

Protein Classification with Extended Sequence Coding by Sliding Window

Thiago de Souza Rodrigues ^a Antônio de Pádua Braga

^a Santuza Maria Ribeiro Teixeira ^b Sérgio Costa Oliveira ^b

^a Dept. of Electrical Engineering, Federal University of Minas Gerais, MG, Brasil

^b Dept. of Biochemistry and Immunology, Federal University of Minas Gerais, MG, Brasil

1 Introduction

Many methods for finding similarity between protein sequences and to infer classifications to new unseen data are found in the literature [4, 1]. In spite of the availability of these algorithms, there is still a large number of unclassified sequences in the public repositories, what suggests that there is still need for new investigations in the area.

One of the alternatives to improve labelling is to investigate new methods for representing and extracting information from the protein sequences. One of such approaches is the use of *Artificial Neural Networks* (ANN) [7].

The main difficulty on dealing with ANNs to classify protein sequences is that it has a fixed number of inputs and protein sequences are variable in length [9]. Since a representative sample of proteins have different number of amino acids, a direct coding scheme can not be applied directly. However, the *Sequence Coding by Sliding Window* (SCSW) scheme [6] can be used to represent different sizes sequences by providing a fixed length vector representation, regardless of the variability of input sequences sizes.

2 Material and Methods

The coding scheme used in this work, here called *Sequence Coding by Sliding Window* (SCSW), can be described as [6]:

A sequence X , of length N , is defined as a linear succession of n symbols from a finite alphabet, A , of length r . A segment of n symbols, with $n \leq N$, is defines a n -tuple. The vector W_n consists of all possible n -tuples that can be extracted from sequence X . Furthermore, W_n has length r^n .

The elements of this vector are obtained from a sliding window w_n of length n that is run through the sequence X , from position 1 to $N - n + 1$.

Other works used the *SCSW* codification for measure similarity/dissimilarity between sequences,

although with different metrics [10, 5].

In [8] ANN was used to classify sets of proteins according to their function. The percentage of correctly classified patterns varied from 61.99% to 90.40%, depending on the cutoff point and the used window w_n . The best result was obtained to $n = 1$ and $n = 2$ concatenated.

It is easy to see that the *SCSW* scheme does not preserve the sequence order. So the ambiguity problem can arise, where different sequences can result on identical vectors [5].

The ambiguity problem can be sorted out by growing the window w_n length. Furthermore, for a larger enough window, the ambiguity problem does not exist, but with the window size growing, another problem arise. The similarity between subsequences that are smaller than the window size is neglected. Consequently, small similarity regions will be ignored.

Our proposal here is to use more than one window size, similarly to the one used in [8], but with a weighth proportional to the window size used. Furthermore, the outcome vector dimension is r^n , where n is the largest window used. Our new coding, called *Extended Sequence coding by Sliding Window* (E-SCSW), consist of:

Considering the *SCSW* definition, for $k_{min} \leq k \leq k_{max}$, $k_{min} \geq 1$ and $k_{max} \leq n$, the outcome vector $W_{k_{max}}$ has dimension $r^{k_{max}}$, where each position is calculated by:

- a sliding window $w_{k_{max}}$ is run through the sequence X from position 1 to $N - k_{max} - 1$. For each subsequence found, the corresponded element in $W_{k_{max}}$ is incremented with a eight $E_{k_{max}}$;
- for each k_i , $i = k_{max-1}, k_{max-2} \dots k_{min}$ a sliding window w_{k_i} is run through the sequence X from position 1 to $N - k_i - 1$. For each subsequence found, all elements in $W_{k_{max}}$ where the k_i 's first elements was found, is incremented with a eight E_{k_i} , where $E_{k_{max}} > E_{k_{max-1}} > E_{k_{max-2}} > \dots > E_{k_{min}}$.

In this way, we can avoid the ambiguity problem without ignoring the similarity between small subsequences.

3 Results and Discussion

In order to test the *E-SCSW* coding, a set of proteins classified from COG database¹ was selected. Proteins of the 22 functional classes in the COG was used, where one ANN for each class was built.

For a given ANN mapping a class, 150 proteins were selected for each one of the other 21 classes, totaling 3150 proteins for each ANN classifier. For the corresponding class, 3150 additional proteins were selected. Therefore, the total number of proteins was then 6300.

For those classes that did not have enough proteins, like the *nuclear structure*, *cytoskeleton* and others, gaussian resampling was used in order to provide the same number of training vectors for all ANN's.

For each protein, the sliding windows $n = 2$ and $n = 1$ was used with weights $E_2 = 1$ and $E_1 = 0.5$. The values in the outcome vectors were normalized between 0 and 1.

For the validation set, 500 vectors, from each set of 3150 proteins, was randomly chosen totaling 5300 proteins for the training set and 1000 proteins for the validation set.

The training algorithm was the *Bayesian Regularization* [3], for an ANN with a 30 units in the hidden layer and 1 unit in the output layer. Training was carried out in 200 epochs.

In order to compare our method, we used the one proposed in [8], with the same data sets and the same ANN architecture.

After obtaining the set of the output vectors, they were all mapped onto a lower dimensional space using *Principal Component Analysis* (PCA) [2]. The reduced dimension data resulted on lower computational costs for the ANN training.

For all ANN built (one for each class), the predictive accuracies varied from 61% to 79% (figure 1), using the method described in [8] and with our method, the resulted a predictive accuracy ranged from 84% to 95% (figure 2).

The results show that our method is a promising alternative approach as a protein sequences search method. In addition, our classification is not constrained by the database size, because pair-to-pair comparison is not necessary.

¹<http://www.ncbi.nlm.nih.gov/COG>

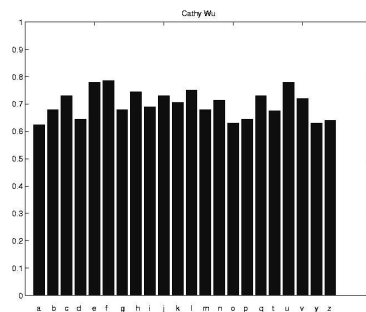


Figure 1: SCSW COG classification result

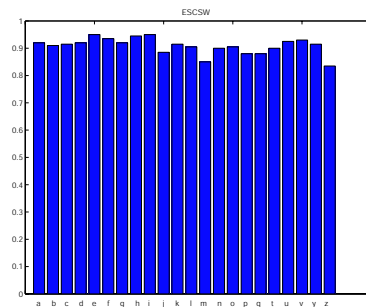


Figure 2: E-SCSW COG classification result

References

- [1] S F Altschul, T L Madden, A A Schaffer, J Zhang, Z Zhang, W Miller, and D J Lipman. *Nucleic Acids Research*, 25(17):3389–3402, 1997.
- [2] S Haykin. *Neural Networks: a comprehensive foundation*. 2 edition, 1999.
- [3] D Mackay. *Neural Computation*, 4(3):415–447, 1992.
- [4] S Needleman and C Wunsch. *Journal of Molecular Biology*, 48(3):443–453, 1970.
- [5] P Petrilli. *CABIOS*, (2):205–209, 1993.
- [6] T S Rodrigues, A P Braga, L G Pacífico, S M R Teixeira, and S C Oliveira. *Genetics and Molecular Biology*, 4(27):673–678, 2004.
- [7] R A Teixeira, A P Braga, R H C Takahashi, and R R Saldanha. *Proceedings VII Brazilian Symposium on Neural Networks*, 2002.
- [8] C Wu, G Whitson, J McLarty, A Ermongkonchai, and T Chang. *Protein Science*, (1):667–677, 1992.
- [9] C H Wu. *Computers Chemistry*, 21(4):237–256, 1997.
- [10] T J Wu, J Burke, and D B Davison. *Biometrics*, 53:1431–1439, 1997.