Exploiting Long-Range Dependencies in Protein β-Sheet Secondary Structure Prediction

Yizhao Ni and Mahesan Niranjan

ISIS Group, School of Electronics and Computer Science
University of Southampton, U.K
Yizhao.NI@googlemail.com,
mn@ecs.soton.ac.uk

Abstract. We investigate if interactions of longer range than typically considered in local protein secondary structure prediction methods can be captured in a simple machine learning framework to improve the prediction of β sheets. We use support vector machines and recursive feature elimination to show that the small signals available in long range interactions can indeed be exploited. The improvement is small but statistically significant on the benchmark datasets we used. We also show that feature selection within a long window and over amino acids at specific positions typically selects amino acids that are shown to be more relevant in the initiation and termination of β-sheet formation.

Keywords: Protein Secondary Structures, β-Sheet, Feature Selection, Machine Learning.

1 Introduction

Predicting the secondary structure of proteins from their amino acid sequences using machine learning methods has been of interest for several decades. Examples of early work in the topic include that of Qian and Sejnowski [12]. Work in the area appears to have stabilized over the years, with the availability of several stable web based prediction servers (e.g. `JPred` [2] and its previous incarnations). An overview of development in the area approximately halfway through the period of the above papers is given by Rost [13].

The basic strategy for prediction of secondary structure has largely been to encode a local window of amino acids (usually $11-15$), using a one in Ω binary coding method, where $|\Omega| = 20$, leading to an input space of dimension in the range $220-300$. The output space is usually three dimensional predicting if the secondary structure at the centre of the window (namely the *central residue*) is an α-helix, β-sheet or of an unspecified structure, usually referred to as coil. A mapping between such a multivariate input and the three dimensional output space can be learned by a machine learning technique of one's choice, in which artificial neural networks of the multi-layer perceptron type [11] is the most popular in the literature.

T.M.H. Dijkstra et al. (Eds.): PRIB 2010, LNBI 6282, pp. 349–357, 2010.

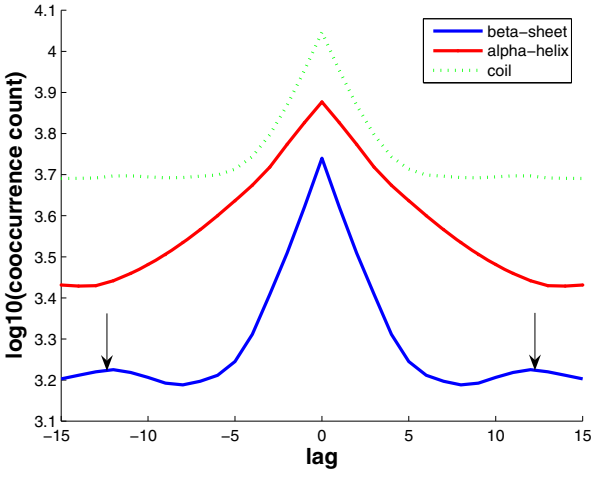

Fig. 1. Distribution of co-occurrences of secondary structures separated by a lag from the central residue of the input window for the three secondary structure classes. Arrows show that for the β-sheet there is some long range interaction outside the usually considered analysis lengths.

Of the three classes usually considered for predictions in this setting, it is known that β-sheets are the most difficult to predict. This observation is usually attributed to the fact that sheet structures are formed by interactions of longer range than is accommodated within the local windows. The obvious solution to dealing with this by increasing the window length is usually not expected to be successful because with each additional position included, we increase the dimensionality by 20, and a corresponding increase in the amount of training data will be required.

In this paper we explore the possibility of longer windows for β-sheet prediction with feature subset selection to keep the input dimensionality low. We first observe, using co-occurrence counts, that β-sheets contain a small amount of long range dependencies. Fig. 1 shows this co-occurrence counts for the three classes of secondary structures, where we plot the counts at different position lags from the central residue to a logarithmic scale. We observe a small but noticeable difference between the β-sheet and the other two classes. Motivated by this observation, we show that *recursive feature elimination* picks up a small subset of amino acids and their positions in the window to achieve a quantifiable improvement in prediction accuracies.

2 Materials and Methods

2.1 Kernel Classifiers

Let us denote the protein sample pool as $\mathcal{S} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{m}$, where $\mathbf{x}_i = (x_i^1, x_i^2, \ldots, x_i^{N_i})$ is the i-th amino acid sequence with N_i denoting the length of the

sequence, $x_i^j \in \Omega$ and Ω represents the set of amino acids appeared in $\mathcal{S}$. Similarly, the i-th secondary structure sequence is denoted by $\mathbf{y} = (y_i^1, y_i^2, \ldots, y_i^{N_i})$ where $y_i^j \in \{1, -1\}$ with 1 representing β-sheet (E) and -1 otherwise ($\sim$E). Whenever this can be done without loss of clarity, each example (x_i^j, y_i^j) is also abbreviated as (x_n, y_n), where the number of examples is defined by $N = \sum_{i=1}^{m} N_i$.

In order to solve the presented binary classification problem (i.e. E and $\sim$E), the support vector machine (SVM) technique is applied. It learns a linear operator $\mathbf{w}$ by solving the following optimisation problem

$$\begin{array}{ll} \min\limits_{\mathbf{w}, w_0, \boldsymbol{\xi}} & \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\mathbf{1}^T\boldsymbol{\xi} \\ s.t. & y_n\big(\mathbf{w}^T\phi(x_n) + w_0\big) \geq 1 - \xi_n \quad n = 1, \ldots, N \\ & \boldsymbol{\xi} := \{\xi_n | \xi_n \geq 0,\, n = 1, \ldots, N\} \end{array} \tag{1}$$

such that a new amino acid residue x has the prediction $f(x) = sgn(\mathbf{w}^T\phi(x))$, where $\phi(x) \in \mathbb{R}^D$ is an embedding feature function which will be specified in Section 2.2 and $sgn(\cdot)$ indicates the sign of the expression.

In addition, one can turn to solving the dual representation of (1)

$$\begin{array}{ll} \max\limits_{\boldsymbol{\alpha}} & -\frac{1}{2}\boldsymbol{\alpha}^T\mathbf{K}_{\mathbf{xy}}\boldsymbol{\alpha} + \mathbf{1}^T\boldsymbol{\alpha} \\ s.t. & \mathbf{y}^T\boldsymbol{\alpha} = 0 \\ & \boldsymbol{\alpha} = \{\alpha_n | 0 \leq \alpha_n \leq C,\, n = 1, \ldots, N\} \end{array} \tag{2}$$

which allows the use of kernels

$$\mathbf{K}_{\mathbf{xy}} = \{y_k y_l \phi(x_k)^T \phi(x_l) :\ k, l = 1, \ldots, N\}, \tag{3}$$

and it is expected to provide more flexibility in the feature expression.

2.2 Feature Extraction

Following [6,9], we consider the *position-dependent residue* features extracted from the amino acid sequences. Mathematically, the feature expression is given by the formula

$$\phi_u^p(x_n) = \delta(x_{n+p}, u), \tag{4}$$

with the indicator function $\delta(\cdot, \cdot)$, $u \in \Omega$ and $p = \{-d_l, \ldots, d_r\}$. Fig. 2 illustrates an example. To predict the secondary structure of the n-th central residue, a windowed residue environment $(x^{n-d_l}, \ldots, x^{n+d_r})$ is selected, from which the position-dependent residue features are extracted. As discussed in [6], a proper window size can lead to good performance, because a too short residue segment (e.g. the green box in Fig. 2) may omit some important classification information while a too long segment (e.g. the red box in Fig. 2) may decrease signal-to-noise ratio. Although a reasonable window size (e.g. the blue box in Fig. 2) seems to be a perfect fit, as pointed out in [15], the β-sheets are formed between two strings of complementary residues that maybe distantly separated in the protein sequence, and a long segment is probably beneficial in β-sheet classification. This

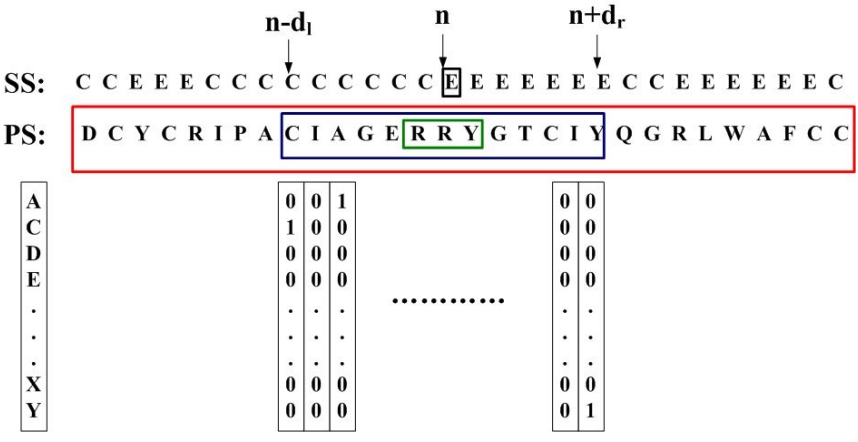

Fig. 2. Schematic diagram of encoding a window of amino acids to predict the secondary structure at the centre position (i.e. the central residue). PS and SS denote the primary sequence and secondary structure labels respectively. A local windowed residue environment $(x^{n-d_l}, \ldots, x^{n+d_r})$ is defined and the presence of each amino acid is encoded using a one out of Ω binary coding scheme as shown. These vectors are concatenated to form the high dimensional input space from which predictions are made via a classification method.

poses a dilemma for current research on predicting the secondary structure of proteins (particularly on β-sheet classification), and more sophisticated machine learning technologies are required. In order to capture long-range dependencies in β-sheet secondary structures and show that they can indeed be exploited, we compare two window size setups in the experiments: one is length-13 (i.e. $d_l = 6$ and $d_r = 6$) that is commonly used [6,9]; the other is length-31 (i.e. $d_l = 15$ and $d_r = 15$), with the intention of exploiting long-range interactions. For the datasets we used, there are 286 features for the length-13 setup; by extending the window size to length-31, the dimensionality of feature space increases to 682, leaving the classifier a feature exploitation challenge.

2.3 Data Sets and Experiment Setup

Two sets of non-homologous protein chains, namely `RS126`[1] and `CB513`[2], are studied in the experiments, where the automatic assignments of secondary structure to experimentally determined 3D structures are performed by DSSP [7].

[1] The set of 126 non-homologous globular protein chains is used in [14] and has been tested by many current secondary structure prediction methods. It contains 23,349 residues with 32% α-helix, 23% β-sheet, and 45% coil. Therefore, when treated as a binary-class classification problem, the data set contains few positive examples.

[2] The set of 513 protein sequences was constructed by [3], which includes almost all the sequences in the `RS126` dataset. It contains 84,119 residues of which 22.7% are β-sheets.

Different from to [6,9,15], we reduced the eight classes of the DSSP assignments to a binary state: E (β-sheet) and B (β-bridge) to E, and all other states to $\sim$E. A seven fold cross validation[3] was then carried out to estimate the predictive accuracy. Following [15], the statistical significance of differences in prediction quality between window size setups was then evaluated by a paired t-test over the cross-validation results. To avoid the selection of extremely biased partitions, the `RS126` (or `CB513`) dataset was randomly divided into seven subsets with each subset having similar size of each type of secondary structures.

As experienced in the literature, the secondary structure prediction task tends to be non-linear and the *radial basis function* (RBF) kernel is commonly used [6,9,15]. Therefore, we also adopt the RBF kernel

$$K(x_k, x_l) = \exp(-\gamma \|\phi(x_k) - \phi(x_l)\|^2) \tag{5}$$

for optimisation (2), where the parameter γ is tuned by cross-validation.

Finally, *the area under the ROC curve* [1] is applied to evaluate the performance of SVM with different window size setups.

3 Results and Discussion

Tables 1 and 2 show the classification performances of linear and RBF classifiers, and the RBF classifier working with the best selected subset of features on the two datasets used. We first note that increasing window length improves performance, implying that some long-range residue patterns are helpful in detecting β-sheets. This is consistent with the postulation discussed in [15].

The classification performance of SVM with RBF kernels displayed in Table 1 and Table 2 is consistently better than SVM with linear kernels on both data sets. Moreover, in this scenario SVM with length-13 performs better than SVM with length-31, which is consistent with the "concern" in [6]. We believe that this is due to interference terms of irrelevant residue patterns brought in by the long window size. Since the feature space of the RBF kernel is of the form [8]

$$\varphi(x) = \exp(-\gamma \|\phi(x)\|^2) \Big(\sqrt{\frac{(2\gamma)^k C_{\boldsymbol{\theta}}^k}{k!}} \phi(x)^{\boldsymbol{\theta}} \Big)_{|\boldsymbol{\theta}|=k, k=0}^{\infty} \tag{6}$$

where $\boldsymbol{\theta} = \big\{ (\theta_i)_{i=1}^D | \theta_i \in \mathbb{N}, |\boldsymbol{\theta}| = \theta_1 + \ldots + \theta_D = k \big\}$, $C_{\boldsymbol{\theta}}^k = \frac{k!}{\theta_1! \ldots \theta_D!}$ and $\phi(x)^{\boldsymbol{\theta}} = \phi_1(x)^{\theta_1} \cdots \phi_D(x)^{\theta_D}$; each feature would have influence on many other features. In this case, irrelevant residue patterns can decrease the signal-to-noise ratio severely and deteriorate performance.

In order to reduce or eliminate the influence of irrelevant features (i.e. residue patterns at specific positions), we applied the *Recursive Feature Elimination* (RFE) [5] technique to select important features (RFE-RBF). Specifically, the length-31 features are first ranked by a linear SVM with RFE[4]. To speed up

[3] The seven fold cross validation setup is inherited from [6,9].

[4] We also tried to rank features by a RBF SVM with RFE, however, this setup biased towards very rare features, which conversely destroyed the performance (the results are not shown in this paper).

Table 1. Prediction performances on the RS126 dataset, as measured by areas under ROC curves, at two different window lengths and with feature elimination from the longer of the windows. Performance of linear and RBF kernels are shown. P-values of T-test for statistical significance in the differences between each method and the RFE-RBF method (results in bold) are shown in the lower part of the table.

Window size	**The area under the ROC curve**	
	LINEAR kernel	RBF kernel
length-13	75.24 ± 1.19	77.30 ± 0.83
length-31	76.22 ± 0.85	76.80 ± 0.75
RFE-RBF	N/A	$\mathbf{77.65} \pm 0.75$

P-value in T-test		
Window size	LINEAR kernel	RBF kernel
length-13	$3.60e-4$	$3.98e-2$
length-31	$1.97e-4$	$3.70e-3$

Table 2. Prediction performances on the CB513 dataset. See caption of Table 1.

Window size	**The area under the ROC curve**	
	LINEAR kernel	RBF kernel
length-13	75.59 ± 0.50	78.28 ± 0.64
length-31	76.96 ± 0.59	78.03 ± 0.73
RFE-RBF	N/A	$\mathbf{78.78} \pm 0.73$

P-value in T-test		
Window size	LINEAR kernel	RBF kernel
length-13	$7.86e-7$	$1.00e-3$
length-31	$1.25e-7$	$4.00e-5$

the process, we eliminate 10 features each time. A proportion of the top ranked features is then selected and the RBF kernel is constructed using these features only. For the experiments on the RS126 dataset, the proportion is taken from $\{\frac{1}{4}, \frac{1}{3}, \frac{1}{2}, \frac{2}{3}\}$ and the performance with respect to the proportion of features is depicted in Fig. 3. We observed that when the proportion increases, the performance first increases because of the increasing contribution of features to the classification. But after a certain point (i.e. $\frac{1}{2}$ in this experiment), the performance decreases, possibly due to the influence of irrelevant features. In addition, if we choose a proper proportion[5], we are able to obtain better performance compared with SVM with length-13 (see results in Table 1 and Table 2).

Fig. 4 depicts the features selected when the proportion achieved the best performance on the RS126 dataset (i.e. using 50% of the features). It is clear that not all the features selected are close to the central residue and certain long distance positions (e.g. $d_r = 7$ and $d_r = 15$ in this experiment) are also important for the classification. Meanwhile, when analysing the residue patterns

[5] Best performance is achieved using about 50% of the features on the RS126 dataset; while this proportion increases to 57% on the CB513 dataset.

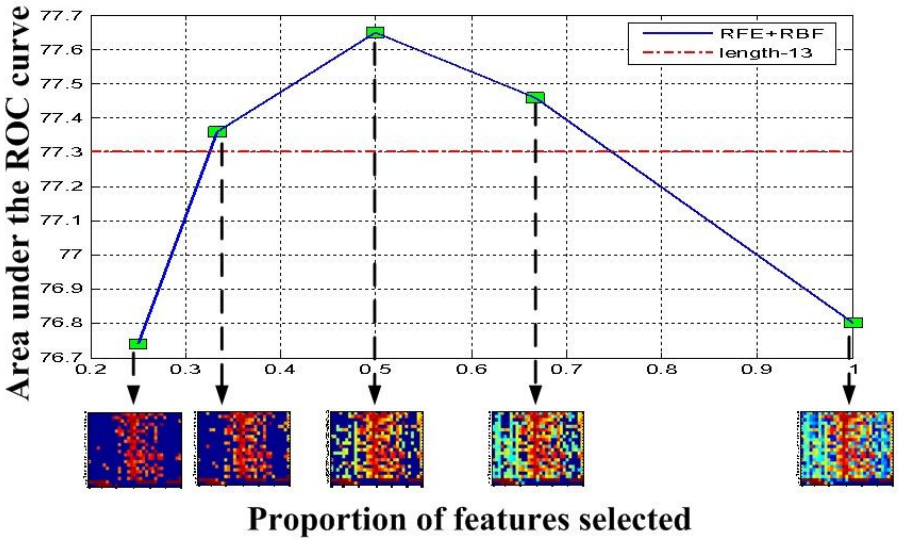

Fig. 3. Feature selection performance at various proportions of features used (`RS126` dataset). From a window size of 31, best performance is achieved using about 50% of the features. While the shorter window considered (length-13) is also about 50%, feature selection selects those amino acid positions, consistent with the distribution observed in Fig. 1.

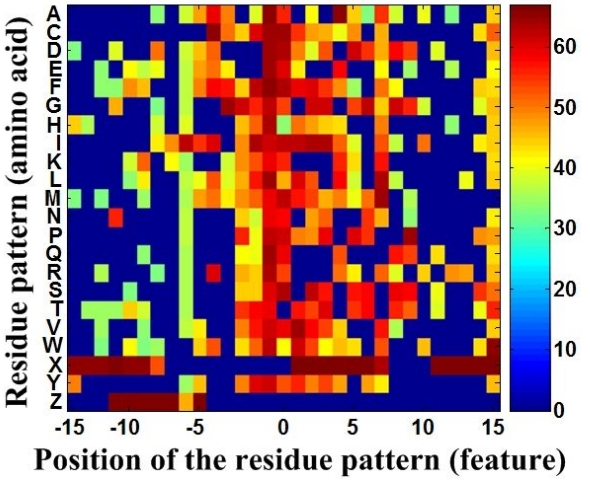

Fig. 4. Selection of relevant residue patterns (`RS126` dataset). The relevance of each amino acid at each position with respect to the centre is shown as an intensity plot. Automatically selected features include amino acids known to have a bias towards β-sheet formation: `D (Asp)`, `F (Phe)`, `G (Gly)`, `I (Ile)`, `L (Leu)`, `M (Met)`, `T (Thr)`, `W (Trp)` and `X`.

(amino acids), some patterns are shown to receive very popular votes. They are: D (Asp), F (Phe), G (Gly), I (Ile), L (Leu), M (Met), T (Thr), W (Trp) and X (unknown amino acids). This observation is consistent with some discussion in [4]:

- The frequency of observation of a hydrophobic amino acid (e.g. Ile, Leu, Met, Trp, Phe) one position before and one position after β-sheets is low. Therefore, when they appear very close to the central residue, the central residue is more likely to be $\sim$E.
- Asp and Gly tend to act as a β-sheet terminator and are therefore very important in formatting β-sheets. In similar fashion, Thr has high propensity for initiating a β-sheet and is also important for β-sheet formation.

In addition, another residue pattern: X (unknown amino acids) is also highly weighted in this experiment, although it was not analysed in [4]. The reason is that X is a rare feature, which appears only 11 times in the `RS126` dataset. Moreover, all examples containing this pattern are in class $\sim$E, which explains why it is selected as an important residue pattern by RFE.

4 Conclusion and Future Work

Our observations show that some long range amino acid interactions can be captured in a feature reduction setting for improved prediction of β-sheet secondary structures. In the feature selection process, the top ranked amino acids are those that are specifically associated with the initiation and termination of β-sheet formations.

In the immediate future, we will verify that the prediction advantage we found for β-sheets is not observed when trying to classify α-helices from coil structures. We also intend formulating the prediction problem as a structured learning problem to exploit long-range dependencies in a principled manner, as for example in the phrase disambiguation task of machine translation [10].

References

1. Bamber, D.: The area above the ordinal dominance graph and the area below the receiver operating characteristic graph. Journal of Mathematical Psychology 12, 387–415 (1975)
2. Cole, C., Barber, J., Barton, G.: The jpred 3 secondary structure prediction server. Nucleic Acids Research, doi:10.1093/nar/gkn238
3. Cuff, J., Barton, G.: Evaluation and improvement of multiple sequence methods for protein secondary structure prediction. Proteins: Struct. Funct. Genet. 34, 508–519 (1999)
4. FarzadFard, F., Gharaei, N., Pezesnk, H., Marashi, S.: β-sheet capping: Signals that initiate and terminate β-sheet formation. Journal of Structure Biology 161(1), 101–110 (2008)
5. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. Machine Learning 46(1-3), 389–422 (2002)

6. Hua, S., Sun, Z.: A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach. J. Mol. Biol. 308(2), 397–407 (2001)
7. Kabsch, W., Sander, C.: Dictionary of protein secondary structure: pattern recognition of hydrogen bonded and geometrical features. Biopolymers 22, 2577–2637 (1983)
8. Minh, H.Q., Niyogi, P., Yao, Y.: Mercer's theorem, feature map, and smoothing. In: COLT, pp. 154–168 (2006)
9. Nguyen, M., Rajapakse, J.: Multi-class support vector machines for protein secondary structure prediction. Genome Informatics 14 (2003)
10. Ni, Y., Saunders, C., Szedmak, S., Niranjan, M.: The application of structure learning in natural language processing. Machine Translation (in Press)
11. Qian, N., Sejnowski, T.: Predicting the secondary structure of globular proteins using neural network models. J. Mol. Biol. 202, 865–884 (1988)
12. Qian, N., Sejnowski, T.: Predicting the secondary structure of globular proteins using neural network models. Journal of Molecular Biology 202(4), 865–884 (1988)
13. Rost, B.: Protein secondary structure prediction continues to rise. Journal of Structural Biology 134, 204–218 (2001)
14. Rost, B., Sander, C.: Prediction of protein secondary structure at better than 70% accuracy. J. Mol. Biol. 232, 584–599 (1993)
15. Ward, J., McGuffin, L., Buxton, B., Jones, D.: Secondary structure prediction with support vector machines. Bioinformatics 19(13), 1650–1655 (2003)