Received: September 4, 2020. Revised: October 31, 2020. 232

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Predicting Secondary Structure of Protein Using Hybrid of Convolutional Neural Network and Support Vector Machine

Vincent Michael Sutanto[1] Zaki Indra Sukma[2] Afiahayati Afiahayati[1]*

[1]*Department of Computer Science and Electronics, Universitas Gadjah Mada, Indonesia*
[2]*Business Intelligence Data Engineering Division, Gojek, Indonesia*
* Corresponding author's Email: afia@ugm.ac.id

Abstract: Protein secondary structure prediction is one of the problems in the Bioinformatics field, which conducted to find the function of proteins. Protein secondary structure prediction is done by classifying each sequence of protein primary structure into the sequence of protein secondary structure, which fall in sequence labelling problems and can be solved with the machine learning. Convolutional Neural Network (CNN) and Support Vector Machine (SVM) are 2 methods that often used to solve classification problems. In this research, we proposed a hybrid of 1-Dimensional CNN and SVM to predict the secondary structure of the protein. In this research, we used a novel hybrid 1-Dimensional CNN and SVM for sequence labelling, specifically to predict the secondary structure of the protein. Our hybrid model managed to outperform previous studies in term of Q3 and Q8 accuracy on CB513 dataset.

Keywords: Bioinformatics, Convolutional neural network, Protein secondary structure prediction, Support vector machine.

1. Introduction

Protein is an impactful part of organisms which greatly affects its functions. Protein has many functions, namely facilitating chemical reactions, regulating cell activity, antibodies, cell-binding structural elements, and motor elements [1]. It is also well known that protein structure is influencing the mechanical functions and interactions between proteins, resulting in certain biological phenomenon [2].

Proteins affect the function of cells of organisms through the process of protein synthesis, thus it can be said that the properties of living things are determined by proteins. The process of protein synthesis is carried out through several stages. The Deoxyribonucleic Acid (DNA) chains are opened and one of the chains is transcribed into Messenger Ribonucleic Acid (mRNA). Furthermore, Ribosomes translate the Ribonucleic Acid into amino acid sequences. The amino acids then fold and form the secondary structure of the protein.

In Bioinformatics, the structure of proteins is often determined using several experimental procedures, namely X-Ray Crystallography, Nuclear Magnetic Resonance Spectroscopy (NMR), and Computational Methods. Prediction of the secondary structure of proteins using computational methods can be resolved by predicting each position in the protein primary structure sequence (20 types of amino acids) into its secondary structure sequence (α-helix, 310-helix, B-bridge, B-strand, π-helix, Bend, B-turn, and Loop / Irregular). In machine learning, the process of predicting each position from a sequence and producing a new sequence is often referred to as sequence labelling.

One common approach is to use Convolutional Neural Network (CNN), as it is well known for its performance in spatial-related task and its ability to extract and enrich features from a sequence. However, in protein secondary structure prediction the CNN is often combined with another technique, as in example Generative Stochastic Network [3], Multilayer Shift-and-Stitch and CNN (MUST-CNN) [4], Deep-CNF [5], and Multi-scale CNN [6].

Another approach is by using SVM with Gaussian kernel because of its ability in classifying data with a large number of features as shown in several studies related to protein secondary structure prediction [7-8]. However, these methods share the same common unsolved problem in predicting the secondary structure of the protein, which is a low accuracy score in both Q3 and Q8. Because of that, we tried to increase the accuracy score by proposing a new hybrid method of CNN and SVM to predict the secondary structure of the protein.

A combination of CNN and SVM has been widely used to solve a classification task and showed a good performance in comparison with plain CNN [9-11]. However, most of the researches implemented 2-Dimensional CNN as they were dealing with image classification instead of sequence labelling. Thus, it cannot be used to solve the sequence labelling task, as CNN has to move in 1 direction instead of 2 direction when convoluting through sequences. Some studies [12-13] also used the hybrid CNN and SVM to do text classification, but in contrast, we used it to do a sequence labelling instead of classifying entire sequence into one specific class.

In this research, we proposed a hybrid of 1-Dimensional CNN and SVM to do a sequence labelling task, which is predicting the secondary structure of proteins. 1-Dimensional CNN is used to extract and enrich the features, bringing the data into higher dimensional space and retrieving important features. We also used the CNN to capture long-range interdependencies between each residue in sequences. However, there is a disadvantage when using 1-Dimensional CNN for sequence labelling task. The convolution and the pooling process of the 1-Dimensional CNN will cause the sequence length to be reduced. Therefore, in this study, we implemented the Multilayer Shift-and-Stitch technique [4] so that the length of the sequences does not decrease after the convolution and pooling stages. Afterwards, we exploit the capability of SVM to classify data with high dimensionality, as it is safe to increase the number of features that will be fed into the SVM because the regularisation parameter of SVM will decide which of these features are impactful and which are not. The 1-Dimensional model managed to capture the relation of each sequence's residue and increase the data's features, while the SVM showed a better performance when replacing the dense layer in classifying high dimensional data. The SVM processes the large feature map data generated by CNN and predicts secondary protein structure labels for each position in the sequences. This research is an extended version of the authors' thesis [14, 15].

This study has made specific contributions as follows: (1) A new hybrid architecture of 1-Dimensional CNN and SVM has been introduced in sequence labelling domain, and we used the technique to predict the secondary structure of the proteins. (2) The hybrid model managed to outperform MUST-CNN [4], DeepSeqVec [16], DeepProf with SeqVec [16], Deep-CNF [5], Multi-scale CNN One-Hot Encoded [6], and Bi-RNN Single Model [17] in term of Q8 accuracy. In term of Q3 accuracy, our model managed to outperform SVM with Genetic Algorithm [7], SVM with Sequence Features [8], DeepSeqVec [16], and DeepProf with SeqVec [16].

We organised this paper as follows: Chapter 2 explains the domain problem of predicting the secondary structure of the protein, Chapter 3 shows numbers of related studies, Chapter 4 explains the conventional and the hybrid architecture proposed in this study, Chapter 5 explains the results of this research, Chapter 6 discussed the findings from the results, and Chapter 7 shows the conclusion and possibilities for future work.

2. Related work

Various machine learning models have been used for predicting the secondary structure of the protein. Some works are highly related to this study, namely Generative Stochastic Network [3], MUST-CNN [4], and Support Vector Machine [7-8]. We also compared our work with other studies, namely DeepSeqVec [16], DeepProf +SeqVec [16], Deep-CNF [5], Multi-scale CNN [6], and Bi-RNN Single Model [17].

Zhou and Troyanskaya predicted the secondary structure of proteins using the Generative Stochastic Network (GSN) architecture combined with CNN [3]. GSN was used to study the probability of the relationship between the output and input data obtained from the Markov chain. 6128 sequences of CullPDB dataset were used in this study and divided into 5600 training data, 256 validation data, and 272 test data. The GSN architecture with 3 convolutional layers produced Q8 accuracy of 0.721 ± 0.006% with CullPDB test data and Q8 accuracy of 0.664 ± 0.005% with CB513 benchmark dataset. Our study used the dataset provided by the authors of this study [3].

Z. Lin, J. Lanchantin, and Y. Qi predicted the secondary structure of proteins using a combination of MUST-CNN [4]. The architecture used a technique called Multilayer-Shift-and-Stitch, which

were used to tackle the reduced data resolution problem that occurs because of the convolution and pooling stages in CNN. Two models were formed in this study, namely a small model with 189 feature maps and a large model with 1024 feature maps. The models were trained on 4prot dataset and CullPDB dataset and tested on 4prot dataset and CB513 dataset respectively. The small and large models were tested on 4prot dataset, achieving Q8 accuracy of 0.706 and 0.767 respectively. The large model was also tested with CB513 dataset and achieved Q8 accuracy of 0.684. Our study is an improved implementation of this study [4], as we tried to combine the MUST-CNN architecture with SVM.

Y. Wang, J. Cheng, Y. Liu, and Y. Chen predicted the secondary structure of the protein on the CB513 dataset using the SVM RBF kernel [7]. Amino acid sequences and Position-Specific Scoring Matrix (PSSM) were used as inputs for the SVM. Sliding Window with a size of 13 was used to retrieve 260 features vector. PSSM was used as it can store the evolution information of proteins. Grid search method and genetic algorithms were used to optimize the parameters of SVM. The models reached Q3 accuracy up to 76.11% for the use of genetic algorithms and 76.08% for the use of grid search. Y. Chen, Y. Liu, J. Cheng, and Y. Wang also conducted the prediction of the secondary structure of proteins with SVM [8]. Sliding Window of 13 was used to capture the input feature (Position-Specific Scoring Matrix combined with Sequence Feature) for SVM. CB513 Dataset was used in this study, which 440 are used as a training data and 53 as test data. The model achieved a Q3 accuracy of 78%. These study [7-8] differs from our paper as they use SVM as the main architecture while our study proposed the SVM as a part of a hybrid architecture.

Heinzinger, A. Elnaggar, Y. Wang, C. Dallago, D. Nechaev, F. Matthes, and B. Rost predicted the secondary structure of the protein using DeepSeqVec and DeepProf with SeqVec [16]. These models predicted the secondary structure of proteins by combining CNN and sequence embedding, with the only differences is the second model is an upgraded version of the first model. The embedding used in DeepSeqVec models increased the number of features each position of the sequences into 1024, and the DeepProf + SeqVec to 1074 (50 addition from 20 orthogonal input, 20 evolutionary information, 7 state transition probability from hidden markov model, and 3 local alignment features). These models scored Q8 accuracy of 62.5 ± 0.6 for DeepSeqVec model and 66.0 ± 0.5 for DeepProf with SeqVec model. This study [16] used sequence embedding features as additional features while our study didn't use such a technique. This study [16] also used different CNN architecture to ours.

Wang, J. Peng, J. Ma, and J. Xu used Deep-CNF to predict the secondary structure of the protein [5]. Deep-CNF is a combination of Conditional Random Fields and deep convolutional neural networks aiming to capture relationship in a sequence and also the correlation between each nearby residue. This model was trained with CullPDB dataset and used 21 orthogonal encodings and 21 PSSM as its features. This study [5] differs from our paper by the CNN architecture, which theirs were a combination of Conditional Random Fields and Deep CNN, while ours were MUST-CNN and SVM.

Multi-scale CNN was a model by J. Zhou, H. Wang, Z. Zhao, R. Xu, and Q. Lu [6]. The model was trained using CullPDB dataset and tested on the one-hot encoded feature of CB513 dataset. This technique used a connecting highway between two neighbours of convolutional layers to keep local context that has been retrieved by the lower layers while also getting the relation of long-range sequence. The Multi-scale CNN managed to score 68.3% in Q8. This model differs with our model, which the previous study [6] used a highway connector between each convolutional layer and we didn't use such a technique.

Bi-RNN Single Model was used by A. R. Johansen, C. K. Sønderby, S. K. Sønderby, and O. Winther to predict the secondary structure of the protein [17]. This study implemented a combination of layers, such as the fully-connected layer, Bi-directional Recurrent Neural Network layer, and Conditional Random Field Layer [17]. The model was trained on CullPDB dataset and tested on CB513 dataset. The model reached a Q8 accuracy of 68.5%. This study [17] implemented the most distant and complex architecture compared to ours.

3. Protein secondary structure prediction

Protein is a macromolecule formed by a sequence of different amino acids linked by peptide bonds and covalent bonds. Amino acids themselves are often referred as the primary structure of a protein. There are 20 types of amino acids that compose proteins. Table 1 shows the symbols, abbreviations and amino acids of these amino acids. The secondary structure of a protein can be predicted using its primary structure. The problem of predicting the secondary structure of proteins based on their primary structure is included in the sequence labelling domain.

Sequence labelling Classification is done by predicting the Y label of a secondary protein structure sequence at position i, based on the input in the form

Table 1. Symbols, abbreviations, and name of amino acids

Symbol	Abbreviation	Name
A	Ala	Alanina
C	Cys	Cysteine
D	Asp	Aspartic Acid
E	Glu	Glutamic Acid
F	Phe	Phenylalanine
G	Gly	Glycine
H	His	Histidine
I	Ile	Isoleucine
K	Lys	Lysine
L	Leu	Leucine
M	Met	Methionine
N	Asn	Asparagine
P	Pro	Proline
Q	Gln	Glutamine
R	Arg	Arginine
S	Ser	Serine
T	Thr	Threonine
V	Val	Valine
W	Trp	Tryptophan
Y	Tyr	Tyrosine

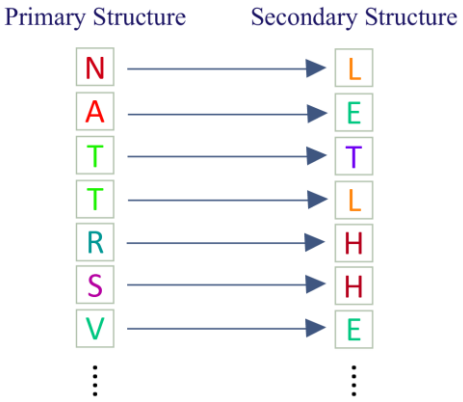

Figure. 1 Protein primary structure with its secondary structure pairs

of X from the protein primary structure sequence at the same position (Fig. 1). The secondary protein structure itself consists of 8 structural classes that can be grouped into 3 main classes, namely the Helix, Sheet, and Coil / Loop as shown in Table 2. The prediction results are evaluated by looking at the Q3 and Q8 accuracy values as shown by Eq. (1) and Eq. (2), where i is the row index of the confusion matrix, j is the column index of the confusion matrix, and P is the value of (i, j) of the confusion matrix. The results are also evaluated by looking at the precision and recall scores.

Table 2. 8 classes and 3 classes of protein secondary structure

8 Class Name	8 Class Symbol	3 Class Name	3 Class Symbol
α-helix 3_{10}-helix π-helix	H G I	*Helix*	H
B-Strand B-bridge	E B	*Sheet*	E
Loop/Irreguler B-Turn Bend	L T S	*Coil / Loop*	C

$$Q3 = \frac{\sum_{i=K} P_{ii}}{\sum_{i=K} \sum_{j=K} P_{ij}}, K \in \{H, E, C\} \quad (1)$$

$$Q8 = \frac{\sum_{i=K} P_{ii}}{\sum_{i=K} \sum_{j=K} P_{ij}}, K \in \{L, B, E, I, S, T, H, G\} \quad (2)$$

4. Data and method

4.1 Datasets

In this study, we used filtered CullPDB dataset [3] and the CB513 dataset [18] as an input and training target for the hybrid of CNN and SVM (see Fig. 2). The filtered CullPDB dataset consists of 5365 sequences of protein primary structure with a length of 700 for each sequence. We divided the data into 4292 training data and 1073 validation data. We used the CB513 Dataset as a test data. This dataset consists of 514 sequences of protein primary structure and a length of 700 for each sequence. These datasets contain features and label for each length of sequence: 21 Position-Specific Scoring Matrix and 21 Orthogonal Input Profile as the input feature, and One-hot encoded label of 8 different class of protein secondary structure as the output label.

4.2 Convolutional neural network

Convolutional Neural Network (CNN) [19] is an artificial neural network that is specifically designed to solve spatial problems. The hidden units contained in CNN often have the same dimensions or size as the processed data.

The hidden units convolve on the data and store relationship information from the data. The information formed by each hidden unit will be stored as a feature-maps, with the number of feature maps that are formed will be as many as the number of hidden units used. The pooling stage is then carried out on the existing feature maps, retrieving dense information from the feature maps. Convolution process with kernel on the data with 2-dimensional input and produce as the output can be written as [19]:

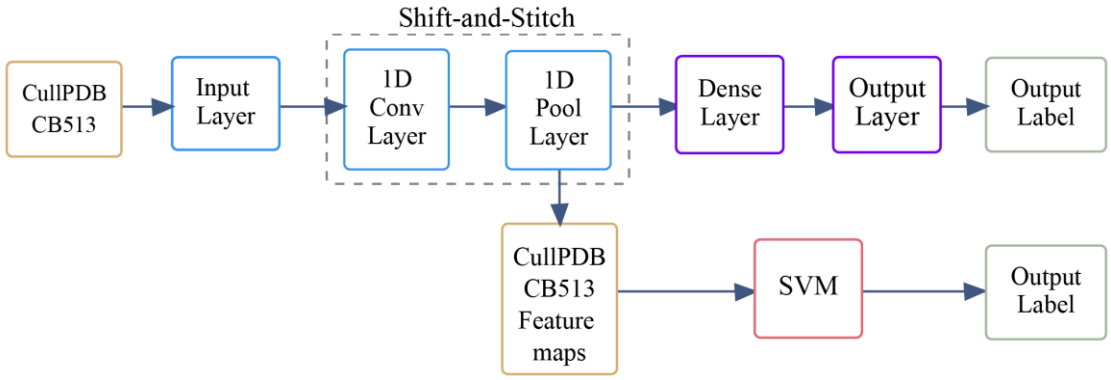

Figure. 2 General Architecture of CNN-SVM Hybrid

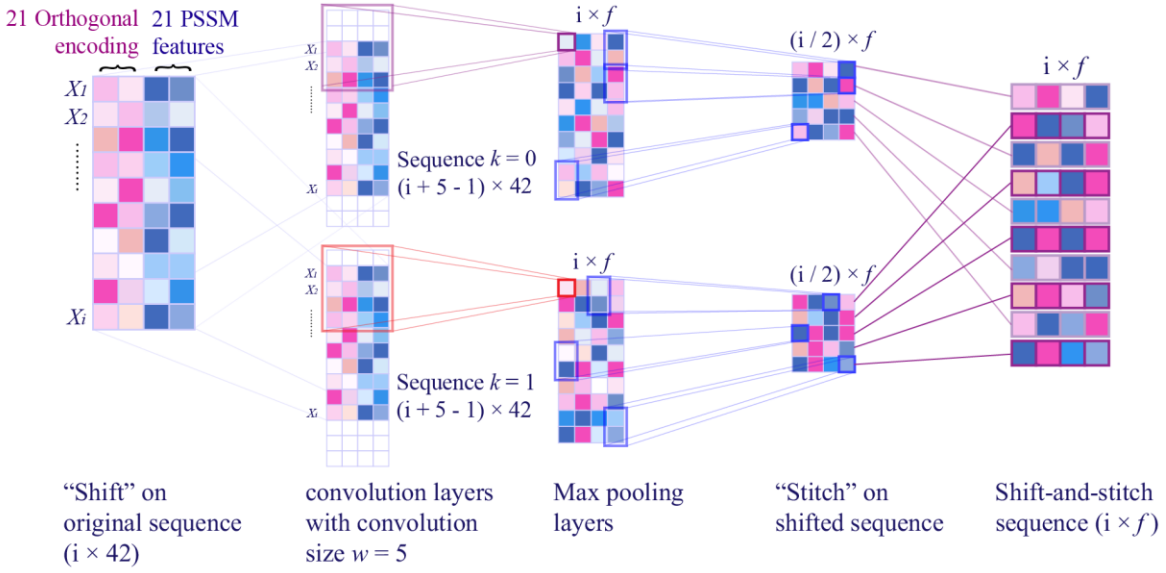

Figure. 3 Shift-and-stitch technique on a single sequence

$$S(i,j) = \sum_m \sum_n I(i+m, j+n) K(m,n) \quad (3)$$

A technique called Shift-and-stitch was introduced to avoid reduction of resolution caused by convolution and pooling stages in CNN [20]. This technique has been modified to be a technique called Multilayer Shift-and-Stitch [4], so it can be applied in sequence labelling task. The shift-and-stitch technique (see Fig. 3) works by duplicating each sequence with a resolution of 700 × 42 as much as the pooling size used (we determined the pooling size p = 2). Then, we give zero-paddings as much as w - 1 for each sequence (at the front and at the back of each sequence, where w denotes the convolution size and k denotes the order of the sequence). The process of adding zero-padding to both sequences is called "Shift". Afterwards, f number of 1-Dimensional convolutional kernels are applied to each sequence, followed by the pooling process. These 2 steps will cause each sequence having a resolution of 350 × f. "Stitch" aims to make the sequence length return to the same length as the original length and is applied after the pooling stage has completed. "Stitch" is done by combining the two sequences from the "Shift" stage, so the resolution of the sequence becomes 700 × f. The dense layer then processes the "shift-and-stitch" sequences, followed by outputting the predicted result by the output layer. In this study, the Multilayer Shift-and-Stitch CNN technique [4] was implemented, fine-tuned and ran using Keras [21]. Keras library was chosen as its widely used as a deep learning tool and its ability to run on GPU.

4.3 Support vector machine

Support Vector Machine (SVM) [22] is one of the classifications in the machine learning fields. The main idea of SVM is to find a hyperplane that is used as a decision surface so the gap between one class and another class is maximized. In general, SVM works by following these steps: firstly, the features of the data are mapped into high-dimensional space using kernel functions. Secondly, the data is separated using the best hyperplane. The best hyperplane is obtained by maximizing the distance to the

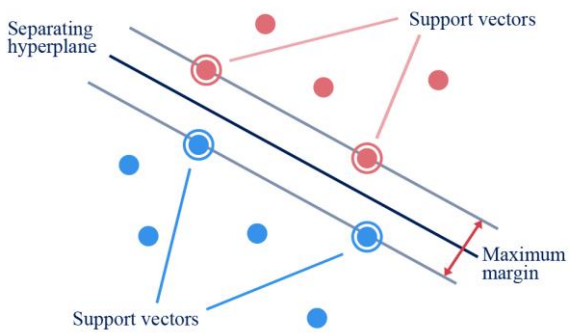

Figure. 4 Separating hyperplane with maximum margin

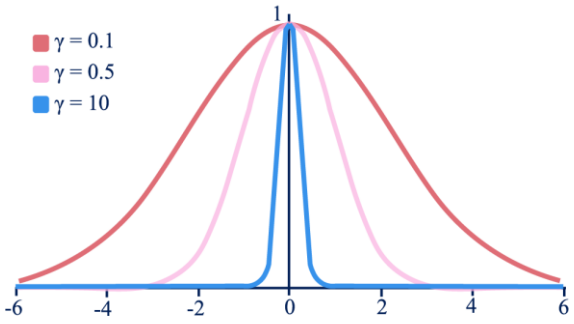

Figure. 5 Effects of γ value on gaussian curve

hyperplane between each closest sample (Supporting Vector) from each class (see Fig. 4).

A regularisation parameter *C* is used in SVM to find the best margin from the hyperplane. A proper value of *C* has to be searched so that the SVM finds the balance between maximal margin and minimal misclassification. A high value of *C* indicates that the model will receive more penalty when the model failed to classify data appropriately. On the contrary low value of *C* indicates that the model will tolerate few misclassified data. The kernel used in this study is the Radial Basis Function (RBF). RBF was chosen because of its to classify data on datasets that have many features or have high dimensions. Apart from the *C* parameter RBF's prediction results are also influenced by the γ parameter as shown in Eq. (4).

$$K(X, X_i) = \exp(-\gamma \|X - X_i\|^2) \quad (4)$$

Fig. 5 shows how the value of γ from Eq. (4) affects the gaussian curve. A high value of γ will be resulting in more curvature on the decision boundaries. Otherwise, a small value of γ will be resulting in smoother decision boundaries. The exact value of *C* and γ have to be found so that the SVM can generalize the data properly.

In this study, SVM was used to predict the secondary structure of proteins by using transformed feature maps from CNN's last shift-and-stitch layer as the input. *f* number of features from every index of

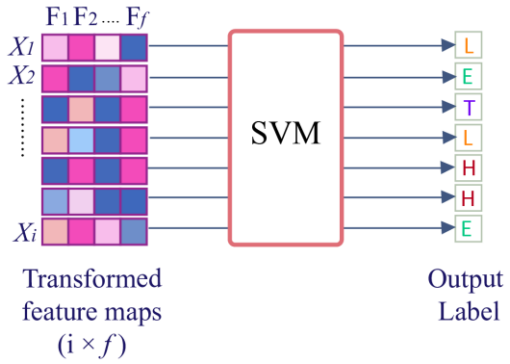

Figure. 6 SVM trained on transformed feature maps from CNN and gives label as an output

each sequence was fed into the SVM and had its label predicted by the model, as shown in Fig. 6. We chose ThunderSVM [23] as we are dealing with data with large features and large in quantities.

4.4 Hybrid CNN-SVM

This study proposed a hybrid of CNN and SVM to predict the secondary structure of the protein. The hybrid architecture used in this research was designed by combining 1-Dimensional CNN with SVM is illustrated by Fig. 7. Multilayer Shift-and-Stitch CNN [4] was chosen as it is able to extract and enrich the relationship patterns of the protein primary structures sequences, and also keeping the data resolution. We fine-tuned the CNN and then modified it, so instead of giving orthogonal label as an output, the models produce feature maps. By doing this step, we project the data into higher dimensional spaces. We aimed to exploit the high dimensional space by using SVM as the classifier. Thus, the feature maps produced by the modified CNN had to be transformed so it can be used for training by the SVM.

The modified CNN model produced 3-dimensional array feature maps as an output, and therefore, a transformation must be applied so the feature maps can be used by the SVM model (Fig. 8). For example, sequences of feature-maps with a size of 5,365 × 700 × *f* have to be stacked in the first place, making it one long sequence with a size of 3,755,500 × 512. Furthermore, zero-padding is then removed from the sequence as the SVM does not need to predict any zero-paddings. The transformation was done so that the SVM could train and test the data seamlessly. The transformed feature maps were then used as an input for the SVM models. SVM was chosen as a replacement of the dense layers as it has shown several good performances when it comes to predicting data with a high dimension of features [6-7]. Moreover, some studies have shown that SVM is as powerful as dense layers when it comes to

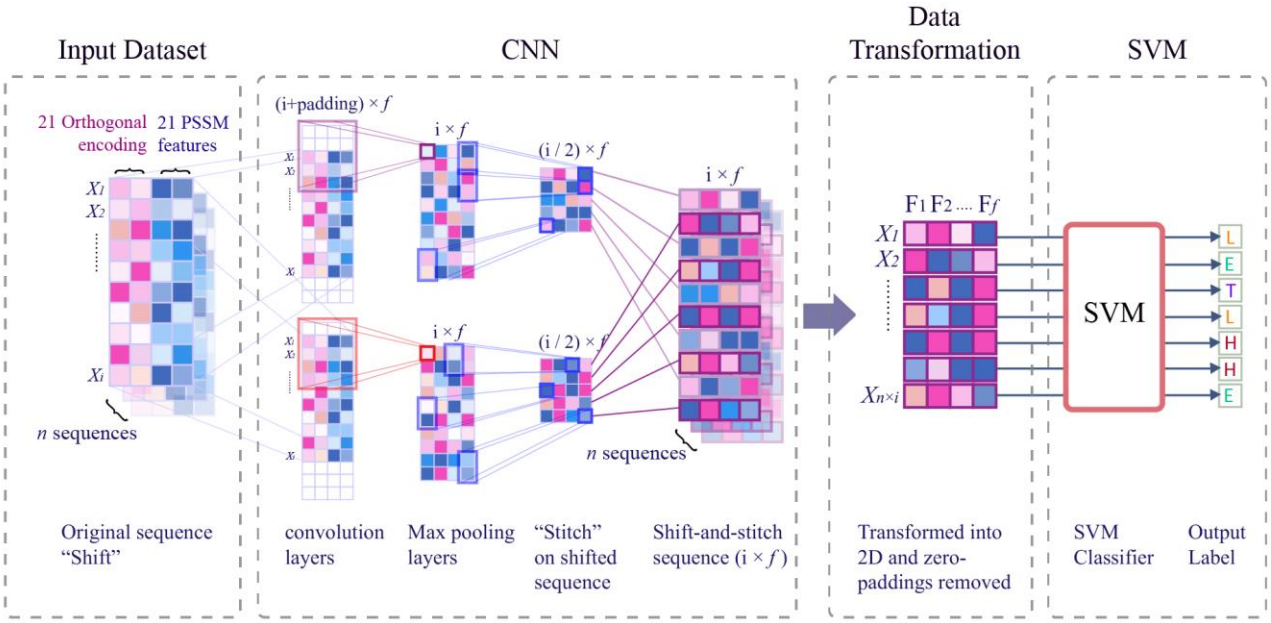

Figure. 7 Detailed architecture of Hybrid CNN-SVM

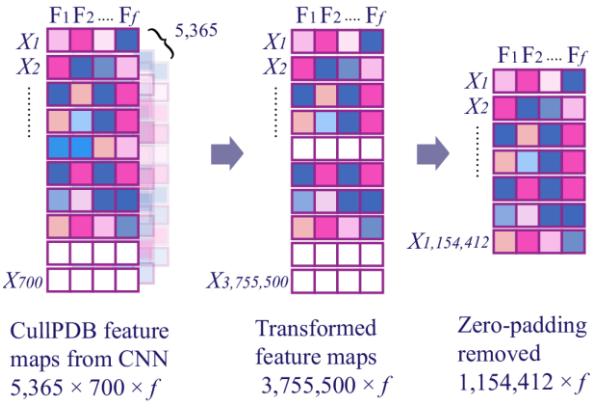

Figure. 8 Feature maps from CNN being transformed and cleaned from zero-padding

recognizing patterns [22,24]. Thus, instead of using dense layers in CNN, we are using SVM as a substitute. We then fine-tuned, trained, and tested the models. The SVM will produce a secondary structure class output, which will then be evaluated using the accuracy methods Q3 and Q8. The Q3 and Q8 accuracy methods are the accuracies obtained by calculating the number of correct predictions divided by the total number of test data.

5. Results

There are several stages conducted in this study: searching for optimal parameter values for the CNN, modifying the CNN model so it can produce input data for the SVM, and searching for optimal parameter values for the SVM. We decided to determine some unchangeable parameters value for the CNN, namely pooling size, the number of dense layers, dropout rate, activation function, and dense layers hidden units (see Table 3).

We then fine-tuned the models by looking for the best combination of 3 parameters, which are convolution layers, convolution size, and feature maps size, with their value tabulated in Table 4. We provided the detailed results of the CNN models in Table 5. Table 5 shows that increasing the number of feature maps did increase the accuracy score for some models, but at the same time did not increase the accuracy score for the rest.

Thus, we conclude that the number of feature maps does not give significant effects in this study. The convolution size also seems doesn't have any impact on increasing the accuracy score. Each model with the same number of convolution layers and feature maps show a no increasing nor decreasing trend. However, increasing the number of convolution layers does increase the accuracy for the

Table 3. CNN's fixed parameters

Parameter	Value
Dense Layers	1
Dense Layer Neurons	512
Input Dropout	0.2
Dropout	0.5
Activation Function	{ ReLU, Softmax }

Table 4. CNN's fine-tune parameters

Parameter	Value
Convolution Layers	{ 2, 3, 4}
Convolution Size	{ 6, 7, 8, 9, 10 }
Feature maps	{ 512, 1024}

Table 5. CNN's performance measures

Convolution Layers	Convolution Size	Feature map	Accuracy (%)
2	6	512	67.701
2	7	512	67.509
2	8	512	67.990
2	9	512	68.012
2	10	512	68.097
3	6	512	68.521
3	7	512	68.329
3	8	512	68.387
3	9	512	68.442
3	10	512	68.291
4	6	512	68.467
4	**7**	**512**	**68.711**
4	8	512	68.535
4	9	512	68.594
4	10	512	68.344
2	6	1024	67.918
2	7	1024	67.799
2	8	1024	67.731
2	9	1024	67.890
2	10	1024	67.750
3	6	1024	68.445
3	7	1024	68.266
3	8	1024	68.450
3	9	1024	68.468
3	10	1024	68.341
4	6	1024	68.486
4	7	1024	68.492
4	8	1024	68.434
4	9	1024	68.223
4	10	1024	68.174

majority of models with the same convolution size and number of feature maps. The results show that the highest accuracy was achieved when the number of the convolution layers, convolution size, and feature maps were 4, 7, and 512 respectively.

Afterwards, we modified the best model by removing the dense layer, resulting in the model giving 512 feature maps as an output. Note that by doing this, we increase the number of features up to 12 times (from 42 features into 512 features). We use SVM capabilities to exploit these high dimensional features, but firstly the feature maps produced by the modified CNN need to be transformed so it can be used by the SVM. The CullPDB data produced from the modified model.

Had a size of 5,365 × 700 × 512, which then transformed into a 2-dimensional matrix with the size of 1,154,412 × 512. We also applied the same steps for CB513 dataset, which had a size of 514 × 700 × 512 after being transformed by the modified CNN model. We then transformed the CB513 data into a 2-dimensional matrix, resulting in the data had a size of 84,765 × 512. Notice that it is not 3,755,500 × 512 (CullPDB) and 359,800 × 512 (CB513) as we removed the zero-padding from both datasets. Labels of both datasets were also transformed by the same

Table 6. Measurements of fine-tuned CNN

Label	Total Label	True Positive	Precision	Recall
H	26,157	24,045	0.8378	0.9193
E	18,016	14,525	0.7574	0.8062
L	17,920	11,576	0.5527	0.646
T	10,013	5,325	0.53	0.5318
S	8,316	1,909	0.4792	0.2296
G	3,132	834	0.454	0.2663
B	1,181	29	0.3867	0.0246
I	30	0	0	0
Total	**84,765**	**58,243**		

Table 7. SVM's fine-tune parameters

Parameter	Value
C	{ 0.01, 0.1, 1, 10 }
γ	{ 0.001, 0.01, 0.1, 1

Table 8. SVM's fine-tune parameters

Parameters		Accuracy (%)
C	γ	
0.01	0.001	0.035
0.01	0.01	57.048
0.01	0.1	68.353
0.01	1	67.549
0.1	0.001	55.826
0.1	0.01	68.299
0.1	0.1	68.548
0.1	1	68.626
1	0.001	68.284
1	0.01	68.48
1	0.1	68.636
1	0.5	68.689
1	0.75	68.682
1	1	68.721
1	**1.25**	**68.735**
1	1.5	68.734
10	0.001	68.46
10	0.01	68.527
10	0.1	68.665
10	1	68.085

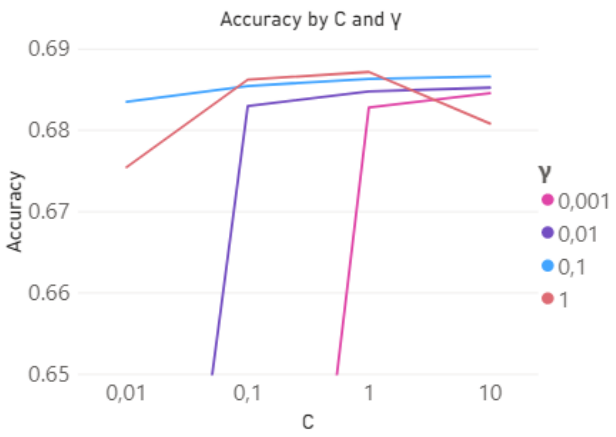

Figure. 9 Line chart of accuracy by C and γ

Table 9. Measurements of fine-tuned CNN-SVM hybrid

Label	Total Label	True Positive	Precision	Recall
H	26,157	23,739	0.8534	0.9076
E	18,016	14,412	0.7629	0.8
L	17,920	11,845	0.5480	0.661
T	10,013	5,339	0.5291	0.5332
S	8,316	1,986	0.4907	0.2388
G	3,132	896	0.4075	0.2861
B	1,181	46	0.4259	0.039
I	30	0	0	0
Total	**84,765**	**58,263**		

steps with the addition of argmax operation, resulting in a 2-dimensional matrix with the size of 1,154,412 × 1 for the CullPDB dataset and 84,765 × 1 for the CB513 dataset. Fig. 8 illustrates how the transformation was done with CullPDB dataset as an example.

We selected the Radial Basis Function as the kernel for SVM. We also conduct a kind of grid-search, aiming to find the best combination of several values of *C* and γ as tabulated in Table 7. From Fig. 9, we noticed an increasing trend in accordance with increasing values of C and γ. We also saw that when $C = 1$ and $\gamma = 1$ the model managed to outmatch the other models. Thus, we fine-tuned the SVM for the second time, this time using 1 as the value of *C* and the values of γ were set to be close at 1 {0.5, 0.75, 1.25, 1.5} (see Table 8). The model reached its top performance when the value of $C = 1$ and the value of $\gamma = 1.25$. The detailed performance of the best model is tabulated in Table 9.

6. Discussion

We have introduced a combination of CNN and SVM for predicting the secondary structure of the protein. The CNN architecture was inspired by MUST-CNN [4] and, in this study, we tried to maximize the potencies of the model by fine-tuning, increasing the dimension of the features, and replacing the dense layers with SVM.

The best CNN model encountered common problems when classifying imbalanced data, which was failing to classify labels with low occurrences. As tabulated in Table 6, the model failed to classify several minor labels as shown by low recall score for label S, G, B, and I. We believe that the CNN's convolution layers only managed to capture the features of the labels with high occurrences while ignoring labels with lower occurrences. We tried to tackle this issue by doing class weighting and random sampling, yet the model still failed to capture the minor labels' signature features. There is also a possibility where the minor classes don't have a signature feature that differs them from the other classes. We also suspect that the missing of low-level feature information from lower layers might be the cause of the CNN model failing to classify minor classes in higher layers. In the other words, our model might be able to capture long-range interdependencies but failed to retrieve local context. The use of SVM instead of dense layers surprisingly increased the Q8 accuracy, up to 0.024% higher than the fine-tuned CNN model (see Table 10). However, the SVM suffered the same problem as CNN, which is failing to classify minor labels (see Table 9). Once again, we tried to weight the training data, hoping to tackle the imbalanced data issue but ended up failing to get a model with high accuracy. We believe our SVM also suffers the same issue because it used feature maps from the CNN model as an input, which mostly contains features from labels with high occurrences.

Furthermore, our models obtained higher Q8 accuracy compared to various models [4-6,16-17]. Our models managed to outmatch previous related study that used MUST-CNN architecture [4]. The study has the most similarity with our study's CNN model. However, our non-hybrid CNN model scored 0.311% higher Q8 score than the previous study by scoring Q8 68.711% in CB513 dataset (see Table 10). We suspect that it is happened because of by using a different architecture on the same technique will result in different prediction capabilities. Consequently, our hybrid model scored higher Q8 score compared to a previous study [4], scoring 68.735% in Q8 accuracy or about 0.355% higher (roughly 283 true positives labels differences with the previous study). Thus, proof that the SVM is capable of increasing the classifier's ability to predict.

Our proposed architecture scored 0.435% higher in Q8 accuracy to a study that used Deep-CNF [5].

We suspect that this previous study [5] has head-to-head capabilities compared to MUST-CNN study [4], while our model has a slightly better performance when compared to them. Moreover, our model scored higher Q8 accuracy score than Multi-scale CNN [6]. Our model's CNN architecture differs from this Multi-scale CNN study in terms of the existence of highway connector between each convolutional layer. We realised that it might be possible for our model to score higher by applying a connecting-highway technique as used by the previous study [6].

Our hybrid model outperforms DeepSeqVec and DeepProf+SeqVec model [16] by 2.735% and 6.235%. We notice that our study used the same feature maps size (1024) as theirs and got similar Q8 accuracy results. However, they showed an increase up to 3.8% by combining embedding with several different features. We suspect that the previous study [16] produced a different result compared to our study purely because of the different implementation of CNN architecture (as we were using the MUST-CNN technique and previous study use their own CNN architecture, proofing that the MUST-CNN is a reliable technique to be used in sequence labelling task). Finally, we also outperform a study that implemented Bi-RNN Single Model as their architecture [17]. This architecture was way more complex compared to ours, while our architecture still managed to score a higher Q8 accuracy score, beating theirs by 0.235%.

Table 10. Comparison of different models' performance on CB513 dataset (Q8)

Model	Q8 (%)
DeepSeqVec [16]	62.5 ± 0.6
DeepProf+SeqVec [16]	66.0 ± 0.5
Deep-CNF [5]	68.3
Multi-scale CNN One-Hot Encoded [6]	68.3
MUST-CNN [4]	68.4
Bi-RNN Single Model [17]	68.5
Fine-tuned CNN (Ours)	**68.711**
Fine-tuned CNN-SVM (Ours)	**68.735**

Table 11. Comparison of different models' performance on CB513 dataset (Q3)

Model	Q3 (%)
DeepSeqVec [16]	76.9 ± 0.5
SVM-GA [6]	76.11
SVM-SF [7]	78.0
DeepProf + SeqVec [16]	80.7 ± 0.5
Fine-tuned CNN-SVM (Ours)	**81.49**

We also converted the prediction results into 3 class of secondary structure of the protein and compared it with several related studies as tabulated in Table 11. Our model performs better than studies that used SVM [6-7] and study that used sequence embedding [16], scoring Q3 accuracy of 81.49% for CNN-SVM hybrid model. Our model performed better than studies that used SVM [6-7]. These studies used the sliding window as an input receiver for the SVM, which limits the models to capture relation of sequences' distant features. Our model didn't suffer the problem, and we suspect that it has happened because the convolution process in our model managed to capture the sequences' distant features. We notice that the limitation of both previous studies in capturing distant relation cause our model to outperform theirs. The Previous study [16] also converted its Q8 accuracy result into Q3 accuracy, with our model having higher accuracies than theirs (in line with Q8 accuracy).

7. Conclusion

We have proposed and demonstrated the possibility of combining CNN and SVM to do a sequence labelling task, specifically to predict the secondary structure of the protein. We used CB513 dataset in this study. Orthogonal encoding and Position Specific Scoring Matrix (PSSM) were used as features. We used Shift-and-Stitch technique in the CNN to tackle the resolution problems that occur because of convolution and pooling stages. We then modified the CNN by removing its dense layers, resulting in the CNN producing feature maps of the dataset. This step was meant to transform the data into higher dimensional space and enrich the features. Afterwards, we trained the SVM with the feature maps produced from previous steps. The SVM was used as it is capable to classify data with high dimensional features. We fine-tuned both CNN and SVM parameters to find the best combinations that produce the best accuracy. We showed that our CNN managed to capture long-range interdependencies between each residue in sequences. Using SVM as an alternative of dense layers in classifying high dimensional data enables the model to achieve a higher accuracy score by up to 0.024% in Q8 accuracy. Our hybrid model achieved Q3 accuracy of 81.49% Q8 accuracy of 68.735% on CB513 dataset. In future works, it is possible to add more features like relative and absolute solvent accessibility.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization, Afiahayati; methodology, Vincent Michael Sutanto, Zaki Indra Sukma, and Afiahayati; software, Vincent Michael Sutanto, Zaki Indra Sukma, and Afiahayati; validation, Vincent Michael Sutanto, Afiahayati; formal analysis, Vincent Michael Sutanto, Afiahayati; writing - original draft preparation, Vincent Michael Sutanto; writing - review and editing, Afiahayati; supervision, Afiahayati.

References

[1] M. Zamani and S. C. Kremer, "Protein secondary structure prediction through a novel framework of secondary structure transition sites and new encoding schemes", In: *Proc. of IEEE International Conf. on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB 2016)*, Chiang Mai, Thailand, pp. 1-7, 2016.

[2] T. L. Blundell, S. Bedarkar, E. Rinderknecht, and R. E. Humbel, "Insulin-like growth factor: a model for tertiary structure accounting for immunoreactivity and receptor binding", In: *Proc. of the National Academy of Sciences of the United States of America*, Vol. 75, No. 1, pp.180-184, 1978.

[3] J. Zhou and O. G. Troyanskaya, "Deep Supervised and Convolutional Generative Stochastic Network for Protein Secondary Structure Prediction", In: *Proc. of the 31st International Conf. on Machine Learning (ICML 2014)*, Bejing, China, pp. 745-753, 2014.

[4] Z. Lin, J. Lanchantin, and Y. Qi, "MUST-CNN: A Multilayer Shift-and-Stitch Deep Convolutional Architecture for Sequence-based Protein Structure Prediction", In: *Proc. of the Thirtieth AAAI Conf. on Artificial Intelligence (AAAI-16)*, Phoenix, Arizona, pp. 27-34, 2016.

[5] S. Wang, J. Peng, J. Ma, and J. Xu, "Protein Secondary Structure Prediction Using Deep Convolutional Neural Fields", *Scientific Reports*, Vol. 6, No. 18962, 2016.

[6] J. Zhou, H. Wang, Z. Zhao, R. Xu, and Q. Lu, "CNNH_PSS: protein 8-class secondary structure prediction by convolutional neural network with highway", *BMC Bioinformatics*, Vol. 19, No. 60, 2018.

[7] Y. Wang, J. Cheng, Y. Liu, and Y. Chen, "Prediction of protein secondary structure using support vector machine with PSSM profiles", In: *Proc. of 2016 IEEE Information Technology, Networking, Electronic and Automation Control Conf. (ITNEC)*, Chongqing, China, pp. 502-505, 2016.

[8] Y. Chen, Y. Liu, J. Cheng, and Y. Wang, "Prediction of protein secondary structure using SVM-PSSM Classifier combined by sequence features", In: *Proc. of 2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conf. (IMCEC)*, Xi'an, China, pp. 103-106, 2016.

[9] Afiahayati, E. Anarossi, R. D. Yanuaryska, F. U. Nuha, S. Mulyana, "Comet Assay Classification for Buccal Mucosa's DNA Damage Measurement with Super Tiny Dataset Using Transfer Learning", *Intelligent Information and Database Systems: Recent Developments, Studies in Computational Intelligence*, Vol. 830, pp. 279-289, 2020.

[10] D. U. N. Qomariah, H. Tjandrasa, and C. Fatichah, "Classification of Diabetic Retinopathy and Normal Retinal Images using CNN and SVM", In: *Proc. of 2019 12th International Conf. on Information & Communication Technology and System (ICTS)*, Surabaya, Indonesia, pp. 152-157, 2019.

[11] T. Okamoto, T. Koide, S. Yoshida, H. Mieno, H. Toishi, T. Sugawara, M. Tsuji, M. Odagawa, N. Tamba, T. Tamaki, B. Raytchev, K. Kaneda, and S. Tanaka, "Implementation of Computer-Aided Diagnosis System on Customizable DSP Core for Colorectal Endoscopic Images with CNN Features and SVM", In: *Proc. of TENCON 2018 - 2018 IEEE Region 10 Conf.*, Jeju, Korea (South), pp. 1663-1666, 2018.

[12] Z. Wang and Z. Qu, "Research on Web text classification algorithm based on improved CNN and SVM", In: *Proc. of 2017 IEEE 17th International Conf. on Communication Technology (ICCT)*, Chengdu, China, pp. 1958-1961, 2017.

[13] Y. Chen and Z. Zhang, "Research on text sentiment analysis based on CNNs and SVM", In: *Proc. of 2018 13th IEEE Conf. on Industrial Electronics and Applications (ICIEA)*, Wuhan, China, pp. 2731-2734, 2018.

[14] V. M. Sutanto, Prediksi Struktur Sekunder Protein Menggunakan Convolutional Neural Network dan Support Vector Machine (in English: Protein Secondary Structure Prediction using Convolutional Neural Network and Support Vector Machine), Bachelor Thesis, Universitas Gadjah Mada, Indonesia, 2020.

[15] Z. I. Sukma, Prediksi Struktur Sekunder Protein Menggunakan Convolutional Neural Network (in English: Protein Secondary Structure Prediction using Convolutional Neural

Network), Bachelor Thesis, Universitas Gadjah Mada, Indonesia, 2017.
[16] M. Heinzinger, A. Elnaggar, Y. Wang, C. Dallago, D. Nechaev, F. Matthes, and B. Rost, "Modeling aspects of the language of life through transfer-learning protein sequences", *BMC Bioinformatics*, Vol. 20, No. 723, 2019.
[17] A. R. Johansen, C. K. Sønderby, S. K. Sønderby, and O. Winther, "Deep Recurrent Conditional Random Field Network for Protein Secondary Prediction", In: *Proc. of the 8th ACM International Conf. on Bioinformatics, Computational Biology, and Health Informatics*, Boston, Massachusetts, pp. 73-78, 2017.
[18] J. A. Cuff and G. J. Barton, "Application of multiple sequence alignment profiles to improve protein secondary structure prediction", *Proteins*, Vol. 40, No. 3, pp.502-511, 2000.
[19] Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, 2016.
[20] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks", In: *Proc. of 2nd International Conf. on Learning Representations (ICLR 2014)*, Banff, Canada, 2014.
[21] F. Chollet, *Keras*, 2015.
[22] C. Cortes and V. Vapnik, "Support-vector networks", *Mach Learn*, Vol. 20, pp. 273-297, 1995.
[23] Z. Wen, J. Shi, Q. Li, B. He, and J. Chen, "ThunderSVM: a fast SVM library on GPUs and CPUs", *The Journal of Machine Learning Research*, Vol. 19, No. 1, pp. 1-5, 2018.
[24] B. Schölkopf, C. Burges, and V. Vapnik, "Extracting Support Data for a Given Task", In: *Proc. of the First International Conference on Knowledge Discovery and Data Mining*, Menlo Park, California, pp. 252-257, 1995.