Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Original Paper

OXFORD

Laboratory of Bioinformatics 2

Signal peptide prediction: a comparative study of different available methods

Viola Meixian Vuong [1]*

[1] Department of Pharmacy and Biotechnology, University of Bologna, Bologna, 40126, Italy

*To whom correspondence should be addressed.

Abstract

Motivation: Identifying signal peptides in newly synthesized protein sequences can deepen our knowledge in protein localization and function characterization. Here, we address the issue by comparing two already known signal peptide prediction approaches, namely the von Heijne method and the Support Vector Machine implementation.
Results: Comparison of both methodologies performance in a independent benchmark dataset results in equivalent accuracies in predicting signal peptides cleavage site. Furthermore, both approaches possess average prediction capabilities with respect to other more powerful available signal peptide detection tools.
Contact: violameixian.vuong@studio.unibo.it
Supplementary information: Supplementary materials are provided as a separate attachment.

1 Introduction

Protein sorting is a complex biological mechanism by which newly synthesized proteins are transported to their final intra- or extra-cellular destination. Understanding how this mechanism works will grant us knowledge about the protein function, as well as possible insights into potential protein interactors and disease-related complications due to incorrect sorting. The information regarding this controlled delivery process is often encoded directly into the sequence of the protein. In particular, proteins destined to the cellular secretory pathway are endowed with a N-terminal signal peptide carrying this information (Blobel and Dobberstein, 1975).

Signal peptides (SPs) are short transient signal sequences, which are cleaved once the protein reaches its destination. Possible destinations for secretory proteins are the cellular membrane, the endoplasmic reticulum (ER), the Golgi apparatus and the extracellular environment. These signals are structurally organized into three distinct regions: a N-terminal region (n-region), a central hydrophobic region (h-region), and a more polar C-terminal region (c-region) (von Heijne, 1990). Related studies show that, on average, residues -13 to -6 and residues -5 to -1 correspond respectively to the h- and the c-region, with residues +1 and +2 selected as alternative cleavage sites with respect to the canonical one (von Heijne, 1983). In particular, the h-region, also known as the hydrophobic core of SPs, is enriched with the hydrophobic residues Phe, Ile, Leu, Ala, and Val. Their frequency dramatically drops at the h/c boundary, from which the hydrophilic residues Gly, Pro, Ser, Thr (among all others) start to dominate. The n-region, on the other hand, is variable both in terms of length and composition (von Heijne, 1985). Due to functional features correlated to the identification of the cleavage site (von Heijne, 1984) and the recognition by a ubiquitous machinery for ER translocation (Walter *et al.*, 1981), the h- and c-regions are mostly conserved and shared across different domains and kingdoms.

SPs prediction is already a well-addressed biological problem. In particular, this issue concerns two sub-problems: discriminating between SP and non-SP proteins, and predicting the position of the cleavage site. In order to address the matter, several methods have already been developed and improved since 1983. From simpler strategies involving weight matrices able to localize the cleavage site to machine learning-based predictive methodologies, neural networks are by far the more successful tools ever developed in the field (reviewed by Nielsen *et al.*, 2019). Starting from 2018 (Savojardo *et al.*, 2018), the introduction of deep learning has boosted previous approaches performance. Yet, new improvements can still be introduced in order to solve currently open challenges in the SPs prediction field, for example the discrimination between transmembrane and SPs sequences due to their matching hydrophobic regions and the discrimination between similar N-terminal sorting signals.

In this work we present two of the already known SPs detection procedures, the von Heijne method and the support vector machine (SVM) implementation. The former, a weight matrix-based approach, aims to predict the position of the cleavage site from a set of aligned SPs. The latter, a machine learning-rooted algorithm, designs a model able to predict the presence/absence of SPs in unseen proteins by assuming their length and composition. We strive to statistically evaluate and compare the performance of both methodologies during the training and benchmark phases. Concluding results show that both approaches have

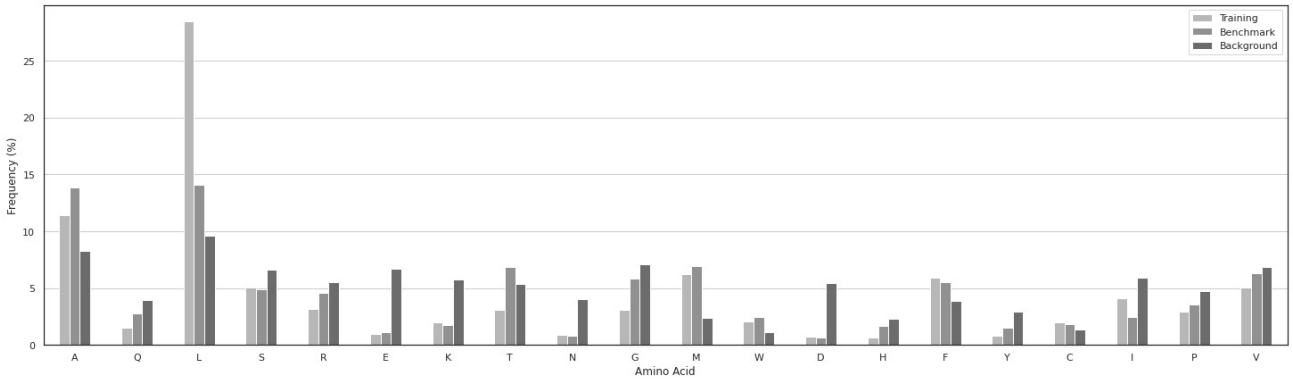

Fig. 1: Amino acid composition of signal peptides: comparison between training, benchmark and SwissProt background model. Signal peptides are hydrophobic-residues rich. Notice the clearly dominance of residues Leu and Ala with respect to others, both in the training and benchmark datasets. Polar residues are generally poorly represented.

similar cleavage site prediction ability, which is indeed outperformed by more sophisticated available tools.

2 Methods

2.1 Data

Protein sequences used to perform the overall analysis were extracted from a previous experiment (Almagro Armenteros *et al.*, 2019). Only sequences which were *i)* reviewed (UniProt Knowledgebase SwissProt, release 2018_04) (Consortium *et al.*, 2018); *ii)* at least 30-residue long and *iii)* experimentally proven to have a SP for the cleavage site (ECO: 0000269) were considered.

Both training and benchmark datasets were provided. The training dataset consists in 1723 eukaryotic sequences, separated into 250 positive examples and 1465 negative ones; the benchmark dataset contains 7456 sequences, 209 of which are positive examples. We define as positive example those sequences endowed with a N-terminal secretory SP, and as negative examples those eukaryotic proteins with either a cytosolic, nuclear, mitochondrial, plastid, and/or peroxisomal subcellular localization and not belonging to the secretory pathway with experimental evidence. All protein sequences were shorten to the first 50 N-terminal residues. No-redundant data is present.

2.1.1 Data analysis

In order to secure an accurate and deep understanding of the whole data, statistical investigations were conducted.

Comparison of the SP length distribution of both datasets (which singly reflects the high-quality data collection procedure adopted) shows that they behave similarly, displaying at the same time SPs tendency of being typically 20-25 residues long (Supplementary materials Fig. 1). High frequency of shorter peptides, which can be observed in the training dataset, indicates possible sampling errors.

In accordance to related studies (von Heijne, 1985), the amino acid composition-wise analysis exhibits hydrophobic residues propensity of being highly represented (Fig. 1). This is confirmed by the consensus SP sequence logo, built in WebLogo 2.8.2 (Crooks *et al.*, 2004). The logo (Supplementary materials Fig. 3) captures the peculiar features of SP sequences starting 13 positions downstream and 2 positions upstream the cleavage site, i.e. the long Leu-enriched hydrophobic core of eukaryotic SPs (position -13 to -6), and the small [AV]XAA motif directly preceding the cleavage site (position -3 to +1). Unfortunately, the identification of this small motif is not significantly enough to independently localize the cleavage site. Due to its size, in fact, the same motif can be identified in other regions of the protein.

Furthermore, taxonomic classifications performed on both datasets reveal half of the sequences belonging to the Metazoa kingdom, leaving the rest to other kingdoms such as Plants and Fungi. On top of that, we observe that 75% of the overall sequences are clustered in the top 5 most represented taxas, i.e. *Homo sapiens*, *Arabidopsis thaliana*, *Saccharomyces cerevisae*, *Saccharomyces pombe*, and *Mus musculus* (Supplementary materials Fig. 2).

2.1.2 Cross validation

To avoid overfitting (i.e. inability of generalization), sequences belonging to the training dataset were further split, randomly, into equally-sized subsets in order to perform a 5-fold cross validation procedure. Four subsets (set 0, set 1, set 2, set 3) contain 345 sequences each; set 4, on the other hand, has only 343 sequences.

Each model will be therefore built on $k-1$ training subsets, with k: number of folds, and then tested against the k^{th} remaining subset. This procedure will be repeated until each k-fold subset has served as testing set.

2.2 von Heijne

The von Heijne method was implemented in Python 3.8.2 with the support of the Python libraries Numpy 1.23 (Harris *et al.*, 2020), Pandas 1.5.1 (pandas development team, 2020), and scikit-learn 1.1.3 (Pedregosa *et al.*, 2011).

2.2.1 Training-testing

The following procedure was performed for each k-fold retrieved from the training dataset.

Training subset's sequences proven to have a SP were fragmented respectively 13 positions upstream and 2 positions downstream their cleavage site. These fragments were collected and aligned in order to build a $20 \times n$ Position-Specific Probability Matrix (PSPM), with n: length of the alignment. The matrix collects in each of its j position the frequency of each residue r in all aligned fragments. In order to avoid errors in further implementations of the method, a regularization procedure via pseudocounts was introduced (i.e. each PSPM cell was initialized to 1)

(Henikoff and Henikoff, 1996):

$$M_{r,j} = \frac{1}{N+20}(1 + \sum_{i=1}^{N} I(s_{i,j} = r)) \quad (1)$$

where $M_{r,j}$ denotes the frequency of the residue r in position j, N denotes the number of aligned fragmented sequences, $s_{i,j}$ denotes the observed residue in position j of the sequence i, and $I(s_{i,j} = r)$ denotes the indicator function (equal to 1 if the condition is met, 0 otherwise).

Each residue frequency was then divided by its corresponding UniProtKB SwissProt frequency (release 2022_04) (uni, 2021) from which the log_2-transformation was computed:

$$W_{r,j} = log_2 \frac{M_{r,j}}{b_r} \quad (2)$$

where $W_{r,j}$ denotes the log odd score of the residue r in position j, $M_{r,j}$ denotes the frequency of the residue r in position j, and b_r denotes the frequency of the residue r in the SwissProt background model.

The resulting Position-Specific Weight Matrix (PSWM) was then trained to predict the position of the cleavage site. This was achieved through a "sliding window" procedure that fractures each sequence into 15-residues long fragments, assigning to each of them a weighted score:

$$score_{(S|W)} = \sum_{i=1}^{L} W s_i, i \quad (3)$$

where S denotes the sequence, W the PSWM, L the length of the sequence S, and s_i the residue of the sequence S in position i. The cleavage site was appointed to the highest scoring fragment among the 35 possible ones (being 50-residues long, each sequence is cut 35 times as the window is shifted one position towards its C-terminal per time).

All maximal PSWM-based scores computed for all training sequences were then used to select the optimal threshold which separates SP and non-SP proteins apart. A precision-recall curve was therefore built to test varying threshold values:

$$precision = \frac{TP}{TP + FP} \quad (4)$$

$$recall = \frac{TP}{TP + FN} \quad (5)$$

where TP denotes the number of true positives (actual positives that are correctly predicted as positives), FP denotes the number of false positives (actual negatives that are incorrectly predicted as positives), and FN denotes the number of false negatives (actual positives that are incorrectly predicted as negatives).

Precision quantifies the number of positive class predictions that actually belong to the positive class; recall quantifies the number of positive class predictions made out of all positive examples in the dataset. Both scoring indexes range between [0, 1], as well as their harmonic mean, the so-called F1 score. Our optimal threshold coincides with the highest F1 score among all possible ones:

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (6)$$

The selected threshold and the corresponding weight matrix were then tested on the testing-fold subset, following the same "sliding window" procedure adopted in the training phase.

2.2.2 Benchmark

A benchmark step to further evaluate the von Heijne model performance was conducted.

Consistently to sum (1) and the logarithmic transformation (2), a new weight matrix was built from SP sequences belonging to the entire training dataset (independently on the cross-validation classes). Following the same procedure adopted during the training-testing phase, a PSWM-based score (3) was assigned to each benchmark sequence. Only sequences satisfying the optimal threshold condition (i.e. with a higher weighted score) were predicted as having a SP. This optimal threshold corresponds to the average value of the five thresholds computed during the training-testing phase.

2.3 Support vector machine

Support Vector Machine (SVM) is a supervised machine learning method able to solve both classification and regression problems (Cortes and Vapnik, 1995).

Here, we exploit SVM to classify unseen protein sequences as being secretory proteins (i.e. belonging to the positive class) or not (i.e. belonging to the negative class). We train the model with a set of known experimentally-proven SP sequences, allowing the extraction of those distinct features that make this classification possible.

In order to optimally solve this non-linearly separable problem, we build a classifier using as kernel the radial basis function (RBF), which maps the data into a new feature space where this linear separation is possible:

$$K(x, y) = exp^{-\gamma||x-y||^2} \quad (7)$$

$$\gamma = \frac{1}{n * 2\sigma^2} \quad (8)$$

where $K(x, y)$ represents the kernel function of the two vector points x and y, $||x - y||^2$ denotes the squared Euclidean distance between the two vector points x and y, γ is the positive scalar that defines how far the influence of a single training example reaches, n is the number of features, and σ^2 is the variance.

The implementation of the SVM classifier was carried out in Python 3.8.2 operating the same Python libraries of the previous von Heijne method. The model was trained considering the hyperparameters k, C, γ, with k: expected SP length; C: cut-off value that compromises between correct classification of training examples and maximization of the decision function's margin.

2.3.1 Training-testing

A minimal grid search procedure aiming to find the best hyperparameters combination between the three selected hyperparameters k, C, and γ was carried out. Each hyperparameter has three potentially ideal values, respectively $(20, 22, 24)$, $(1, 2, 4)$, $(0.5, 1, 'scale')$.

These values were combined into all possible 27 different k-C-γ combinations, i.e. 27 different classifiers were built, trained and fitted against a $20 \times k$ matrix, with k: expected SP length, containing all residues frequency in each k_i position of the training sequences alignment. Following the fitting procedure, each classifier was tested against the remaining testing fold (satisfying therefore the 5-fold cross validation procedure adopted), from which the corresponding Matthews Correlation Coefficient (MCC) was computed:

$$MCC = \frac{TP \times TN - FN \times FP}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (9)$$

where TN denotes the number of true negatives (actual negatives that are correctly predicted as negatives), FP denotes the number of false positives, FN denotes the number of false negatives, and TP denotes the number of true positives.

Table 1. Confusion matrix for binary classification.

	Predicted Negative	Predicted Positive
Actual Negative	TN	FP
Actual Positive	FN	TP

Note: TN: true negatives; FP: false positives; FN: false negatives; TP: true positives.

As this scoring index represents the proportion between the predicted and actual values in the range [-1, +1] without being affect by unbalanced datasets (Matthews, 1975), we exploit it to report each classifier real performance, computed as the average between the 5 MCC scores returned after each testing procedure.

2.3.2 Benchmark

Predictions on the benchmark dataset were made by modeling an optimal classifier, built on the set of hyperparameters corresponding to the maximum MCC value retrieved during the training-testing phase. The model was then fitted against the entire training dataset, i.e. independently on the cross-validation fold.

2.4 Scoring indexes

A qualitatively evaluation of the models performance is provided via a $n \times n$ confusion matrix, with n: number of target classes (Table 1). Quantitatively, accuracy (ACC), along side the already proposed precision (4), recall (5), F1 (6) and MCC (9), is computed:

$$ACC = \frac{TP + TN}{TP + FP + TN + FN} \tag{10}$$

where TP is the number of true positives, FP is the number of false positives, FN is the number of false negatives, and TP is the number of true positives.

ACC represents the ratio, in the range [0, 1], between the correctly predicted instances and all the instances in the dataset. However, it fails in providing a fair estimate of the classifier performance in case of unbalanced classes.

2.5 Evaluation

In order to deepen our knowledge on both implementations performance, benchmark's FP and FN were further investigated.

An incorrect classification as FP arises mainly for two reasons: i) presence of an α-helix/β-sheet in the first 50 N-terminal residues (i.e. transmembrane proteins); ii) presence of a targeting N-signal towards mitochondria (mTPs), chloroplasts (cTPs) and/or peroxisomes (PTS2) organelles (i.e. transit peptides). This information was retrieved from UniProtKB, exploiting UniProt's tool ID mapping (Jain *et al.*, 2009).

The overall set of FP was evaluated computing the corresponding false positive rate (FPR):

$$FPR = \frac{FP}{FP + TN} \tag{11}$$

where FP denotes the number of false positives, while TN denotes the number of false negatives. The same ratio was computed for each transit peptide type, as well for the overall set of transit peptides.

On the other hand, transmembrane (TM) proteins were analysed computing the ratio between FP which are also TM proteins, and the overall number of TMs:

$$\frac{FP^{TM}}{TM} \tag{12}$$

where FP^{TM} represents those FP which are also TM proteins, while TM denotes the total number of TM proteins. Both evaluations were determined only for manually curated proteins, annotated with the following ECO codes: ECO:0000269, ECO:0000303, ECO:0000305, ECO:0000250, ECO:0000255, ECO:0000312, ECO:0007744.

Misclassifications that lead to FN, on the other hand, derived from assumptions made during the modeling procedure. As different assumptions were speculated, different investigations were conducted.

We explain von Heijne's FN by representing graphically the consensus SP sequence logo (from position -13 to position +2 of the cleavage site), while SVM-derived ones were assessed by observing possible differences in the distribution of SP length and composition with respect to the real secretory proteins ones.

FN of both methodologies were also investigated in order to identify possible inclinations towards one kingdom or another.

3 Results

3.1 Cross validation

The 5-fold cross validation procedure applied during the von Heijne training-testing phase returned five threshold values, one for each testing fold analyzed. These threshold values are, respectively, 7.93 (testing set 0), 7.87 (testing set 1), 8.61 (testing set 2), 8.04 (testing set 3), 8.58 (testing set 4). Their average value 8.21 corresponds to the optimal threshold value selected to predict presence/absence of the SP in benchmark sequences. The same protocol performed in the SVM implementation returned as maximum MCC score 0.84, which corresponds to the optimal k-C-γ combination $(20, 2, 'scale')$.

A statistical description of both methodologies performance is reported in Table 2. As both ACC and F1 score (and therefore, precision and recall) can generate misleading performance assessment results due to imbalanced datasets, we rely on the MCC score to make our conclusions (Chicco and Jurman, 2020). Upon comparison, we can derive that the SVM implementation slightly outperforms the weight matrix approach in predicting secretory proteins.

3.2 Benchmark

Benchmark results (Table 3) of the weight matrix approach shows that our implementation ability in correctly predict SP cleavage site slightly falls behind the original work performance (von Heijne, 1986).

Nevertheless, the same conclusion made in the cross-validation performance assessment is recreated in the benchmark evaluation. However, in agreement with previous studies, due to the minimal difference in terms of MCC scores, SVM prediction ability can be reduced to a typical von Heijne approach (Wang *et al.*, 2005). The comparison of all other scoring indexes confirms this perspective, further validated by the

Table 2. Cross validation scoring metrics.

	von Heijne	SVM
ACC	0.95±0.00	0.94±0.01
F1	0.81±0.01	0.80±0.03
MCC	0.78±0.01	0.84±0.02
Precision	0.84±0.03	0.85±0.02
Recall	0.79±0.01	0.77±0.04

Note: ACC: accuracy; MCC: Matthews Correlation Coefficient; SVM: support vector machine. For each averaged value the corresponding standard error (SE) was determined as $SE = \sigma \setminus \sqrt{n}$, where σ denotes the standard deviation, and n denotes the number of cross-validation folds.

Table 3. Benchmark scoring metrics.

	von Heijne	SVM
ACC	0.97	0.97
F1	0.60	0.62
MCC	0.60	0.62
Precision	0.52	0.52
Recall	0.72	0.76

Note: ACC: accuracy; MCC: Matthews Correlation Coefficient; SVM: support vector machine.

numerical balance between the prediction classes retrieved, in order 7108 TN, 139 FP, 59 FN, 150 TP for von Heijne, and 7106 TN, 141 FP, 51 FN, 158 TP for SVM.

3.2.1 False positive evaluation

Both strategies FPR was found to be 0.02, reflecting indeed the numerical balance in term of number of FP and TN retrieved in both approaches. The same observation is remarked in the computation of ratio (12), even though a minimal difference is observed (0.22 for von Heijne, 0.23 for SVM). In particular, we retrieved 40 FP TM proteins over 182 total ones in the von Heijne method, and 42 FP TM proteins over 180 total in the SVM classification. These rates reflect indeed one of major challenges in the SP prediction field, the discrimination between SPs hydrophobic core and the hydrophobic regions of amphipathic TM proteins (Lao *et al.*, 2002). This on-going issue was addressed recently by introducing deep learning in TM proteins topology prediction (Hallgren *et al.*, 2022), which may also contribute in improving SPs prediction methods (i.e. lowering therefore the error rate value).

The FPR was also computed for transit peptides:

1. von Heijne
    - Mitochondria FPR: 0.02 (18 FP mTPs over 704 total).
    - Chloroplast FPR: 0.03 (23 FP cTPs over 647 total).
    - Peroxisome FPR: 0.00 (0 FP PTS2 over 5 total).
    - Total FPR: 0.03 (41 FP over 1345 total).
2. SVM
    - Mitochondria FPR: 0.05 (34 FP mTPs over 688 total).
    - Chloroplast FPR: 0.02 (15 FP cTPs over 655 total).
    - Peroxisome FPR: 0.00 (0 FP PTS2 over 5 total).
    - Total FPR: 0.04 (49 FP over 1337 total).

The erroneous recognition as transit peptide proteins still depends on the hydrophobic composition of this type of sorting signals (Roise *et al.*, 1986; Roise and Schatz, 1988; von Heijne *et al.*, 1989; Petriv *et al.*, 2004). Indeed, related studies show that eukaryotic SPs are remarkably more hydrophobic with respect to intra-organelle sorting signals, from which we can theorize that these errors involve transit proteins with an hydrophobicity content comparable to that of a typical secretory protein.

3.2.2 False negative evaluation

The graphical representation of the consensus sequence derived from the analysis of the von Heijne FN discloses hints that may have contributed to this incorrect classification (Supplementary materials Fig. 4). In fact, upon comparison with the consensus sequence derived from true SPs (which is very similar to the benchmark dataset sequence logo), we observe that the preponderance of the most represented residues Leu and Ala is not evident anymore, especially along the hydrophobic core region (von Heijne, 1985). In particular, the observed frequency of hydrophobic residues Leu, Ala, Iso, Val, Phe is now comparable to that of hydrophilic Gly, Ser and Thr (Kyte and Doolittle, 1982). We may derive that SPs with low hydrophobic character in the window [-13, +2] may be erroneously classified due to this very reason.

On the other hand, the length distribution of SVM FN reveals the existence of "longer than expected" SPs (Supplementary materials Fig. 5). Indeed, these "hard to detect" signals (Hiss and Schneider, 2009) are characterized by a mean length of 27 residues, in opposition with the standard 23-residues long SP. The trend of FN distribution also appears more uniformly organized with respect to that of true SPs, which is confirmed by the corresponding standard deviation values, 8.28 for FN vs 4.92 for TP. From the FN 25^{th} percentile (corresponding to the SP length value 20), we may infer that more than 50% of FN sequences are indeed long SPs. Composition-wide speaking (still, with respect to true SPs), the significantly higher observed frequency of the hydrophilic residues Arg, Asn, Trp, and Pro matches the frequency-loss of typically highly-expressed residues Leu and Ala (Supplementary materials Fig. 6).

A taxonomic-dependent insight regarding both methodologies reveals that the majority of FN are metazoans (Supplementary materials Fig. 7):

1. von Heijne
    - Metazoa: 47.
    - Fungi: 7.
    - Plants: 4.
    - Other: 1.
2. SVM
    - Metazoa: 43.
    - Fungi: 3.
    - Plants: 5.

Due to the kingdoms distribution observed in the overall dataset, this predominance is not surprising at all (refer to section 2.1.1). However, as differences between SPs of various eukaryotic kingdoms has not been explored yet, we cannot entirely explain why this happens. Developing a strategy able to capture the composition of SP sequences integrating the kingdom of belonging may contribute solving this open question.

4 Conclusion

In this paper, we presented a comparative benchmark approach in order to confront the performances of two known SP prediction tools, i.e. von Heijne method and support vector machine. Both methodologies accuracy in predicting the position of the cleavage site was proven to be indeed similar and average with respect to other currently available implementations. Furthermore, an in-depth assessment of both approaches allow us to better explore one of the major challenges in the field, the discrimination between SPs and other similarly hydrophobic N-terminal signals/structures. Solving this issue will therefore greatly improve SP tools detection skills and overall performance, deepening at the same time our knowledge on related topics, such as subcellular localization and specificity in cellular sorting mechanisms.

References

(2021). Uniprot: the universal protein knowledgebase in 2021. *Nucleic acids research*, **49**(D1), D480–D489.

Almagro Armenteros, J. J. *et al.* (2019). Signalp 5.0 improves signal peptide predictions using deep neural networks. *Nature biotechnology*,

37(4), 420–423.

Blobel, G. and Dobberstein, B. (1975). Transfer of proteins across membranes. i. presence of proteolytically processed and unprocessed nascent immunoglobulin light chains on membrane-bound ribosomes of murine myeloma. *The Journal of cell biology*, **67**(3), 835–851.

Chicco, D. and Jurman, G. (2020). The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics*, **21**(1), 1–13.

Consortium, U. *et al.* (2018). Uniprot: the universal protein knowledgebase. *Nucleic acids research*, **46**(5), 2699.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, **20**(3), 273–297.

Crooks, G. E. *et al.* (2004). Weblogo: a sequence logo generator. *Genome research*, **14**(6), 1188–1190.

Hallgren, J. *et al.* (2022). Deeptmhmm predicts alpha and beta transmembrane proteins using deep neural networks. *bioRxiv*.

Harris, C. R. *et al.* (2020). Array programming with numpy. *Nature*, **585**(7825), 357–362.

Henikoff, J. G. and Henikoff, S. (1996). Using substitution probabilities to improve position-specific scoring matrices. *Bioinformatics*, **12**(2), 135–143.

Hiss, J. A. and Schneider, G. (2009). Architecture, function and prediction of long signal peptides. *Briefings in bioinformatics*, **10**(5), 569–578.

Jain, E. *et al.* (2009). Infrastructure for the life sciences: design and implementation of the uniprot website. *BMC bioinformatics*, **10**(1), 1–19.

Kyte, J. and Doolittle, R. F. (1982). A simple method for displaying the hydropathic character of a protein. *Journal of molecular biology*, **157**(1), 105–132.

Lao, D. M. *et al.* (2002). The presence of signal peptide significantly affects transmembrane topology prediction. *Bioinformatics*, **18**(12), 1562–1566.

Matthews, B. W. (1975). Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure*, **405**(2), 442–451.

Nielsen, H. *et al.* (2019). A brief history of protein sorting prediction. *The protein journal*, **38**(3), 200–216.

pandas development team, T. (2020). pandas-dev/pandas: Pandas.

Pedregosa, F. *et al.* (2011). Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, **12**, 2825–2830.

Petriv, I. *et al.* (2004). A new definition for the consensus sequence of the peroxisome targeting signal type 2. *Journal of molecular biology*, **341**(1), 119–134.

Roise, D. and Schatz, G. (1988). Mitochondrial presequences. *Journal of Biological Chemistry*, **263**(10), 4509–4511.

Roise, D. *et al.* (1986). A chemically synthesized pre-sequence of an imported mitochondrial protein can form an amphiphilic helix and perturb natural and artificial phospholipid bilayers. *The EMBO journal*, **5**(6), 1327–1334.

Savojardo, C. *et al.* (2018). Deepsig: deep learning improves signal peptide detection in proteins. *Bioinformatics*, **34**(10), 1690–1696.

von Heijne, G. (1983). Patterns of amino acids near signal-sequence cleavage sites. *European journal of biochemistry*, **133**(1), 17–21.

von Heijne, G. (1984). How signal sequences maintain cleavage specificity. *Journal of molecular biology*, **173**(2), 243–251.

von Heijne, G. (1985). Signal sequences: the limits of variation. *Journal of molecular biology*, **184**(1), 99–105.

von Heijne, G. (1986). A new method for predicting signal sequence cleavage sites. *Nucleic acids research*, **14**(11), 4683–4690.

von Heijne, G. (1990). The signal peptide. *The Journal of membrane biology*, **115**(3), 195–201.

von Heijne, G. *et al.* (1989). Domain structure of mitochondrial and chloroplast targeting peptides. *European Journal of Biochemistry*, **180**(3), 535–545.

Walter, P. *et al.* (1981). Translocation of proteins across the endoplasmic reticulum. i. signal recognition protein (srp) binds to in-vitro-assembled polysomes synthesizing secretory protein. *The Journal of cell biology*, **91**(2), 545–550.

Wang, M. *et al.* (2005). Using string kernel to predict signal peptide cleavage site based on subsite coupling model. *Amino Acids*, **28**(4), 395–402.