DETECTING RESIDUES IN TARGETING PEPTIDES

MIKAEL BODÉN & JOHN HAWKINS
School of Information Technology and Electrical Engineering,
The University of Queensland, QLD 4072, Australia
E-mail mikael@itee.uq.edu.au, jhawkins@itee.uq.edu.au

Knowledge of targeting signals is of immense importance for understanding the cellular processes by which proteins are sorted and transported. This paper presents a system of recurrent neural networks which demonstrate an ability to detect residues belonging to specific targeting peptides with greater accuracy than current feed forward models. The system can subsequently be used for determining sub-cellular localisation of proteins and for understanding the factors underlying translocation. The work can be seen as building upon the currently popular series of predictors SignalP and TargetP, by exploiting the inherent bias for sequential pattern recognition exhibited by recurrent networks.

1. Introduction

Essential to their functional integrity, sub-cellular organelles have specific protein and lipid content. Nascent proteins are directed from ribosomes to the appropriate organelle by means of targeting signals. Knowledge of such signals, usually a short N-terminal *targeting peptide*, is of immense importance for understanding the sorting process that underlies the trafficking of proteins. Many diseases (e.g. hypercholesterolemia and cystic fibrosis) are caused by deficient sorting of proteins. Moreover, knowledge of targeting signals enables sophisticated drug design and high throughput annotation of gene products.

The problem of predicting sub-cellular localisation of proteins has been approached using a range of techniques including weight matrices,[17] expert rules and clustering algorithms,[11] machine learning techniques such as support-vector machines[13] and neural networks,[4,8] using text annotation,[9] structural and evolutionary information,[10] and amino acid composition and order.[5]

In spite of the ever increasing number of documented targeting peptides, there are still major computational hurdles to overcome to provide the tools to understand the principles of sorting signals for specific organelles. Not only are targeting peptides a highly diverse group of biological sequences,[19] but they also exhibit extreme sparseness within a large combinatorial sequence space. In combination, diversity and sparseness present obstacles for automated algorithms that attempt to characterise the discriminative features of data sets. The resulting models lack not only predictive precision but typically also explanatory power. The problem spaces in which the algorithm searches are simply too large for an unbiased classifier to operate in.

A series of neural network based predictors have shown special abilities in handling the task of predicting biological sequence features relating to sub-cellular localisation. SignalP, ChloroP and TargetP[7] all predict sub-cellular targets and cleavage sites of various proteins using simple feed forward networks. The most general of the predictors, TargetP, distinguishes between proteins destined for the mitochondrion, for the chloroplast, for the secretory pathway (the endoplasmic reticulum), and proteins which lack a targeting peptide.

By experimenting with various configurations, Emanuelsson *et al.*[8] found that target specific feed forward networks which slide over a limited window of residues can work as targeting peptide detectors, i.e. distinguish between residues that belong to the targeting peptide (to be cleaved off) and those that belong to the mature protein (or a targeting peptide directing the protein to a different organelle). The detection outputs for the 100 first residues (from each of the target specific networks) are fed into another feed forward network – the target sorting network – which makes a final decision on which sub-cellular compartment the protein is destined for (see Figure 1). SignalP operates in a similar fashion to distinguish between proteins with and without a signal peptide, but simplifies the sorting step by employing a simple threshold criterion on the summed detection outputs.[12] The first step of detecting residues as belonging to a specific targeting peptide is crucial. With highly accurate targeting peptide detectors, the sorting problem reduces to a simple decision.[8]

The scientific search for algorithms for recognising features of biological sequences has so far not stressed the importance of *machine architecture*. This is in part due to the absence of a complete understanding of the relationship between machine architecture and task bias. In spite of the relative success of machine learning techniques, we propose that careful consideration should be given to ensuring that a machine architecture is chosen that will be sensitive to biologically relevant properties of the data. Ultimately, this allows the architecture to be used not only for prediction but also for modelling the cellular processes. To this end we choose to employ recurrent networks because of their inherent preference for solutions that give priority to sequence features closer to the point of interest.[3]

In this paper we replicate the classifier produced by Emanuelsson *et al.*[8] and then show by extensive simulation and careful analysis that recurrent networks are able to recognise residues as belonging to targeting peptides with an accuracy exceeding that of feed forward networks – the type used by the most successful predictors today – in some cases by 24%. We argue that recurrent networks are particularly suited to deal with biological sequences.

2. Finding sequential patterns

Recurrent networks do not simply implement a function of the input sequence to an output (as feed forward architecctures do) but a mapping from a moving input window to an output with regard to an internal state. The internal state is represented by the hidden nodes which receive a delayed feedback from themselves. The state is a result of iteratively processing neighboring inputs (see Figure 2) and can be understood as representing the context in which the current input appears. For biological sequences we are often interested in features at a specific position in the sequence. The features are thought to be a product of interactions between the target monomer and other monomers in the sequence.

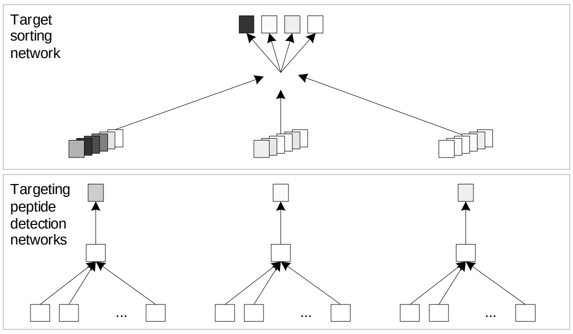

Figure 1. The TargetP neural network architecture. A set of targeting peptide detector networks (one for each target) receive as input a window of residues and output the status of the middle residue. The outputs for each of the peptide detector networks are presented to the target sorting network which outputs the probabilities of the presence of the targeting peptide types (SP, mTP, cTP and the probability of not having a targeting peptide at all).

In general each position in the sequence must be evaluated in the context of the monomers appearing at its flanks, possibly at some distance. Baldi *et al.*[1] proposed the use of bi-directional recurrent networks, an extension of the conventional simple recurrent network which includes inputs from both flanks (N- and C-terminal) by the use of so-called wheels.

Recurrent networks have been noted to exhibit several properties of specific interest. Over multiple iterations, as a state is superimposed onto the next, there is degradation of the impact an input has on the current state. Moreover, training a recurrent network to produce an output which depends on a history of inputs typically results in the creation of structural abstractions in the state space (e.g. a point in state space which is visited when a series of inputs exhibits a particular pattern). With the combinatorial explosion of possible combinations, abstractions are essential for generalising to novel data (cf. recurrent networks trained to process grammars[6,2]). Theoretically, recurrent networks operate on sequences of possibly unspecified length rather than pre-determined windows of input. As the input is presented sequentially by "wheeling in" residues (rather than spatially through a predetermined window of residues), recurrent neural networks can be configured to use less weights. Fewer adaptable parameters means a smaller space to search in for the learning algorithm and reduced risk for over-specialisation.

Computationally, recurrent networks have been shown to be extremely powerful. A number of recent papers have shown that recurrent neural networks exhibit intrinsic properties that naturally lend themselves to general sequential prediction tasks.[16] We have since shown that these theoretical abilities are manifest as a sensitivity to sequential patterns with specific biological characteristsics. In particular, recurrent networks have an inherent bias toward the detection of sequential patterns exceeding that of feed forward architectures.[3] This work contributes to the small, but growing, body of literature that demonstrates the practical utility of recurrent neural networks for bioinformatics problems.[1,15,18]

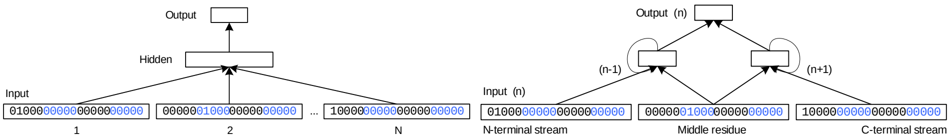

Figure 2. The feed forward architecture (left). Using orthonormal encoding of amino acids, to input a sequence of N residues requires a network with N banks of 20 nodes (one node for each residue). Alternatively, the sequence can be presented by repeatedly moving a smaller window over the sequence – thereby reducing the number of residues that can influence the network output. The recurrent architecture (right). To input a sequence of N residues, a network with one input bank of 20 nodes (one node for each symbol) is needed. In addition, to input all N symbols one can employ two additional banks, one for each flank (N-terminal and C-terminal), iteratively updating an internal state.

3. Method and simulations

To allow objective assessment of recurrent neural networks for sub-cellular localisation prediction we use the standard data set which accompanies TargetP. Each simulation is evaluated by 5-fold cross-validation: The data set is divided into five subsets (of approximately equal size). Four are used for training the system, the remaining subset is used for testing. The procedure is repeated with randomly initialised networks and by shuffling the data subsets so that each of the five subsets appears as a test set exactly once (and each data sample appears as a test case exactly once). Consequently, the five systems are only tested on, for each individual system, unseen sequences. The score we report is the aggregate result for all five test sets (over the five systems). All five-fold cross-validated simulations are then repeated another five times to ensure that final scores are significant.

3.1. *Data sets*

TargetP is able to classify sequences in eukaryotic cells. There are two versions: one for plants, and one for non-plants. The plant version is trained to classify sequences into three specific target classes (mitochondrial, chloroplast, signal peptides) or “other”. The non-plant version is trained to classify sequences into two specific target classes (mitochondrial, signal peptides) or “other”.

The plant data set consists of 940 proteins (368 mitochondrial [mTP], 141 chloroplast [cTP], 269 signal peptides [SP] and 162 nucleus and cytosolic [other]). The non-plant set consists of 2738 proteins (371 mitochondrial, 715 signal peptides and 1652 nucleus and cytosolic).

All sequences are, as convention prescribes, presented to the networks as one-hot bit-strings. The set element is unique for the amino acid, resulting in a 20 bit vector for each residue in the sequence, mutually orthogonal to all others. A single bit is added to accommodate unknown residues.

3.2. *Networks*

The TargetP plant version is equipped with three targeting peptide detection networks: one for mitochondrial, one for chloroplast and one for signal peptides. These networks are equipped with an input window of sizes 35, 55, and 31 amino acid residues respectively. Each detection network is also fitted with a hidden layer consisting of four hidden nodes. All networks were reportedly close to optimal with these configurations.[8] Similarly, the TargetP non-plant version has two detection networks: one for mitochondrial and one for signal peptides, fitted with input windows of sizes 35 and 29 residues respectively, and four hidden nodes. For comparison, we re-produce the simulations reported for the above configuration.

The recurrent networks are similarly used to scan and detect targeting peptides. By iteratively creating a state from the residues next to each position in the sequence, the middle residue is classified as being part of the specific targeting peptide or not (see Figure 3). We tried a few configurations and the results reported below are taken from recurrent networks with wheels of $k = 10$ residues both from the N-terminal and the C-terminal flank. States consist of $h = 4$ nodes of which all are fully recurrent (all nodes feed back to all others within the same state layer). As configurations have yet to be fully explored, we do not claim that the reported configuration is optimal. We use the same configuration ($k = 10$ and $h = 4$) for both plant and non-plant data, and for all sub-cellular targets.

In all cases, we use the logistic output function and the cross entropy error function. All networks are trained using backpropagation and for the recurrent networks the error is "unfolded" through the sequence both upstream and downstream as described by Baldi *et al.*[1] For practical reasons the error flow is truncated after five steps. For both feed forward and recurrent networks, the learning rate (η) is fixed to 0.01, and all weight values are randomly initialised with a Gaussian distribution around 0.0 (variance 0.1). By monitoring errors throughout learning, slow convergence and minor fluctuations were noted. However, the consistency of generalisation results reported below denies the presence of major learning issues.

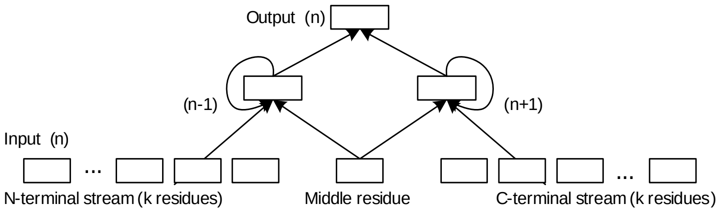
Output (n)
(n-1)
(n+1)
Input (n)
...
...
N-terminal stream (k residues)
Middle residue
C-terminal stream (k residues)

Figure 3. The recurrent peptide detector network operates by traversing the sequence from two directions, accumulating two separate states, until the middle residue is reached, and when the network produces the classification (part of targeting peptide '1' or not '0') at its output. As an example, the symbol G within the sequence ABCDEFGHIJKLM is classified by presenting a window of residues, say 2, from each direction: [AB:-:LM], then [CD:-:JK] and finally [EF:G:HI], where '-' represents a *nil* pattern (all zeros) and ':' indicates node bank boundaries between residues taken from the N-terminal flank, the residue at the point of prediction, and residues taken from the C-terminal flank, respectively.

3.3. *Non-plant proteins*

The data is divided into plant and non-plant proteins. Non-plant proteins are used to train two separate targeting peptide detection networks: one for SP, and one for mTP. A third class (other) of proteins is used as additional negatives for both networks. Training is performed by presenting each network with a sequence randomly drawn from the training subsets (uniformly over the target classes). The sequence is processed by training the network to classify each residue as '1' or '0', in the same manner as TargetP.[8]

After 30,000 training sequences have been presented, the actual output for each position in each test sequence is recorded. Moreover, the squared difference between the target output ('1' or '0') and the actual output is used to assess the classification ability of the network. As the cleavage site determines the end of the string of 1's, the error indicates success of both the classification of the peptide and identification of the cleavage point. In Table 1 the mean errors are shown for both targeting peptide detector networks and for both types of networks. Residues within signal peptides are generally easy to detect for both network types. However, the recurrent network is 24% better than TargetP's SP detection network. Mitochondrial targeting peptides also demonstrate an advantage for recurrent networks (15%).

Table 1. Errors for non-plant proteins.

Target \ Network type	TargetP replica		Recurrent network	
SP	0.0177	(0.0008)	0.0143	(0.0005)
mTP	0.0273	(0.0016)	0.0238	(0.0031)

Note: The mean summed squared error over all test patterns for the two detector networks (SP and mTP), and over six repeats of each five-fold cross-validated configuration. Standard deviations between repeats are given in parenthesis. The mean increase in accuracy provided by RNs is 24% for SP and 15% for mTP.

We collected the outputs for all test sequences that are known to have a signal peptide and presented them to the SP detection networks. The position-specific errors are shown in Figure 4. The cleavage site of signal peptides is usually located at position 15-30 of the nascent protein relative to the N-terminal end (Mean=23, SD=6, in the data set). The classification error is generally higher around the cleavage site. However, the error is considerably higher for the feed forward network employed by TargetP for most residues preceeding the cleavage site. Moreover, there is a sharp downturn in performance after position 13. Position 14 is the first position which is classified using a window fully populated with real amino acids (when the window ranges over non-existing sequence positions a nil-pattern pads out the window). It is thus quite likely that the TargetP detection network uses such weak, encoding-specific indicators. The recurrent network – avoiding the pre-fixed window approach – shows no dramatic changes in performance. After the cleavage site, both network types are performing equally well.

The errors for mitochondrial test sequences were similarly analysed. The performance of recurrent targeting peptide detection networks is considerably better before and around the cleavage sites of the nascent protein sequence. The cleavage sites of matrix mitochondrial processing peptidases occur further along the nascent protein (Mean=34, SD=16, in our data). Being very close to their mean, the error profiles of individual networks show little variation.

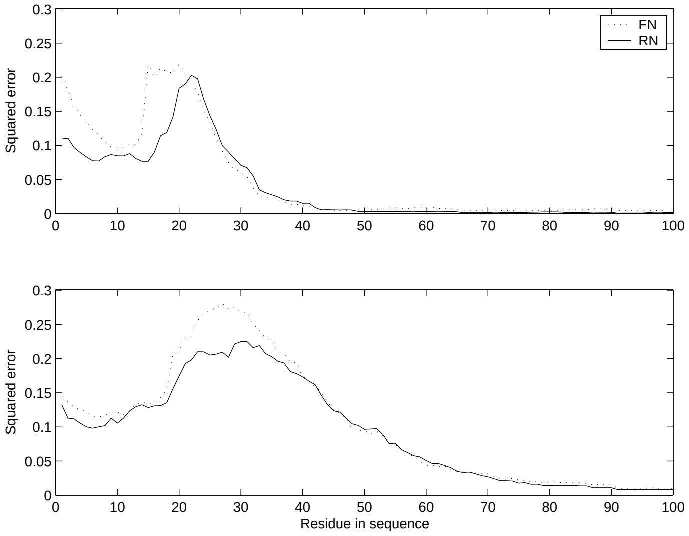

Figure 4. Mean errors in the two types of targeting peptide detection networks (FN/TargetP dotted line, RN solid line) for each sequence position (1-100). Above: The errors in the SP detecting network, for sequences known to contain a signal peptide. Below: The errors in the mTP detecting network, for sequences known to contain a mitochondrial targeting peptide. All errors are means over six repeats of the 2738 sequence five-fold cross-validated non-plant test data.

3.4. *Plant proteins*

For plant proteins in our data set there are three targets and recurrent networks improve on the feed forward networks employed by TargetP for SP and mTP sequences. cTP sequences are handled better by the original feed forward detector networks. However, this advantage is only present in the latter end of the sequence (after position 55). See Table 2 for details and Figure 5 for the position-specific error profiles.

The errors are generally higher for the mTP detector network when trained on plant proteins. This may seem odd at first – considering that the same set of proteins is re-used for the plant-version of TargetP.[8] However, since other plant-specific proteins (including cTP sequences) are used as negatives, the discriminative task of the plant-specific mTP detector is fundamentally different.

The generalisations offered by the two types of network are clearly distinct. Factors determining location are complex in nature, as exemplified by the existence of dual targeting.

Table 2. Errors for plant proteins.

Target \ Network type	TargetP replica		Recurrent network	
SP	0.0174	(0.0011)	0.0142	(0.0004)
mTP	0.0608	(0.0013)	0.0546	(0.0021)
cTP	0.0562	(0.0041)	0.0686	(0.0131)

Note: The mean summed squared error over all test patterns for the three detector networks (SP, mTP and cTP), and over six repeats of each five-fold cross-validated configuration. Standard deviations between repeats are given in parenthesis. The mean increase in accuracy provided by RNs is 23% for SP, and 11% for mTP. For cTP, recurrent networks reduces accuracy with 18%.

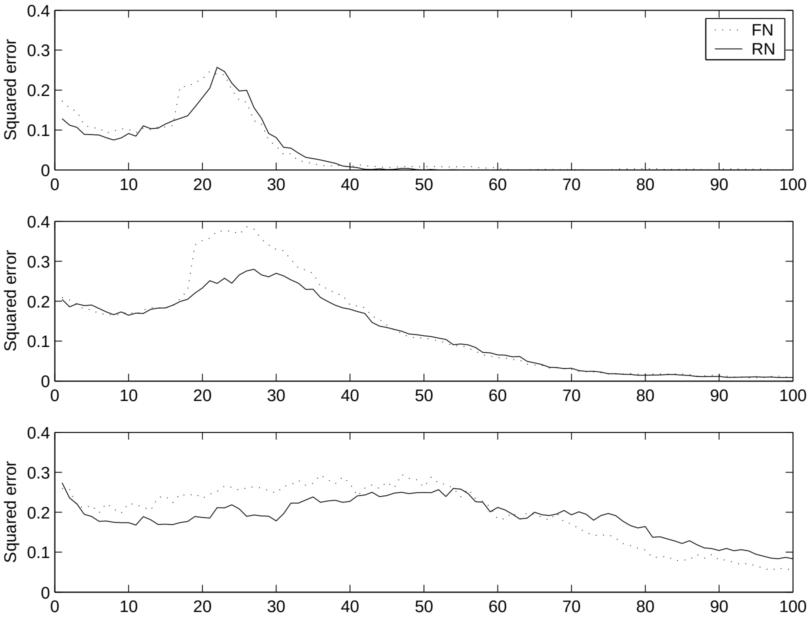

Figure 5. Mean errors in the three types of targeting peptide detection networks (FN/TargetP dotted line, RN solid line) for each sequence position (1-100). Above: The errors in the SP detecting network, for sequences known to contain a signal peptide. Middle: The errors in the mTP detecting network, for sequences known to contain a mitochondrial targeting peptide. Below: The errors in the cTP detector networks for known cTP sequences. All errors are means over six repeats of the 940 sequence five-fold cross-validated plant test data.

A growing number of proteins have been observed to be translocated to both mitochondria and chloroplasts.[14] From the list in Peeter and Small's study, two were annotated with a potential cleavage site (P27456, Glutathione reductase, and P29463, CoxVa/Triose phosphate translocator), and the outputs of two individual networks of each type are shown in Figure 6.

P27456, more frequently found in the chloroplast (only 3% in mitochondria), has a putative cleavage site at position 60, which is well-recognised by the recurrent cTP detector.

P29463 has a putative cleavage site at 78, recognised by the feed forward cTP detector. However, TargetP classifies the precursor as a mTP (0.036 for cTP and 0.769 for mTP),[14] possibly due to the strong detection signal in the beginning of the sequence. The simple sorting mechanism employed by TargetP is thus weak. We note that the two types of network act very differently and it may be beneficial to design sub-cellular localisation methods that employ both types, and a sorter that copes with complex signals.

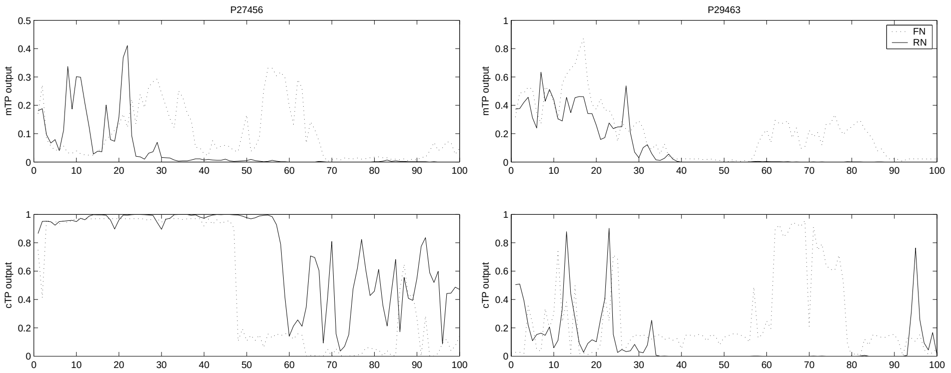

Figure 6. Network outputs for two dual-targeted proteins. The upper graphs show the output of the mTP detector networks, and the lower graphs show the output of the cTP detector networks. The two sequences, P27456 and P29463, have potential cleavage sites at 60 and 78, respectively.

4. Conclusion

We note that recurrent networks are notably better than the feed forward networks used by TargetP at classifying residues as belonging to a targeting peptide. The advantage is particularly clear within the window believed to exhibit the strongest signals used by the translocation machinery.[7] The reason for the success lies partly in the fact that recurrent networks are naturally biased towards detecting sequential patterns.[3,16] The co-occurrence with improved detection accuracy and putative signal sites supports that recurrent networks base their generalisation on real functional regions.

We conclude that in cases where improvements in accuracy are crucial, recurrent neural networks seem well worth exploring. In a recent review Emanuelsson[7] illustrates the superiority of TargetP compared to a representative set of alternative localisation predictors. By improving on TargetP both for plant data and non-plant data, our approach represents the most promising to date. In particular we observe that on all sequence types, recurrent detector networks excel in the first half, and may therefore by the preferred method to be used at the N-terminal end. Informed by the presented analysis, the more accurate targeting peptide detection mechanism is integrated into a full-blown prediction service (the *Protein Prowler* is available online at `http://www.itee.uq.edu.au/˜pprowler`).

References

1. P. Baldi, S. Brunak, P. Frasconi, G. Soda, and G. Pollastri. Exploiting the past and the future in protein secondary structure prediction. *Bioinformatics*, 15:937–946, 1999.
2. M. Bodén. Generalization by symbolic abstraction in cascaded recurrent networks, neurocomputing, 57, pp. 87-104. *Neurocomputing*, 57:87–104, 2004.
3. M. Bodén and J. Hawkins. Improved access to sequential motifs: A note on the architectural bias of recurrent networks. 2004. Submitted.
4. Y.-D. Cai, X.-J. Liu, and K.-C. Chou. Artificial neural network model for predicting protein subcellular location. *Computers & Chemistry*, 26(2):179–182, 2002.
5. K. C. Chou and Y. D. Cai. Prediction and classification of protein subcellular location-sequence-order effect and pseudo amino acid composition. *Journal of Cellular Biochemistry*, 90(6):1250–1260, 2003.
6. J. L. Elman. Learning and development in neural networks: The importance of starting small. *Cognition*, 48:71–99, 1993.
7. O. Emanuelsson. Predicting protein subcellular localisation from amino acid sequence information. *Briefings in Bioinformatics*, 3(4):361–376, 2002.
8. O. Emanuelsson, H. Nielsen, S. Brunak, and G. von Heijne. Predicting subcellular localization of proteins based on their n-terminal amino acid sequence,. *Journal of Molecular Biology*, 300(4):1005–1016, 2000.
9. Z. Lu, D. Szafron, R. Greiner, P. Lu, D. Wishart, B. Poulin, J. Anvik, C. Macdonell, and R. Eisner. Predicting subcellular localization of proteins using machine-learned classifiers. *Bioinformatics*, 20(4):547–556, 2004.
10. R. Nair and B. Rost. Better prediction of sub-cellular localization by combining evolutionary and structural information. *Proteins*, 53(4):917–930, 2003.
11. K. Nakai and P. Horton. PSORT: A program for detecting sorting signals in proteins and predicting their subcellular localization. *Trends in Biochemical Sciences*, 24(1):34–35, 1999.
12. H. Nielsen, J. Engelbrecht, S. Brunak, and G. von Heijne. Identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. *Protein Engineering*, 10:1–6, 1997.
13. K.-J. Park and M. Kanehisa. Prediction of protein subcellular locations by support vector machines using compositions of amino acids and amino acid pairs. *Bioinformatics*, 19(13):1656–1663, 2003.
14. N. Peeters and I. Small. Dual targeting to mitochondria and chloroplasts. *Biochimica et Biophysica Acta (BBA) - Molecular Cell Research*, 1541(1-2):54–63, 2001.
15. G. Pollastri, D. Przybylski, B. Rost, and P. Baldi. Improving the prediction of protein secondary strucure in three and eight classes using recurrent neural networks and profiles. *Proteins*, 47:228–235, 2002.
16. P. Tino, M. Cernansky, and L. Benuskova. Markovian architectural bias of recurrent neural networks. *IEEE Transactions on Neural Networks*, 15(1):6–15, 2004.
17. G. von Heijne. A new method for predicting signal sequence cleavage sites. *Nucleic Acids Research*, 14:4683–4690, 1986.
18. A. Vullo and P. Frasconi. Disulfide connectivity prediction using recursive neural networks and evolutionary information. *Bioinformatics*, 20(5):653–659, 2004.
19. E. J. B. Williams, C. Pal, and L. D. Hurst. The molecular evolution of signal peptides. *Gene*, 253(2):313–322, 2000.