Egyptian Journal of Medical Human Genetics

RESEARCH Open Access

Protein secondary structure prediction (PSSP) using different machine algorithms

Heba M. Afify[1], Mohamed B. Abdelhalim[2], Mai S. Mabrouk[3*] and Ahmed Y. Sayed[4]

Abstract

Background: The computational biology approach has advanced exponentially in protein secondary structure prediction (PSSP), which is vital for the pharmaceutical industry. Extracting protein structure from the laboratory has insufficient information for PSSP that is used in bioinformatics studies. In this paper, the support vector machine (SVM) model and decision tree are presented on the RS126 dataset to address the problem of PSSP. A decision tree is applied for the SVM outcome to obtain the relevant guidelines possible for PSSP. Furthermore, the number of produced rules was fairly small, and they show a greater degree of comprehensibility compared to other rules. Several of the proposed principles have compelling and relevant biological clarification.

Results: The results confirmed that the existence of a particular amino acid in a protein sequence increases the stability for the forecast of protein secondary structure. The suggested algorithm achieved 85% accuracy for the E|~E classifier.

Conclusions: The proposed rules can be very important in managing wet laboratory experiments intended at determining protein secondary structure. Lastly, future work will focus mainly on large protein datasets without overfitting and expand the amount of extracted regulations for PSSP.

Keywords: Support vector machine, Protein structure prediction, Decision tree

Background

Proteins are diverse in shape and molecular weight and are relevant to their function and chemical bonds [1]. Therefore, there are various types of proteins according to their benefits and applications [2]. There are some factors that lead to mutations in the protein shape and lack of protein function, including temperature variations, pH, and chemical reactions [3]. According to the polypeptide structure, proteins are categorized into four classes: primary, secondary, tertiary, and quaternary. Analysis of protein behavior can be difficult due to next-generation sequencing (NGS) technology, time-consuming, and low accuracy, especially for non-homologous protein sequences. Therefore, deep learning algorithms are applied to handling huge datasets for computational protein design by predicting the probability of 20 amino acids in a protein [4]. Because the experimental biologist suffered from the limited availability of 3D protein structure, protein structure prediction is effectively used to define 3D protein structure that supports more genetic information [5]. The prognosis of protein 3D structure from the amino acid sequence has several applications in biological processes such as drug design, discovery of protein function, and interpretation of mutations in structural genomics [6].

Protein folding is a thermodynamic process to create a 3D structure via minimum energy conformation based on entropy [7]. The traditional methods for studying protein folding are minutely discussed [8]. On the other hand, the computational procedures of protein folding are focused on the prediction of protein stability, kinetics, and structure by using Levinthal's paradox or energy landscape or molecular dynamics [9]. The common algorithm is the dictionary secondary structure protein (DSSP) [10], which is based on hydrogen bond

* Correspondence: msm_eng@yahoo.com
[3]Misr University for Science and Technology (MUST), 6th of October City, Egypt
Full list of author information is available at the end of the article

estimation. The DSSP algorithm assigns protein secondary structure to eight various groups: H (α-helix), E (β-strand), G (310-helix), I (π-helix), B (isolated β-bridge), T (turn), S (bend), and (rest). This algorithm holds more information for a range of applications, but it is more complex for computational analysis.

Previously, Pauling et al. [11] presented a PSSP model for recognizing the polypeptide backbone by separating two regular states, α-helix (H) and β-strand (E). The poor PSSP relied on training large datasets that lead to overfitting and classifier inability to estimate unknown datasets [12].

Yuming et al. [13] applied a PSSP model by using the data partition and semi-random subspace method (PSRSM) with a range of accuracy of 85%. Generally, machine learning algorithms are implemented for PSP, but the evaluated accuracy is still limited [14]. To improve the PSSP model, several algorithms used neural networks (NNs) [15], K-nearest neighbors (KNNs) [16], and SVMs [17]. Additionally, deep learning algorithms such as deep conditional neural fields (CNF) [18], MUFOLD-SS [19], and SPINE-X [20] have achieved success with an accuracy of 82–84%.

Also, the output of SVM is employed as input features for a decision tree to extract the rules governing PSSP [21] with high accuracy. It was found that the accuracy rate of protein prediction is based on the gap between current rules from algorithms and rules from biological meaning.

In this work, we developed a former technique [21] by using an SVM model to guess the protein secondary structure and using a decision tree for SVM production to derive regulations surrounding PSSP.

Methods

Data description

The proposed model implemented 126 protein sequences (RS126 set) [22] to predict the PSSP. The dataset contains 23,349 amino acids that formed from 32% α-helices, 23% β-strands, and 45% coils. The proposed model is designed under MATLAB R2010a version 7.10.0 using a Windows platform with an Intel Core i7-6700T@ 2.8.

Proposed model for PSSP

Figure 1 displays the proposed model for PSSP. The following steps explained the four steps of the proposed model.

- The first step includes converting the amino acid residue into a binary number by orthogonal encoding.
- In the second step, the dataset is divided into seven sets using seven-fold cross-validation by the SVM classifier.
- In the third step, compute the accuracy of prediction and select such results with high accuracy and pass it as a training set into the decision tree
- In the fourth step, those rules that are produced by the decision tree are extracted and recorded.

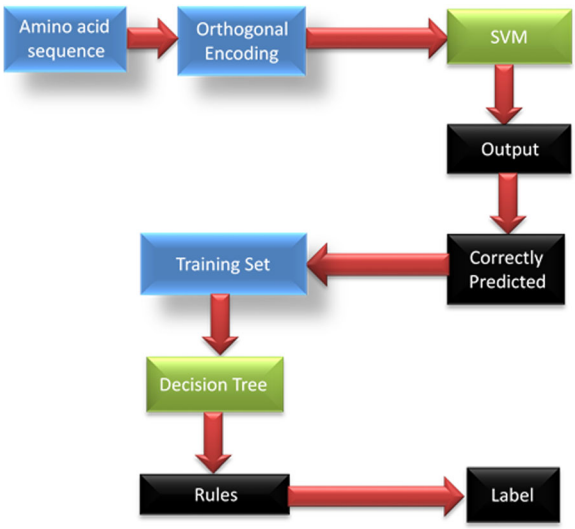

Fig. 1 Architecture of the PSSP model

Orthogonal encoding

Orthogonal encoding was used to convert the amino acid residues to numerical values and to read the inputs of the sliding window. In this paper, a window of size 12 is adopted; in the sliding window method, only the central amino acid is predicted, and binary encoding was utilized to allocate numeric data to the amino acid characteristics. Therefore, there are 20 locations for the characteristics of amino acids. For example, for every window of size 12, the window comprises 12 input amino acids, each amino acid will be denoted by the value 1 depending on its location in the window, and each other location will be assigned 0's. In this case, the input pattern will be 20 × 12 inputs, 12 of which will be assigned the value 1 and all others to 0's. A good example of the sliding window problem is shown in Fig. 2; suppose our input pattern consists of the following protein sequences and secondary structure pattern. If the window size is 7 and the pattern NTDEPGA in Fig. 2 is assumed to be the training pattern, it is applied to estimate the residue 'E' and the next residue 'P' in the window slide 'TDEPGACP.' The window will slide to the next residue until the end of the pattern. The orthogonal

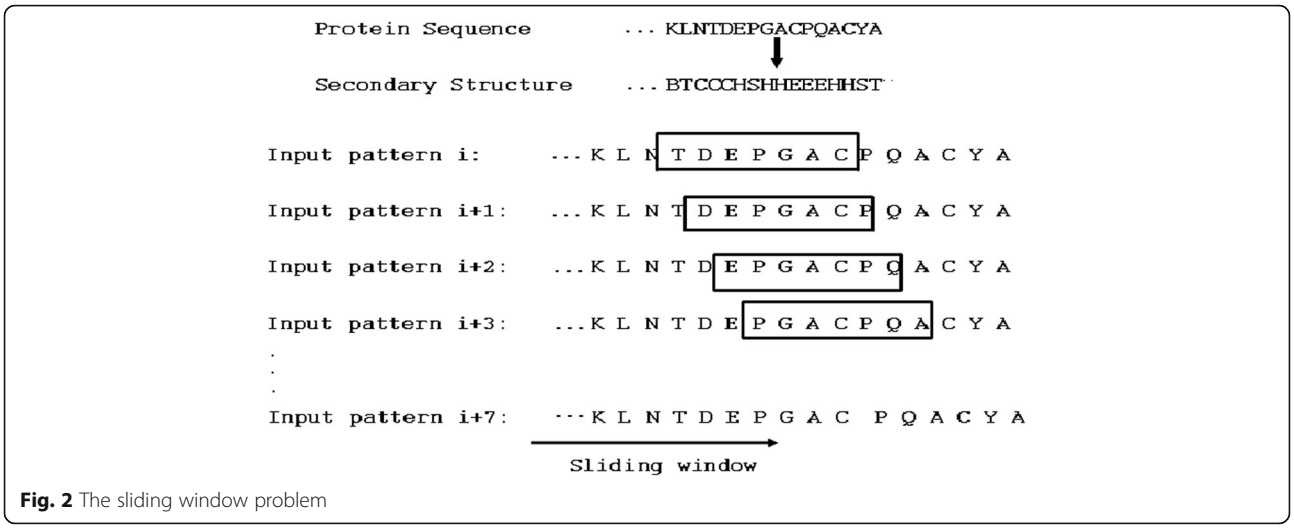

Fig. 2 The sliding window problem

Table 1 Orthogonal encoding of 12 amino acids

K	L	N	T	D	E	P	G	A	C	P	Q	A	C	Y	A
0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

encoding of the pattern KLNTDEPGACPQACYA is shown in Table 1.

In this work, a DSSP [10] model for secondary structure assignment is used because it is a frequently utilized and consistent technique for the PSSP approach. To lessen the complexity of assignments and training, the eight classes of DSSP were reduced to three classes [23]. The reduction problem of eight classes to three classes is shown in Table 2.

SVM classifier

The SVM classifier [17] constructs a hyperplane that separates the protein dataset after orthogonal encoding into various classes. Six categories, namely, (H/~H), (H/~E), (E/~E), (E/~C), (C/~C), and (H/~C), are used. For SVM, the selection of kernel function, kernel parameter, and cost parameter (C) are investigated to evaluate the classification accuracy. In this paper, the RBF kernel is used, and the kernel parameter γ is constant throughout the experiment, but the C varies over the following values: 0.2, 0.4, 0.7, 0.9, 1, and 4 as used in a previous study [24].

Table 2 Conversion of eight secondary structures to three classes

Reduction	DSSP	Description
H	H	Alpha-helix
H	G	3-helix
C	I	5-helix
E	B	Isolated bête-bridge
E	E	Extended - strand
C	T	Turn
C	S	Bend
C	None	Coil

Table 3 Accuracy comparison of various algorithms for protein secondary structure prediction

Classifier	Accuracy (%)			
	Proposed model	**PSSP_SVM [26]**	**PSSP_SVMCE [23]**	**PSSP_SVMCP [27]**
H/~H	76.85	80.36	73.90	**87.24**
E/~E	**85.97**	81.25	78.75	85.65
C/~C	62.83	73.20	70.80	**82.54**
H/~E	75.78	-	68.45	**91.50**
H/~C	74.34	-	60.15	**82.03**
E/~C	73.43	-	69.90	-

Decision tree

A decision tree is composed of several nodes and leaves [25]. Each leaf represents one class corresponding to the target value, and the leaf node may take the probability of the target label. A decision tree inducer is an object that takes a training set and creates a model that generates a link between the input instance and the target variable.

Let DT denote a reference of the decision tree and $DT(T)$ denote the classification tree. These symbols are created by applying DT to the training set T. The prediction of the target variable indicates $DT(T)(x)$.

We can use a classifier created by a decision tree inducer to classify an unknown data set in one of the two ways: by allocating it to a specific class or by supplying the probability of given input data belonging to each class variable. We can estimate the conditional probability in the decision tree by $\widehat{P}\,DT(T)(y|a)$ (probability of class variable given an input instance). In a decision tree, the probability is evaluated for each leaf node distinctly by computing the occurrence of the class through the training samples according to the leaf node.

When a particular class never appears in a specific leaf node, we may end with a zero probability. However, we can avoid such a case by using Laplace rectification.

Laplace's law states the likelihood of the event $j = x_i$ where j is a random parameter and x_i is a potential output of j that has been noticed n_i times out of n notices. It is given by: $\frac{n_i+wp}{n+w}$ where p is the prior probability of the event and w is the pattern size that refers to the weight of the prior estimation according to the noticed data. Additionally, w is described as equivalent pattern size because it denotes the increase of the n tangible notices by other w practical patterns estimated relative to p. Due to assumptions, we can rewrite the prior and posterior probability in the following equations:

$$\frac{n_i + w \cdot p}{n + w}$$

$$\frac{n_i}{n} \cdot \frac{n}{n+w} + p \cdot \frac{w}{n+w}$$

$$p_p \cdot \frac{n_i}{n+w} + p \cdot \frac{w}{n+w}$$

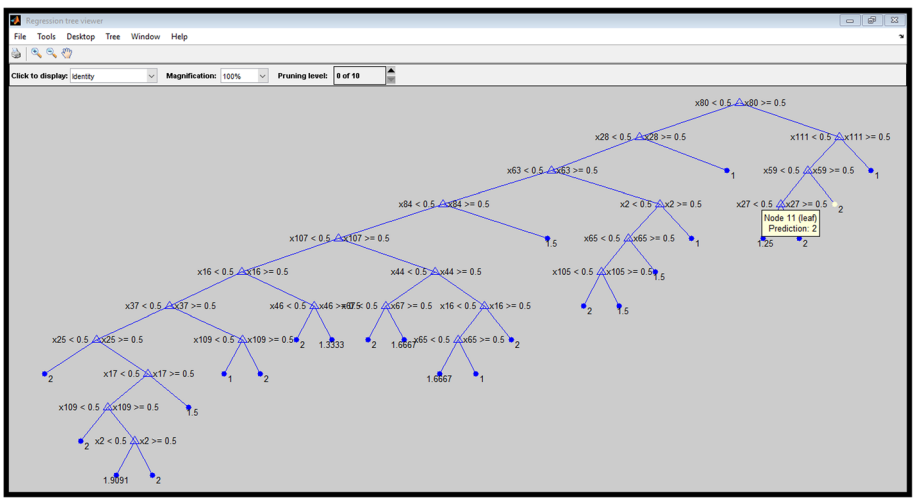

Fig. 3 Screenshot of the decision tree

$$p_p \cdot n_1 + p \cdot n_2$$

In this case, we used the following expression:

$$P_{laplace}(a_i|y) = \frac{|T| + w.p}{|T| + w}$$

By utilizing this expression, the values of p and w are chosen.

Rules' confidence

To define the trust of the rules, we must create the probability allocation that controls the accuracy calculation. The classification task is modeled as a binomial test.

Suppose the test set consists of N records, X is the quantity of sample portions accurately prophesied by the system and p is the correct accuracy of the system. The forming the overall function as a binomial ranking by mean p and variance $p(1 - p)/N$ based on the normal ranking, the empirical accuracy for rules' confidence can be derived from.

$$P\left(-Z_{\alpha/2} \le \frac{ecc-p}{\sqrt{p(1-p)/N}} \le Z_{1-\alpha/2}\right) = 1-\alpha$$

Table 4 Rules produced by the decision tree for the H/~H classifier

Rules		Class
1	if x80<0.5 then node 2 elseif x80>=0.5 then node 3 else 1.81	α
2	if x28<0.5 then node 4 elseif x28>=0.5 then node 5 else 1.84091	
3	if x111<0.5 then node 6 elseif x111>=0.5 then node 7 else 1.58333	
4	if x63<0.5 then node 8 elseif x63>=0.5 then node 9 else 1.85057	
5	if x59<0.5 then node 10 elseif x59>=0.5 then node 11 else 1.7	
6	if x65<0.5 then node 20 elseif x65>=0.5 then node 21 else 1.71429	
7	if x16<0.5 then node 22 elseif x16>=0.5 then node 23 else 1.91525	
8	if x44<0.5 then node 24 elseif x44>=0.5 then node 25 else 1.77778	
9	if x105<0.5 then node 26 elseif x105>=0.5 then node 27 else 1.8	
	fit = 1	

where $-Z_{\alpha/2}$ and $Z_{1-\alpha/2}$ are the high and low bounds provided from a normal ranking at a trust interval of $(1-\alpha)$.

Results

Evaluation criteria for secondary structure prediction

To find optimal rules governing PSSP by the decision tree, a Q_3 accuracy measure is used to estimate the value of exactly predicted secondary structural elements of the protein sequence.

$$Q_3 = \frac{\sum_{i\in(H,E,C)} \text{number of correctly predicted residue}}{\sum_{i\in(H,E,C)} \text{number of secondary structure elements observe}}$$

Performance of SVM

The results of the experiment are summarized in Table 3. A comparison of the accuracy obtained by PSSP based on NMR chemical shift with SVM (PSSP_SVM) [26], based on the codon encoding (CE) scheme with SVM (PSSP_SVMCE) [23], and based on the compound pyramid (CP) model with SVM (PSSP_SVMCP) [27].

From Table 3, the accuracy among the classifiers varies significantly. In the proposed model, the prediction accuracy is in the range of 85–63%. The best prediction accuracy is recoded for the E/~E classifier, and the least accuracy of the prediction is recorded for the C/~C classifier. For the PSSP_SVMCP method [27], the best prediction accuracy is recoded for the H/~H, C/~C, H/~E, and H/~C classifiers compared to the proposed model.

The proposed model achieved the best prediction accuracy compared to other previous models such as PSSP_SVM [26], and PSSP_SVMCE [23]. In contrast, the PSSP_SVMCP [27] model achieved the best prediction accuracy compared to the proposed model.

During the experiment, the general observation is made and observes that the accuracy of the classifier increases with an increase of the C. The better C is obtained at 4 and the least C is also obtained at 0.1.

Performance of decision tree

Figure 3 displays the decision tree of the training dataset extracted from the SVM algorithm. Tables 4, 5, and 6 show some of those rules produced by the decision tree using three different categories (H/~H, E/~E, and C/~C). The x variable specifies the column number, the compared values denote the column's data, and the nodes specify the nodes of the tree. Figures 4, 5, and 6 show the percentage of prediction accuracy related to the proposed rules with a bold symbol referring to the

Table 5 Rules produced by the decision tree for the E/~E classifier

Rules		Class
1	if x37<0.5 then node 28 elseif x37>=0.5 then node 29 else 1.94	
2	if x46<0.5 then node 30 elseif x46>=0.5 then node 31 else 1.77778	
3	if x67<0.5 then node 32 elseif x67>=0.5 then node 33 else 1.90909	
4	if x16<0.5 then node 34 elseif x16>=0.5 then node 35 else 1.57143	β
5	if x30<0.5 then node 2 elseif x30>=0.5 then node 3 else 1.98	
6	if x79<0.5 then node 4 elseif x79>=0.5 then node 5 else 1.9899	
	fit = 2	

Table 6 Rules produced by the decision tree for the C/~C classifier

Rules	Class
1 1 if x86<0.5 then node 10 elseif x86>=0.5 then node 11 else 1.83721	
2 if x111<0.5 then node 12 elseif x111>=0.5 then node 13 else 1.63636	
3 if x107<0.5 then node 14 elseif x107>=0.5 then node 15 else 1.89063	
4 if x121<0.5 then node 16 elseif x121>=0.5 then node 17 else 1.68182	coil
5 if x16<0.5 then node 18 elseif x16>=0.5 then node 19 else 1.77778	
6 if x46<0.5 then node 20 elseif x46>=0.5 then node 21 else 1.92857	
7 if x42<0.5 then node 22 elseif x42>=0.5 then node 23 else 1.625	
8 if x2<0.5 then node 24 elseif x2>=0.5 then node 25 else 1.77778 fit = 2	

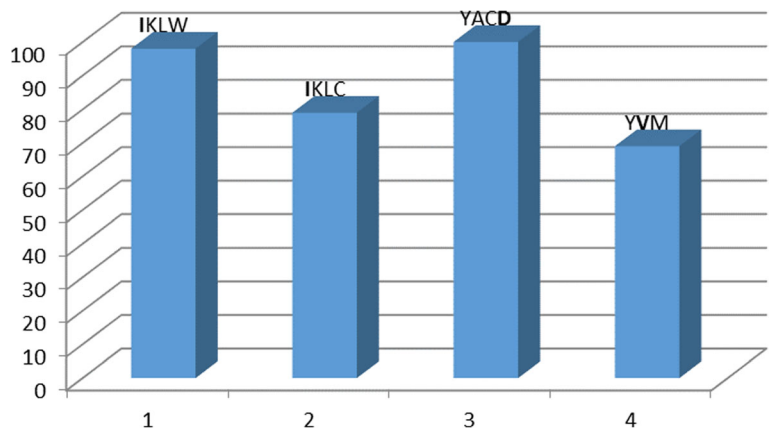

Fig. 4 Rules extracted for the PSSP model using the location of the α-helix

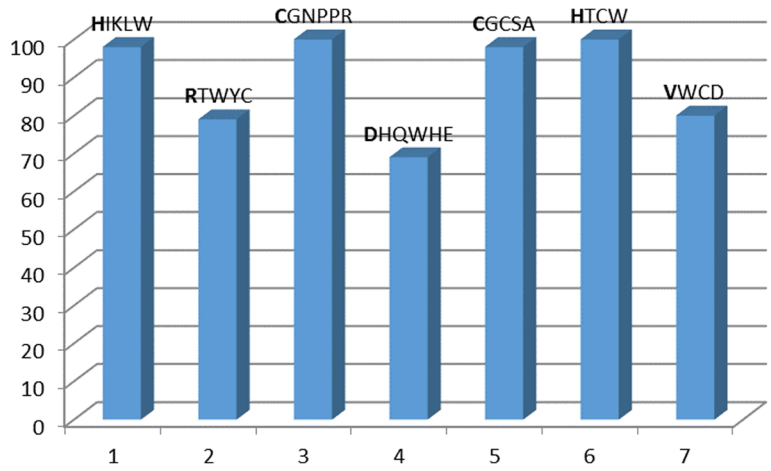

Fig. 5 Rules extracted for the PSSP model using the location of the β-helix

amino acid pattern that created a special protein secondary structural type.

Discussions

Initially, it was noted that the relationship between hydrophobic side chains could lead to α-helix occurrence [28]. In Fig. 4, the forecast of the α-helix is based on four rules according to four patterns, namely, **I**KLW, **I**KLC, YAC**D**, and Y**V**M. In rule 1, the **I**KLW pattern achieved 100% accuracy for α-helix prediction due to isoleucine I, lysine K, leucine L, and tryptophan W displays at the first, second, third, and fourth locations, respectively. Both amino acids I and W are hydrophobic, and their presence at location *i*, *i* + 3 referred to a helix manifestation [29]. In rule 2, the **I**KLC pattern confirmed that I and C are hydrophobic and indicated helix stabilization [29]. In rule 3, the YAC**D** pattern achieved 100% accuracy for α-helix prediction. In rule 4, both amino acids Y and M are hydrophobic, and their occurrence at two locations during the sequence leads to α-helix construction. Valine V has a low rate of helix occurrence [28].

In Fig. 5, the forecast of the β-strand is based on seven rules according to seven patterns, namely, **H**IKLW, **R**TWYC, **C**GNPPR, **D**HQWHE, **C**GCSA, **H**CTW, and **V**WCD. In rule 1, the **H**IKLW pattern achieved 100% accuracy for β-strand prediction due to histidine H, isoleucine I, lysine K, leucine L, and tryptophan W displays at the first, second, third, fourth, and fifth locations, respectively. In rule 2, the **R**TWYC pattern achieved 79% accuracy due to arginine R, threonine T, tryptophan W, tyrosine Y, and cysteine C displays at the first, second,

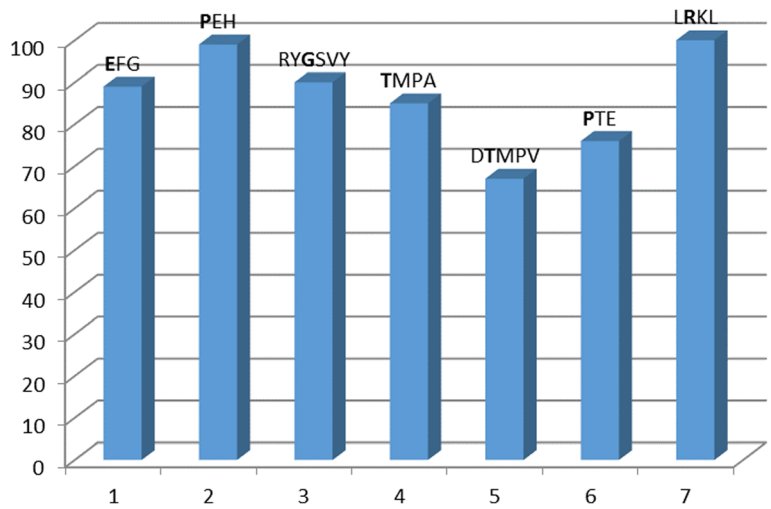

Fig. 6 Rules extracted for the PSSP model using the location of the coil structure

third, fourth, and fifth locations, respectively. The amino acids T, R, and D are employed as N-terminal β-breakers, while S and G are employed as C-terminal β-breakers [30]. Additionally, these patterns, namely, **C**GNPPR, **C**GCSA, and **H**CTW, achieved 100% accuracy for β-strand prediction.

The strengthening of protein structure and protein regulation is related to the appearance of specific amino acids in the loop structure. Proline P and glycine G are considered the most important amino acids in the loop structure. The high load proclivities are achieved when there are nearest to Proline P [30]. On the other hand, low load proclivities are achieved when cysteine C, isoleucine I, leucine L, tryptophan W, and valine V are present [31].

In Fig. 6, the forecast of the coil structure is based on seven rules according to seven patterns, namely, **E**FG, **P**EH, RY**G**SVY, **T**MPA, **D**TMPV, **P**TE, and L**R**KL. In rules 2 and 3, the occurrence of coil structure referred to high load proclivities due to the presence of amino acids P and G. In rule 1, the **E**FG pattern achieved 90% accuracy for coil prediction. This confirmed that E is considered hydrophilic, while F and G are hydrophobic amino acids. In rule 2, the **P**EH pattern achieved 100% accuracy. In rule 4, it was confirmed that T is hydrophilic, while M, P, and A are hydrophobic amino acids. In rule 6, the **P**TE pattern achieved 67% accuracy due to proline P occurrence with threonine T and glutamic E during the series. In rule 7, the L**R**KL pattern achieved 100% accuracy due to the arginine R display at a location with lysine K and leucine L at the first, third, and fourth locations, respectively, through the series.

For comparative analysis, the recent algorithm [32] based on convolutional, residual, and recurrent neural network (CRRNN) showed 71.4% accuracy for DSSP. This indicated that our algorithm is more accurate than that in [30]. On the other hand, the quality of protein structure prediction can affect poor alignments, protein misfolding, few similarity rates between known sequences, evolution theory, and machine learning performance [33].

For results analysis, instead of taking the three binary classifiers: (H/~H), (E/~E), (C/~C) into PSSP account [21], we compared the proposed algorithm with previous studies based on six classifiers: (H/~H), (H/~E), (E/~E), (E/~C), (C/~C), (H/~C) for PSSP as in Table 3 and predict the residue identity of each position one by one. It also found that the PSSP_SVMCP model has shown superior accuracy rather than the proposed model in terms of H/~H, C/~C, H/~E, and H/~C classifiers.

Conclusions

The goal of this paper is to predict the RS126 dataset of 126 protein sequences as a secondary structure via the SVM classifier and decision tree. The proposed model has presented a framework of PSSP for the appearance of α-helix, β-strand, and coil structures. The experiential results coincided with the work of Kallenbach that the presence of isoleucine I and tryptophan W at positions i and i+3 along the sequence proved to be a helix stabilizing. In a β-strand, the presence of arginine R and lysine K is proven to be β-strand. In a coil structure, it is known that proline P and glycine G are the most significant amino acids in the coil structure, which concurs with our findings. This proposed model obtained benefits in the protein analysis domain with a correct prognosis for anonymous sequences. In future work, we expand the proposed algorithm to apply it to the other protein datasets for producing an effective competitive analysis in the PSSP schema.

Abbreviations

SVM: Support vector machine; PSSP: Protein secondary structure prediction; DSSP: Dictionary Secondary Structure Protein; NNs: Neural networks; KNNs: K-nearest neighbors; CNF: Conditional neural fields; NGS: Next-generation sequencing; PSSP_SVM: Protein secondary structure prediction based on support vector machine; PSSP_SVMCE: Protein secondary structure prediction based on support vector machine and codon encoding (CE) scheme; (PSSP_SVMCP): Protein secondary structure prediction based on support vector machine and compound pyramid (CP) model; C: Cost parameter

Acknowledgements

Not applicable.

Authors' contributions

HA and MM designed the methodology. MA and AS analyzed the data. All authors shared in writing the manuscript and read and approved the final version of this manuscript.

Funding

No funding was received.

Availability of data and materials

The datasets used and analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]Systems and Biomedical Engineering Department, Higher Institute of Engineering, El-Shorouk City, Cairo, Egypt. [2]College of Computing and Information Technology (CCIT), Arab Academy for Science Technology and Maritime Transport (AASTMT), Cairo, Egypt. [3]Misr University for Science and Technology (MUST), 6th of October City, Egypt. [4]Department of Engineering Mathematics and Physics, Faculty of Engineering El-Mataria, Halwan University, Cairo, Egypt.

Received: 16 January 2021 Accepted: 27 April 2021
Published online: 07 June 2021

References

1. Anand N, Huang P (2018) Generative modeling for protein structures. In: Advances in neural information processing systems, pp 7494–7505
2. Zhang Y (2009) Protein structure prediction: when is it useful? Curr Opin Struct Biol 19(2):145–155. https://doi.org/10.1016/j.sbi.2009.02.005
3. AlQuraishi M (2019) End-to-end differentiable learning of protein structure. Cell Syst 8(4):292–301. https://doi.org/10.1016/j.cels.2019.03.006
4. Wang J, Cao H, Zhang JZH, Qi Y (2018) Computational protein design with deep learning neural networks. Sci Rep 8(1):6349
5. Hopf TA, Colwell LJ, Sheridan R, Rost B, Sander C, Marks DS (2012) Three-dimensional structures of membrane proteins from genomic sequencing. Cell 149(7):1607–1621. https://doi.org/10.1016/j.cell.2012.04.012
6. Marks DS, Hopf TA, Sander C (2012) Protein structure prediction from sequence variation. Nat. Biotechnol 30(11):1072–1080. https://doi.org/10.1038/nbt.2419
7. Dill KA, MacCallum JL (2012) The protein-folding problem, 50 years on. Science 338(6110):1042–1046. https://doi.org/10.1126/science.1219021
8. Kubelka J, Hofrichter J, Eaton WA (2004) The protein folding 'speed limit'. Curr Opin Struct Biol 14(1):76–88. https://doi.org/10.1016/j.sbi.2004.01.013
9. Dobson CM (2003) Protein folding and misfolding. Nature 426(6968):884–890. https://doi.org/10.1038/nature02261
10. Kabsch W, Sander C (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen bonded and geometrical features. Biopolymers 22:2577–2637
11. Pauling L, Corey RB, Branson HR (1951) The structure of proteins: two hydrogen-bonded helical configurations of the polypeptide chain. Proc Natl Acad Sci USA 37(4):205–211. https://doi.org/10.1073/pnas.37.4.205
12. Rashid S, Saraswathi S, Kloczkowski A, Sundaram S, Kolinski A (2016) Protein secondary structure prediction using a small training set (compact model) combined with a Complex-valued neural network approach. BMC Bioinformatics 17(1):362. https://doi.org/10.1186/s12859-016-1209-0
13. Ma Y, Liu Y, Cheng J (2018) Protein secondary structure prediction based on data partition and semi-random subspace method. Sci Rep 8(1):9856. https://doi.org/10.1038/s41598-018-28084-8
14. Yoo PD, Zhou BB, Zomaya AY (2008) Machine learning techniques for protein secondary structure prediction: an overview and evaluation. Curr Bioinform 3(2):74–86. https://doi.org/10.2174/157489308784340676
15. Malekpour SA, Naghizadeh S, Pezeshk H, Sadeghi M, Eslahchi C (2009) Protein secondary structure prediction using three neural networks and a segmental semi markov model. Math Biosci 217(2):145–150. https://doi.org/10.1016/j.mbs.2008.11.001
16. Tan YT, Rosdi BA (2015) Fpga-based hardware accelerator for the prediction of protein secondary class via fuzzy k-nearest neighbors with lempel–ziv complexity based distance measure. Neurocomputing 148:409–419. https://doi.org/10.1016/j.neucom.2014.06.001
17. Ward JJ, McGuffin LJ, Buxton BF, Jones DT (2003) Secondary structure prediction with support vector machines. Bioinformatics 19(13):1650–1655. https://doi.org/10.1093/bioinformatics/btg223
18. Wang S, Peng J, Ma J, Xu J (2016) Protein secondary structure prediction using deep convolutional neural fields. Sci Rep 6(1):18962. https://doi.org/10.1038/srep18962
19. Fang C, Shang Y, Xu D (2018) MUFOLD-SS: New deep inception-inside-inception networks for protein secondary structure prediction. Proteins 86(5):592–598. https://doi.org/10.1002/prot.25487
20. Faraggi E, Zhang T, Yang Y, Kurgan L, Zhou Y (2012) SPINE X: improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles. J Comp Chem 33(3):259–267. https://doi.org/10.1002/jcc.21968
21. Muhamud AI, Abdelhalim MB, Mabrouk MS (2014) Extraction of prediction rules: Protein secondary structure prediction. In: 10th International Computer Engineering Conference (ICENCO), 29-30 Dec. 2014, Giza, Cairo, Egypt
22. Hobohm U, Scharf M, Schneider R, Sander C (1992) Selection of representative protein data sets. Protein Sci 1(3):409–417. https://doi.org/10.1002/pro.5560010313
23. Zamani M, Kremer SC (2012) Protein secondary structure prediction using supporting vector machine and codon encoding scheme. In: 2012 IEEE international conference on bioinformatics and biomedicine workshop, pp 22–27
24. Hua S, Sun Z (2001) A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach. J Mol Biol 308(2):397–407. https://doi.org/10.1006/jmbi.2001.4580
25. Freund Y, Mason L (1999) The alternating decision tree learning algorithm. In: Proceeding of the sixteenth international conference on machine learning, pp 124–133
26. Ahmed S, Abdel A, Reza S (2010) Prediction of protein secondary strucutre based on NMR chemical shift data using supporting vector machine. In: 12th international conference on computer modelling and simulation
27. Bingru Y, Lijun W, Yun Z, Wu Q (2010) A novel protein secondary structure prediction system based on compound pyramid model. In: Second international conference on information technology and computer science
28. Padmanabhan S, Badwin RL (1994) Helix stabilizing interaction between tyrosine and leucine or valine when the spacing is i, i+4. J Mol Biol 241(5):706–713
29. Lyu PC, Sherman JC, Chen A, Kallenbach NR (1991) α-helix stabilization by natural and unnatural amino acids with alkyl side chains. Proc Natl Acad Sci. USA 88(12):5317–5320
30. Colloch N, Cohen FE (1991) β-breakers: an aperiodic secondary structure. J Mol Biol 221(2):603–613
31. Feng J-a, Crasto CJ (2001) Sequence codes for extended conformation: a neighbor-dependent sequence analysis of loops in proteins. Protein Struct Funct Bioinform 42(3):399–413
32. Zhang B, Li J, Lü Q (2018) Prediction of 8-state protein secondary structures by a novel deep learning architecture. BMC Bioinformatics 19:293
33. Torrisi M, Pollastri G, Le Q (2020) Deep learning methods in protein structure prediction. Comput Struct Biotechnol J, in press

Publisher's Note