Machine Learning approaches for Contact Maps prediction in CASP9 experiment

Giuseppe Tradigo[1], Pierangelo Veltri[1], and Gianluca Pollastri[2]

[1] University Magna Græcia of Catanzaro, Catanzaro 88100, Italy,
{gtradigo,veltri}@unicz.it
[2] University College Dublin, Ireland,
gianluca.pollastri@ucd.ie

(Extended Abstract)

Abstract. Residue contact maps are bi-dimensional data structures that encode the three-dimensional structure of a protein by storing the presence of contacts among protein backbone residues. Contact maps have a key role in most state-of-the-art protein structure prediction pipelines, i.e. the prediction of three dimensional space conformation of aminoacids composing the proteins. We have designed a system (XXStout-beta) for the prediction of residue contact maps from the sequence of amino acids composing the protein. The system is based on Recursive Neural Networks, which are capable of learning an input-output mapping from sets of examples. Moreover data structures and loading/unloading algorithms have been designed for efficiently managing contact maps in primary and secondary memory. XXStout-beta has performed well at the latest CASP9 world-wide protein structure prediction competition, and is integrated in the public, high-throughput structure prediction server Distill[3]. In this paper we present informally results of XXStout-beta in CASP9 competition.

1 Contact Map prediction

In Nature, the process of protein folding is observed when a protein is synthesized in the cell. Although many factors may facilitate the folding process, the shape of a protein, with only limited exceptions is directly determined by its amino acid sequence, i.e. one can assume that the map between the sequence and the structure is a function. Finding this function is the so called Protein Folding Problem (*PFP*). PFP has been an hot topic for decades among a vast community of scientists. This has been tackled in various ways, ranging from physical simulations to knowledge-based methods such as machine learning. In the latter case one needs to observe a (sufficiently large) number of known protein structures and try to understand/learn the laws behind the folding process, or more simply the map between sequences and structure, either in part (e.g. sequence to secondary structure) or as a whole. Solving the PFP computationally

[3] http://distill.ucd.ie/distill/

is appealing, as experimental determination of the structure is a complex, time-consuming and expensive process, and as a consequence as of 2011 we only know the structures of approximately 70,000 proteins, or approximately one every 200 proteins for which we have revealed the sequence.

Every two years the CASP experiment challenges computational folding methods to predict the (then unknown) structure of several tens of proteins. Among other categories, CASP assesses the prediction of protein residue contact maps, i.e. the set of mutual distances between residues in a protein, quantised into two states (contact, for distances smaller than a threshold, and non-contact otherwise).

Correct contact maps have been shown to be lead to reasonably good 3D structures [5, 6], and predicted contact maps have been used for driving protein folding in the ab initio case (that is, when a protein is folded without relying on homology to another protein of known structure), for selecting and ranking folded protein models, for predicting folding times, protein domain boundaries, secondary structure, etc.

We have designed a novel predictor of protein residue contact maps. The predictor exploits a diverse, complex set of inputs, including the residue sequence, evolutionary information in the form of a profile of residue frequencies extracted from a multiple sequence alignment of homologous proteins (of unknown structure), predicted secondary structure, solvent accessibility and contact density and, most importantly, near and remote structural templates (when available) obtained by various methods. The predictor has two types of outputs: a simple contact/non-contact binary classification (as per CASP rules); a 4-class distance map. The latter output is used as a constraint to reconstruct 3-dimensional protein structures.

2 Neural Networks for Prediction

2.1 Methods

We predict contact and distance maps by 2D-RNNs (two-dimensional Recursive Neural Networks), which were previously described in [14] and [15]. This is a family of adaptive models for mapping two-dimensional matrices of variable size into matrices of the same size.

If $o_{j,k}$ is the entry in the j-th row and k-th column of the output matrix, and $i_{j,k}$ is the input in the same position, the input-output mapping is modelled as:

$$\begin{aligned}
o_{j,k} &= \mathcal{N}^{(O)}\left(i_{j,k}, h^{(1)}_{j,k}, h^{(2)}_{j,k}, h^{(3)}_{j,k}, h^{(4)}_{j,k}\right)\\
h^{(1)}_{j,k} &= \mathcal{N}^{(1)}\left(i_{j,k}, h^{(1)}_{j-1,k}, .., h^{(1)}_{j-s,k}, h^{(1)}_{j,k-1}, .., h^{(1)}_{j,k-s}\right)\\
h^{(2)}_{j,k} &= \mathcal{N}^{(2)}\left(i_{j,k}, h^{(2)}_{j+1,k}, .., h^{(2)}_{j+s,k}, h^{(2)}_{j,k-1}, .., h^{(2)}_{j,k-s}\right)\\
h^{(3)}_{j,k} &= \mathcal{N}^{(3)}\left(i_{j,k}, h^{(3)}_{j+1,k}, .., h^{(3)}_{j+s,k}, h^{(3)}_{j,k+1}, .., h^{(3)}_{j,k+s}\right)
\end{aligned}$$

$$
\begin{aligned}
h^{(4)}_{j,k} = \mathcal{N}^{(4)} &\left(i_{j,k}, h^{(4)}_{j-1,k}, .., h^{(4)}_{j-s,k}, h^{(4)}_{j,k+1}, .., h^{(4)}_{j,k+s} \right) \\
&j, k = 1, \ldots, N \\
&s = 1, \ldots, S
\end{aligned}
$$

where $h^{(n)}_{j,k}$ for $n = 1, \ldots, 4$ are planes of hidden vectors transmitting contextual information from each corner of the matrix to the opposite corner. We parametrise the output update, and the four lateral update functions (respectively $\mathcal{N}^{(O)}$ and $\mathcal{N}^{(n)}$ for $n = 1, \ldots, 4$) using five two-layered feed-forward neural networks, as in [15]. Stationarity is assumed for all residue pairs (j, k), that is the same parameters are used across all $j = 1, ..., N$ and $k = 1, ..., N$. Each of the 5 neural network contains its own individual parameters, that are not constrained to the ones of the other networks.

Since we are trying to predict both a 4-class map and a binary map, we model both classification problems within the same 2D-RNN. Hence the output $o_{j,k}$ will have two components:

$$o_{j,k} = (o^{(4)}_{j,k}, o^{(2)}_{j,k})$$

where $o^{(4)}_{j,k}$ is a vector of four numbers representing the estimated probabilities of residues j and k belonging to each of the four distance classes, and $o^{(2)}_{j,k}$ is the same for the two binary (contact vs. non-contact) classes. Both components are implemented by (independent) softmax units.

As modelled in the input-output mapping equations above, we use 2D-RNNs with *shortcut connections*. This means that a memory state depends explicitly on more that the memory state immediately previous to it along the direction of contextual propagation, i.e. the memory span is greater than one. This is effective because gradient-based learning in deep layered architectures suffers from the well known vanishing gradient problem [18]. Allowing shortcuts of length S (i.e. the memory state in position i depends directly on the state in position $i-S$) creates new paths of roughly $1/S$ of the length of the ones induced by 1-step memory dependencies, thus facilitating the transmission of contextual information over larger distances. Indeed, shortcut connections can be placed starting at any of the previous states $i - s$ for any $s \in 1, .., S$. A selective placement of shortcuts was used to produce near perfect secondary structure predictions in a bidirectional recurrent neural network when (i, s) represent native contacts [16]. Notice that increasing the number of shortcuts increases the parameters resulting in a model that may more easily overfit the data. Extending the shortcut idea beyond the 2D case or in any direction of contextual propagation is straightforward. Shortcut directions and patterns are not strictly constrained (so long as cycles are not introduced in the directed graph representing the network) and may even be learned.

The choice of input $i_{j,k}$ is an important factor for the algorithm. In the case of contact map prediction the simplest input is the amino acid symbols at (j, k). Different input encodings can be constructed to improve the algorithm. In the Input Design section we describe the input encoding we used in this study.

Training Learning proceeds by gradient descent by minimising the relative cross entropy between target and output. Since there are two independent output components (a 4-class and a binary one), the error is in fact the sum of two cross entropies, which are weighed equally. Careful management of the gradient must take place, not letting it be too small or too large: the absolute value of each component of the gradient is kept within the [0.1,1] range, meaning that it is set to 0.1 if it is smaller than 0.1, and to 1 if it is greater than 1. The learning rate is set to 0.0375 divided by the the total number of proteins in the dataset. The weights of the networks are initialised randomly.

Input format Input $i_{j,k}$ associated with the j-th and k-th residue pair contains primary sequence information, evolutionary information, structural information, and direct contact information derived from the PDB templates:

$$i_{j,k} = (i_{j,k}^{(E)}, i_{j,k}^{(T)}) \tag{1}$$

where, assuming that e units are devoted to evolutionary sequence information and structural information in the form of secondary structure[11, 10], solvent accessibility [11] and contact density [13]:

$$i_{i,j}^{(E)} = (i_{j,k}^{(1)^{(E)}}, \dots, i_{j,k}^{(e)^{(E)}}) \tag{2}$$

Template information is placed in the remaining t units:

$$i_{j,k}^{(T)} = (i_{j,k}^{(1)^{(T)}}, \dots, i_{j,k}^{(t)^{(T)}}) \tag{3}$$

Hence $i_{j,k}$ contains a total of $e + t$ components.

In this work $e = 58$. 20+20 units correspond to the frequencies of residues observed in the two columns j and k of the multiple sequence alignment. Structural information in the form of secondary structure (three classes), solvent accessibility (two classes), and contact density (four classes) for residue j and k are placed in the remaining 6,4 and 8 input units respectively.

For the template units we use $t = 5$, representing weighted contact class information from the templates and one template quality unit. Assume that $d_{j,k}^{(p)}$ is a 4-component binary vector encoding the contact class of the (j,k)-th residue pair in the p-th template. Then, if P is the total number of templates for a protein:

$$(i_{j,k}^{(1)^{(T)}}, \dots, i_{j,k}^{(4)^{(T)}}) = \frac{\sum_{p=1}^{P} w_p d_{j,k}^{(p)}}{\sum_{p=1}^{P} w_p} \tag{4}$$

where w_p is the weight attributed to the p-th template. If the sequence identity between template p and the query is id_p and the quality of a template (measured as X-ray resolution + R-factor/20 or 10 for NMR hits, as in [9]) is q_s, then the weight is defined as:

$$w_p = q_p id_p^3 \tag{5}$$

Taking the cube of the identity between template and query allows us to drastically reduce the contribution of low-similarity templates when good templates are available. For instance a 90% identity template is weighed two orders of magnitude more than a 20% one. In preliminary tests (not shown) this measure performed better than a number of alternatives.

The final unit of $i_{j,k}$, the quality unit, encodes the weighted average coverage and similarity of a column of the template profile as follows:

$$i_{j,k}^{(5)^{(T)}} = \frac{\sum_{p=1}^{P} w_p c_p}{\sum_{p=1}^{P} w_p} \tag{6}$$

where c_p is the coverage of the sequence by template p (i.e. the fraction of non-gaps in the alignment). Encoding template information for the binary maps is similar.

Ab initio based predictions use only the first part of the input, $i_{j,k}^{(E)}$ from equation 2, including secondary structure, solvent accessibility and contact density, although these are predicted ab initio. The template based predictions use the complete $i_{j,k}$ as input.

2.2 Data

We use two datasets to train our predictors. The first set (D1) is obtained from the January 2007 25% pdb_select list [9]. After processing and selection of proteins no longer than 200 residues, D1 contains 2,452 proteins (and 70 million residue pairs), which we divide into a training set of 1,978 instances, and a test set of 474. The dataset is further processed to generate maps between C_β atoms. We only use this dataset to predict binary C_β maps with a 8Å threshold, as per CASP rules. The second set is obtained from the October 2009 25% pdb_select list, containing 4,818 proteins, which become 3,645 (over 100 million residue pairs) after processing and selection of sequences no longer than 200 residues. This second set, in which C_α distances are taken into account, is split into 5 approximately equal parts and 5-fold cross validation trainings are run on it in two settings: prediction of 4-class distance maps without structural homologues (templates) as inputs, or free-modelling setting (FM); including templates, or template-based modelling setting (TBM).

For CASP9 contact map predictions we ensemble 15 models from 5 different trainings (with different structural parameters, such as shortcut lengths) on D1. The models trained on D2 feed into our 3D predictor Distill [13].

3 Experimental results

The trained systems, set up as web servers, took part to the CASP9 world-wide competition. Overall results of top participating servers for contact map prediction are reported in Table 1, during the CASP9 conference held in Asilomar, California in December 2010 and are available at [19]. According to the

Server	Group	N Targets	Z score	Metapredictor
MULTICOM-CLUSTER	2	25	1.258	
Infobiotics	51	28	1.073	
Distill	**214**	**28**	**0.880**	
SAM-T08-server	103	28	0.840	
ProC_S1	375	25	0.740	*n.a.*
MULTICOM-REFINE	119	26	0.674	
PSICON	422	28	0.628	*n.a.*
SMEG-CCP	391	27	2.391	yes
MULTICOM	490	27	2.388	yes
ProC_S3	138	24	1.011	yes
MULTICOM-CONSTRUCT	80	26	0.776	yes
SAM-T06-server	244	25	0.678	yes

Table 1. The top 12 groups partecipating at the IX edition of CASP 2010 in the contact prediction category, from the official CASP assessment.

official assessment our system was one of the top three standalone predictors, and the second best that submitted all the proteins that were considered in the evaluation. The Z score value for Distill with XXStout-beta server has been calculated and evaluated in 0.880. The Z score is a standard performance measure for protein structure prediction.

References

1. Murzin A. G., Metamorphic Proteins, *Science*, 230, 1725-26, 2008
2. Fraenkel A. S., Complexity of protein folding, *Bulletin of Mathematical Biology*, 55(6): 1199-1210, 1993
3. Hart W. E., Istrail S., Robust Proofs of NP-Hardness for Protein Folding: General Lattices and Energy Potentials, *Journal of Computational Biology*, 4(1): 1-22, 1997
4. Crescenzi P., Goldman D., Papadimitriou C., Piccolboni A., Yannakakis M., On the Complexity of Protein Folding, *Journal of Computational Biology*, 5(3): 423-465, 1998
5. Vendruscolo M., Kussell E., Domany E., Recovery of protein structure from contact maps, *Folding and Design*, 2(5): 295-306, 1997
6. Walsh I., Baú D., Martin A.J.M., Mooney C., Vullo A., Pollastri G., Ab initio and template-based prediction of multi-class distance maps by two-dimensional recursive neural networks, *BMC Structural Biology*, 9(1): 5, 2009
7. Zhang Y., Skolnick J., Scoring function for automated assessment of protein structure template quality, *Proteins*, 57: 702-710, 2004
8. Berman H., Westbrook J., Feng Z., Gilliland G., Bhat T., Weissig H., Shindyalov I., Bourne P., The Protein Data Bank, *Nucl Acids Res*, 28: 235-242, 2000
9. Griep S., Hobohm U., PDBselect 1992-2009 and PDBfilter-select, *Nucleic Acids Research*, 38(1), 2009
10. Pollastri G., McLysaght A., Porter, a new, accurate server for protein secondary structure prediction, *Bioinformatics*, 21(8): 1719-1720, 2005

11. Mooney C., Pollastri G., Beyond the Twilight Zone: Automated prediction of structural properties of proteins by recursive neural networks and remote homology information, *Proteins*, 77(1): 181-90, 2009
12. Mooney C., Vullo A., Pollastri G., Protein Structural Motif Prediction in Multidimensional Φ-Ψ Space leads to improved Secondary Structure Prediction, *Journal of Computational Biology*, 13(8): 1489-1502, 2006
13. Baú D., Martin A.J.M., Mooney C., Vullo A., Walsh I. Pollastri G., Distill: a suite of web servers for the prediction of one-, two- and three-dimensional structural features of proteins, *BMC Bioinformatics*, 7(1): 402, 2006
14. Pollastri G., Baldi P., Prediction of Contact Maps by Recurrent Neural Network Architectures and Hidden Context Propagation from All Four Cardinal Corners, *Bioinformatics*, 18(Suppl 1): S62-S70, 2002
15. Baldi P., Pollastri G., The Principled Design of Large-Scale Recursive Neural Network Architectures - DAG-RNNs and the Protein Structure Prediction Problem, *Journal of Machine Learning Research*, 4(Sep): 575-602, 2003
16. Ceroni A., Frasconi P., Pollastri G., Learning Protein Secondary Structure from Sequential and Relational Data, *Neural Networks*, 18(8):1029-39, 2005.
17. Altschul S., Madden T., Schaffer A., Gapped BLAST and PSI-BLAST: a new generation of protein database search programs, *Nucleic Acids Research*, 25: 3389-3402, 1997
18. Bengio Y., Simard P., Frasconi P., Learning Long-Term Dependencies with Gradient Descent is Difficult, *IEEE Transactions on Neural Networks*, 5: 157-166, 1994
19. Casp9 website, official contact maps predictors assessment, *http://www.predictioncenter.org/casp9/doc/presentations/CASP9_RR.pdf*, 2011