76**, 176–183.

Yang,J. *et al.* (2016) R2C: improving ab initio residue contact map prediction using dynamic fusion strategy and Gaussian noise filter. *Bioinformatics*, **32**, 2435–2443.

Yu,C. *et al.* (2013) Protein space: a natural method for realizing the nature of protein universe. *J. Theor. Biol.*, **318**, 197–204.

Zhang,G.-Z. and Huang,D.-S. (2004) Prediction of inter-residue contacts map based on genetic algorithm optimized radial basis function neural network and binary input encoding scheme. *J. Comput. Aid. Mol. Des.*, **18**, 797–810.

Zhang,Y. *et al.* (2003) TOUCHSTONE II: a new approach to ab initio protein structure prediction. *Biophys. J.*, **85**, 1145–1164.

Zhao,Y. and Karypis,G. (2005) Prediction of contact maps using support vector machines. *Int. J. Artif. Intell. Tools*, **14**, 849–865.