Data Representation Influences Protein Secondary Structure Prediction using Artificial Neural Networks

Owen Lamont, Hiew Hong Liang and Matthew Bellgard*
Centre of Bioinformatics and Biological Computing,
School of IT, Murdoch University, 6150,
Western Australia
*Correspondence to: Matthew I. Bellgard, E-amil: m.bellgard@murdoch.edu.au

Abstract

Artificial Neural Networks (ANN) have been used very successfully for a number of classification problems in the molecular biology field. Protein secondary structure prediction is one of the oldest and best defined of these classification problems. Yet despite the considerable amount of work conducted in this field there still remain a number of fundamental computational issues that have not been thoroughly investigated, if considered at all. One important issue is identifying an appropriate data representation for input into the ANN. In this paper, we have investigated a range of new encoding schemes and evaluated their performance using recently introduced evaluation criterion. We have done this by preserving the redundant information of DNA codons that is lost when they are translated into amino acids. Interestingly, with our new data representation, the β-strand prediction performance was consistently higher (14% improvement) over the accuracy of the ANNs trained when the conventional representation was used.

Introduction

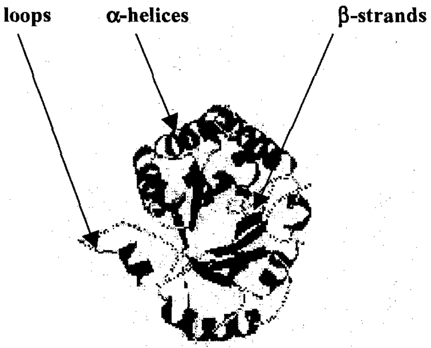
A gene is a sequence of DNA bases that translate into a protein, while a protein is a sequence of amino acids. Once a gene is known it is a relatively simple translation process to determine the sequence of amino acids that will form the protein. There are 4 types of DNA bases named A, C, G and T. Every 3 consecutive bases in a gene, referred to as a codon, translate into one amino acid, although different codons may translate into the same amino acid. A translation table may be seen here[1]. The codon form carries redundant information, as there are $4^3 = 64$ possible codons and just 20 amino acids.

Determining the structure of a protein experimentally has always been an extremely slow and arduous process [1] [2]. This difficulty has provided a strong motivation to seek a computational solution to protein structure evaluation. A protein's amino acid sequence or primary sequence as it is more commonly known folds into a 3 dimensional (tertiary) structure. For several decades attempts have been made at developing automated tools to predict the tertiary structure from the primary sequence but to this date their performance has remained quite low [2]. This is because the process of deriving the tertiary structure from the primary sequence involves immensely complex transformations that result from countless interactions between indeterminate factors[3]. To date the knowledge of such interactions has proved intractable making the creation of a fully automated system unattainable.

Despite this, automated prediction systems have been useful for determining protein secondary structure [4]. As the name suggests secondary structure is an intermediate step between the primary structure and the tertiary structure. There are two forms of secondary structure: an 8 state version and a simpler 3 state version. The 3 state version is the most commonly used in automated secondary structure prediction. In this simpler version that will just be referred to as secondary structures for the rest of this paper, there are 3 categories of sub-structures that each amino acid in the primary sequence can be assigned to. These categories are: α-helix (H), β-strand (B), and loop (C). These represent 3 dimensional shapes but are described in a one-dimensional form. That makes predicting secondary structure from primary structure a 1D to 1D mapping problem [3]. Current prediction systems are limited by how much evolutionary information is incorporated and the difficulty in predicting long-range interactions of β-strands [3] [4].

An example of secondary structures that make up a tertiary structure can be seen below:

[1] Tables http://mel1.angis.org.au/Documents/Tables Last Viewed 22/8/01 Copyright © 1995, AGIC



Representation of the 3D structure of Leishmania mexicana triose phosphate isomerase, accession number *1amk* in the Protein Data Bank [11]

A typical secondary structure format would look something like the following:
Primary Sequence:
AGCGTPSREWQNVTGHLKPCYCVAAHGIKVLHTGLDRAVKNNDNIA
AGGDPSR...
Secondary Structure:
HHHHHHHBBBBBBBHHHHHHHHHHHHCCCCBBBBBBBHHHHHHHHH
HHHBBBB...
That is a secondary structure is assigned to each amino acid in the protein sequence.

The standard data representation for amino acids used for ANNs is to use an orthogonal representation, 20 digit binary number. Thus 10000000000000000000 would represent the amino acid Alanine, and 0100000000000000 would represent Cystine, and so on. The ANNs were then trained to predict the secondary structure of one amino acid at a time in its local context. This involved a 'sliding window' where a region of amino acids of some predetermined length were given as input into the ANN and the secondary structure of the central amino acid used as the target. The target secondary structure would be represented as a 3 digit binary number, 100 for α-helices, 010 β-strands and 001 for loop. The amino acid representation was improved by using frequency profiles [5] from similar proteins when training instead of binary numbers. The binary representation was used in this experiment for simplicity. The best results achieved by others using this representation with an input window width of 11 are around 60% secondary structures predicted correctly [4].

There have been a few notable advances in the methods made over the last decade to improve the performance of ANNs used for secondary structure prediction. One of the most significant advances involved training ANNs with frequency profiles of similar proteins rather than just individual proteins. This takes advantage of evolutionary information where proteins with similar primary sequences will always have similar structures [6]. More recently research has been directed towards polling systems where the results from multiple ANNs or heterogenous prediction systems predictions are combined efficiently to maximise overall accuracy [7] [8]. The most current progress has been in using more sophisticated algorithms to identify related proteins to use in the frequency profiles [5].

In this paper we have experimented with a number of different representations that include evolutionary meaningful information. Our work is based on the premise that important information in the codon form may be lost or obscured when translated into the amino acid format. We tested this theory by preserving this information and training ANNs using various representations of the codon form. We evaluated them using new evaluation methods, and trained and tested the ANNs using a commonly used set of proteins for this purpose [7]. We found significant improvement in β-strand prediction, and the overall performance of ANNs using the new representation is comparable to the system using the conventional orthogonal representations. We discuss these important findings.

Methods

For these experiments we used a multi-layer back-propagation ANN [9]. The topology used consisted of one hidden layer and 3 output units corresponding to each type of secondary structure. The input layer was varied to match the representation and window size used and the number of hidden nodes varied. We chose this representation as it captures the redundancy of the codons that define particular common units. That is, evolutionary information is captured.

Two main codon representations were tested.
The first binary representation was as follows: **A:** 1000, **C:** 0100, **G:** 0010, **T:** 0001,
A or G: 1010, **A or T:** 1001, **C or G:** 0110, **C or T:** 0101, **A or C or T:** 1101, **A or C or G or T:** 1111

The second representation was a more expanded representation of the possible bases used:

A: 1000000000, **C:** 0100000000, **G:** 0010000000, **T:** 0001000000, **A or G:** 0000100000,
A or T: 0000010000, **C or G:** 0000001000, **C or T:** 0000000100, **A or C or T:** 0000000010,
A or C or G or T: 0000000001

Thus an amino acid such as Alanine A would be encoded as: 0010000000 0100000000 0000000001 and Cystine as: 0001000000 0010000000 0000000100. The amino acid representation was the conventional orthogonal representation consisting of a 20 digit binary number with one bit switched on to represent each of the 20 amino acids. For a good review of ANN applications in protein secondary structure see [4].

The CB396 [7] set of proteins developed by James Cuff and Geoffrey Barton was used for the testing and training of the neural network with training sets consisting of a minimum of 20 000 patterns (the remaining patterns were used for testing). Due to the size of the data set and time constraints neither a full nor limited jack knife test was possible. Hence a program was developed to randomly select proteins for the training set which was then manually checked to ensure it had a minimum of 20 000 patterns. The random selection was conducted for each comparison in tables 1 and 2.

The typical measure of accuracy for secondary structure prediction systems is Q3, which measures the percentage correct for each secondary structure individually and overall. A more recent evaluation method referred to as Q8 measures the number of incorrect predictions as well as the ratio correct and determines the Euclidean distance in a four dimensional space between the perfect prediction point and the one achieved to rate performance [10].

Results

The first set of test results is shown in table 1 in which an ANN was trained with an input layer consisting of 9 amino acids and a varying configuration of hidden nodes. In this test the first and simplest codon representation was used. It is interesting to note the performance of the amino acid representation degrades with additional nodes in the hidden layer while the codon representation achieves higher scores in general with more hidden nodes.

Table 1: Results for First Codon Representation Comparison

		$Q_{\alpha\text{-helix}}$	$Q_{\beta\text{-strand}}$	Q_{loop}	Q_3	Q_8
Format	**Hid nodes**					
Amino	2	0.666903	0.302557	0.71094	0.604261	0.611342
Codon	**2**	**0.525348**	**0.265027**	**0.683809**	**0.53336**	**0.540347**
Amino	10	0.633718	0.335938	0.702842	0.596288	0.605839
Codon	**10**	**0.619166**	**0.320502**	**0.678547**	**0.577491**	**0.586001**
Amino	15	0.640521	0.339411	0.690978	0.594595	0.603549
Codon	**15**	**0.629163**	**0.315871**	**0.685769**	**0.583085**	**0.591063**
Amino	20	0.617569	0.356874	0.645433	0.571233	0.582412
Codon	**20**	**0.619698**	**0.339701**	**0.672564**	**0.579463**	**0.588782**

Table 2 shows a comparison between the standard amino acid representation and the expanded codon representation. For this comparison an input window width of 11 was used. Again it is interesting to note the improved codon representation performance with additional hidden nodes while the opposite is true of the amino acid representation. The β-strand prediction accuracy improved to over 0.40, 14% above the amino acid representation.

Table 2: Results for Second Codon Representation Comparison

		$Q_{\alpha\text{-helix}}$	$Q_{\beta\text{-strand}}$	Q_{loop}	Q_3	Q_8
Format	**Hid nodes**					
Amino	2	0.672795	0.342071	0.730052	0.621261	0.62713
Codon	**2**	**0.541506**	**0.312051**	**0.693481**	**0.552682**	**0.559591**
Amino	5	0.657473	0.308527	0.748085	0.612811	0.615737
Codon	**5**	**0.598215**	**0.310078**	**0.700297**	**0.572523**	**0.577272**
Amino	10	0.63601	0.360901	0.746039	0.621426	0.628758
Codon	**10**	**0.621625**	**0.402718**	**0.701265**	**0.606765**	**0.615976**
Amino	20	0.657904	0.309591	0.735352	0.608901	0.611997
Codon	**20**	**0.685484**	**0.32598**	**0.694094**	**0.604786**	**0.608362**

Table 3 and 4 report the results from another test also comparing the expanded codon representation with the amino acid representation. A much larger training set was used for this test consisting of 300 of the 396 amino acids, which is approximately 45 000 training patterns. The observed optimal numbers of hidden nodes from the previous two tests were used in order to better understand the upper limits of each system. With this training set size the overall codon and amino representation results were very similar varying by less then 1% of each other.

Table 3: Comparison of near optimal codon and amino representations

		$Q_{\alpha\text{-helix}}$	$Q_{\beta\text{-strand}}$	Q_{loop}	Q_3	Q_8
Format	**Hid nodes**					
Amino	2	0.656591	0.348354	0.737111	0.624647	0.632672
Codon	**10**	**0.627944**	**0.403004**	**0.708615**	**0.613566**	**0.622718**

Table 4: Comparison of near optimal codon and amino representations

		$Q_{\alpha\text{-helix}}$	$Q_{\beta\text{-strand}}$	Q_{loop}	Q_3	Q_8
Format	**Hid nodes**					
Amino	2	0.631489	0.342585	0.741077	0.613501	0.621817
Amino	5	0.613931	0.380712	0.747221	0.618999	0.628201
Codon	**15**	**0.642748**	**0.373984**	**0.723669**	**0.616825**	**0.624899**

Discussion

From table 2 the codon representation consistently achieved higher β strand prediction accuracies for the all the topologies tried with the exception of the first one. This suggests that the codon representation allows the ANN to recognise non-local interactions more efficiently. In table 3 and 4 the overall accuracy of the expanded codon representation is equivalent to that of the conventional amino acid representation. Table 4 also demonstrates how the extra hidden nodes are used more effectively with the additional training data and how strand prediction raises also. This is expected as strand structures are the least common secondary structure so require more training data to be better recognised, this has been observed when networks are trained with sets of balanced numbers of secondary structures [4] [11]. Recent developments in polling multiple prediction systems could make use of the different strengths of such a codon-based system with its superior strand predictions [12] [7] [8]. These results are extremely encouraging.

Other interesting observations were made during these experiments. The most notable difference being the training times which were on average at least two orders of magnitude greater for the new representations. We suspect the likely reason for this is that the codon representation is more information rich. This can be observed intuitively by noting that amino acids with common properties have codon representations with evolutionary similarities. For instance, hydrophobic amino acids have a "T" at the 2nd codon base position. The improvement in performance with the greater numbers of hidden nodes adds further evidence that the codon representations present the neural network with more sophisticated and subtle relationships that must be learned.

These new representations while incorporating more information also have the potential to increase the 'noise' in the system that may account for some of the lower overall accuracies. Other reasons may be due to uneven use of the input units in the first codon representation where some codons set 7 input bits to 1 and others just 3. In addition some input bits are used by almost half the codons while others are used just once. This may be causing unnecessary ambiguity although it would be possible for new representations to correct this.

While data representation is clearly an important issue and recognised in other applications of intelligent systems these issues have been largely ignored for protein secondary structure prediction. Although considerable progress has been made in secondary structure prediction over the last decade the fundamental data representations and architectures used have changed minimally.

Conclusion

In this research we analysed some of the fundamental approaches to using ANNs for the problem of protein secondary structure prediction. The largest advances in secondary structure prediction with ANNs occurred when relatively minor changes were made to the data representation so there is good reason to investigate this. However the base representation has not been altered since ANNs were first applied to this problem. In this experiment we have trained ANNs with codon representations to take advantage of the information that is lost when they are translated into the amino acid format.

The experiments conducted in this research have shown however that fundamentally different representations can achieve comparable results to the conventional ones currently used and in some areas achieve superior prediction accuracies. Further research will be conducted in this area. Another important area for future investigation is into the appropriate ANN architectures.

References

[1] Tooze, C.B.J., *Introduction to Protein Structure*. first ed. 1991: Garland Publishing, Inc. 302.

[2] Rost, B., *Learning From Evolution To Predict Protein Structure*. 1997.

[3] Rost, B., *Protein structure prediction in 1D, 2D, and 3D*, in *Encyclopedia of Computational Chemistry*, N.L.A. P. von Rague-Schleyer, T. CClark, J. Gasteiger, P. A. Kollman, H. F., Editor. 1998. p. 2242-2255.

[4] Sander, B.R.C., *Third Generation Prediction Of Secondary Structure*, in *Protein structure prediction: methods and protocols*, W.D. M., Editor. 2000, Humana Press. p. 71-95.

[5] Jones, D.T., *Protein Secondary Structure Prediction Based on Position-specific Scoring Matrices*. Journal of Molecular Biology, 1999. **1999**(292): p. 195-202.

[6] Sander, B.R.a.C., *Prediction of Protein Secondary Structure at Better than 70% Accuracy*. Journal of Molecular Biology, 1993. **1993**(232): p. 584-599.

[7] Barton, J.A.C.a.G.J., *Evaluation and Improvement of multiple Sequence Methods for Protein Structure Prediction*. PROTEINS: Structure, Function, and Genetics, 1999. **1999**(34): p. 508-519.

[8] Thomas Nordahl Peterson, C.L., Morten Nielsen, Henrick Bohr, Soren Brunak, Garry P. Gippert, Ole Lund, *Prediction of Protein Secondary Structure .at 80% Accuracy*. PROTEINS: Structure, Function, and Genetics, 2000. **2000**(41): p. 17-20.

[9] Anguita, D., *Matrix Back-propagation*. 1993: Genova.

[10] Zhang, C.-T. and R. Zhang, *A Refined Accuracy Index to Evaluate Algorithms of Protein Secondary Structure Prediction*. PROTEINS: Structure, Function, and Genetics, 2001. **2001**(43): p. 520-522.

[11] Rost, B., *Evolution teaches neural networks to predict protein structure*, in *Scientific Applications of Neural Nets*, T.L. John W Clark, Manfred L Ristig, Editor. 1999.

[12] Karplus, J.-M.C.a.M., *New Methods for Accurate Prediction of Protein Secondary Structure*. PROTEINS: Structure, Function, and Genetics, 1999. **1999**(35): p. 293-306.