SAINT: self-attention augmented inception inside inception network improves protein secondary structure prediction

Mostofa Rafid Uddin[1,2,*], **Sazan Mahbub**[1,*], **M. Saifur Rahman**[1], and **Md Shamsuzzoha Bayzid**[1,†]

[1]Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka 1205, Bangladesh
[2]Department of Computer Science and Engineering, East West University, Dhaka 1212, Bangladesh

Received on December 7, 2019; revised on May 10, 2020; editorial decision on May 12, 2020; accepted on May 16, 2020

Abstract

Motivation: Protein structures provide basic insight into how they can interact with other proteins, their functions and biological roles in an organism. Experimental methods (e.g. X-ray crystallography and nuclear magnetic resonance spectroscopy) for predicting the secondary structure (SS) of proteins are very expensive and time-consuming. Therefore, developing efficient computational approaches for predicting the SS of protein is of utmost importance. Advances in developing highly accurate SS prediction methods have mostly been focused on 3-class (Q3) structure prediction. However, 8-class (Q8) resolution of SS contains more useful information and is much more challenging than the Q3 prediction.

Results: We present SAINT, a highly accurate method for Q8 structure prediction, which incorporates self-attention mechanism (a concept from natural language processing) with the Deep Inception-Inside-Inception network in order to effectively capture both the

*The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.

Associate Editor: Arne Elofsson

†To whom correspondence should be addressed.

Associate Editor: Arne Elofsson

short- and long-range interactions among the amino acid residues. SAINT offers a more interpretable framework than the typical black-box deep neural network methods. Through an extensive evaluation study, we report the performance of SAINT in comparison with the existing best methods on a collection of benchmark datasets, namely, TEST2016, TEST2018, CASP12 and CASP13. Our results suggest that self-attention mechanism improves the prediction accuracy and outperforms the existing best alternate methods. SAINT is the first of its kind and offers the best known Q8 accuracy . Thus, we believe SAINT represents a major step toward the accurate and reliable prediction of SSs of proteins.

Availibility and Implementation: SAINT is freely available as an open-source project at https://github.com/SAINTProtein/SAINT.

Contact: shams.bayzid@gmail.com

1 Introduction

Proteins are bio-molecules made of long chains of amino acid residues connected by peptide bonds. The functions of proteins are usually determined by their tertiary structure and for determining the tertiary structure and related properties, the secondary structure (SS) information is crucial. Protein structure can be experimentally determined by X-ray crystallography and multi-dimensional magnetic resonance in laboratory, but these methods are very costly and time consuming and are yet to be consistent with the proliferation of protein sequence data [10]. Thus, the proteins with known primary sequence continue to outnumber the proteins with experimentally determined SSs. The structural properties of a protein depend on its primary sequence [2], yet it remains as a difficult task to accurately determine the secondary and tertiary structures of proteins. Hence, the problem of predicting the structures of a protein — given its primary sequence — is crucially important and remains as one of the greatest challenges in computational biology.

SS — a conformation of the local structure of the polypeptide backbone — prediction dates back to the work of Pauling and Corey in 1951 [14]

2 Approach

2.1 Feature representation

SAINT takes a protein sequence feature vector $X = (x_1, x_2, x_3, ..., x_N)$ as input, where x_i is the vector corresponding to the ith residue, and it returns the protein structure label sequence vector $Y = (y_1, y_2, y_3, ..., y_N)$ as output, where y_i is the structure label (one of the eight possible states) of the ith residue. Similar to SPOT-1D-base and MUFOLD-SS, our base model contains 57 features from PSSM profiles, HHM profiles and physicochemical properties. To generate PSSM, PSI-BLAST [1]

was run against Uniref90 database [18] with inclusion threshold 0.001 and three iterations. The HHM profiles were generated using HHblits [15] using default parameters against uniprot20_2013_03 sequence database. HHblits also generates seven transition probabilities and three local alignment diversity values, which we used as features as well. Seven physicochemical properties of each amino acid [e.g. steric parameters (graph-shape index), polarizability, normalized van der Waals volume, hydrophobicity, isoelectric point, helix probability and sheet probability] were obtained from [13]. So, in our base model, the dimension of x_i is 57 as this is the concatenation of $x_hhm_i \in R^{d_{hhm}}(d_{hhm} = 30)$, $x_{pss}m_i \in R^{d_{pssm}}(d_{pssm} = 20)$ and $x_{physica}l_i \in R^{d_{physical}}(d_{physical} = 7)$ Additional features were generated by windowing the predicted contact information as was done in SPOT-1D. The contact maps were generated using SPOT-Contact [7] and were used as our features by varying window lengths (the number of preceding or succeeding residues whose pairwise contact information were extracted for a target residue). Our ensemble model constitutes of four different models, that we trained with varying input features: one without the contact maps (base model) and three with different window lengths (10, 20 and 50) of the contact-map-based features. The features were normalized to ensure 0 mean and SD of 1 in the training data, similar to SPOT-1D.

2.2 Architecture of SAINT

The architecture of SAINT can be split into three separate discussions:

i the architecture of our proposed self-attention module

ii the architecture of the existing inception module and the proposed

iii the overall pipeline of SAINT

2.2.1 Self-attention module

Attention mechanism implies paying attention to specific parts of input data or features while generating output sequence [3]. It calculates a probability distribution over the elements in the input sequence and then takes the weighted sum of those elements based on this probability distribution while generating outputs.
In self-attention mechanism [4], each vector in the input sequence is transformed into three vectors- *query, key* and *value*, by three different functions. Each of the output vectors is a weighted sum of the *value* vectors, where the weights are calculated based on the compatibility of the *query* vectors with the *key* vectors by a special function, called *compatibility* function (discussed later in this section).
The self-attention module, we designed and augmented with the Deep3I network [5] is inspired from the self attention module proposed by [19] and is depicted in Figure 1a. Our self-attention module takes two inputs:

i the features from the previous inception module or layer, $x \in R^{d_{protein} \times d_{feature}}$ and

ii position identifiers, $pos_i d \in R^{d_{protein}}$, where $d_{protein}$ is the length of the protein sequence and $d_{feature}$ is the length of the feature vector.

2.2.1.1 *Positional Encoding Sub-module.*

The objective of positional encodings is to inject some information about the relative or absolute positions of the residues in a protein sequence. The $PositionalEncodingPosEnc_\rho$ for a position ρ can be defined as follows [19]

$$PosEnc_{(\rho,2i)} = \sin(\frac{\rho}{1000^{\frac{2i}{d_{feature}}}}) \quad (1)$$

$$PosEnc_{(\rho,2i+1)} = \cos(\frac{\rho}{1000^{\frac{2i}{d_{feature}}}}) \quad (2)$$

where i is the dimension. We used such function as it may allow the model to easily learn to attend by relative positions since for any fixed offset κ, $PosEnc_{\rho+\kappa}$ can be represented as a linear function of $PosEnc_\rho$ [19]. For every position ρ, $PosEnc_\rho$ has the dimension $d_{protein} \times d_{feature}$. The output of positional encoding is added with the inputs x, resulting in new representations h [see Equation 3], which contain not only the information extracted by the former layers or modules, but also the information about individual positions.

$$h_{pos} = x_{pos} + PosEnc_{pos} \quad (3)$$

2.2.1.2 *Scaled dot-product attention sub-module.*

The input features in this sub-module, $h \in R^{d_{protein} \times d_{feature}}$ are first transformed into three feature spaces Q, K and V, representing *query, key* and *value*, respectively, in order to compute the scaled dot-product attention, where $Q(h) = W_Q h, K(h) = W_K h, V(h) = W_V h$ Here, W_Q, W_K, W_V are parameter matrices to be learned. Figue 1b shows a schematic diagram of this module.

Among various compatibility functions [e.g. scaled dot-product attention [19], addictive attention [3], similarity-attention [6], multiplicative-attention [12], biased general attention [17],etc.], we have chosen the scaled dot-product attention as it showed much promise in case of sequential data. [19] showed that in practice, the dot-product attention is much faster and space-efficient as it can be implemented using highly optimized matrix multiplication code, though theoretically both dot product and additive attention have similar complexity. Scaled dot product $s_{i,j}$ of two vectors h_i and h_j is calculated as shown in Equation 4

$$s_{i,j} = \frac{Q(h_i)K(h_i)^T}{\sqrt{d_\kappa}} \quad (4)$$

where d_κ is the dimension of the feature space K. The numerator of the equation, $Q(h_i)K(h_i)^T$ is the dot product between these two

vectors, resulting in the similarity between them in a specific vector space. Here, $\sqrt{d_\kappa}$ is the scaling factor, which ensures that the result of the dot product does not get prohibitively large for very long sequences.

The attention weights $e \in R^{d_{protein} \times d_{feature}}$ are calculated as shown in Equation 5, where $e_{j,i}$ represents how much attention have been given to the vector at position i while synthesizing the vector at position j.

$$e_{j,i} = \frac{exp(s_{i,j})}{\sum_{n=1}^{d_{protein}} e_{j,i} V(h_i)} \quad (5)$$

The attention distribution e is multiplied with the feature vectors $V(h)$ and then in order to reduce the internal covariate shift, this multiplicand is normalized using *batch normalization* [9], producing g, the output of the scaled dot-product attention sub-module, following the Equation 6.

$$g_j = BatchNorm(\sum_{n=1}^{d_{protein}} e_{j,i} V(h_i)) \quad (6)$$

Here, *BatchNorm(.)*) is the batch-normalization function and g_j is the j-th vector in the output sequence of this submodule. Finally, according to the Equation 7, g is multiplied by a scalar parameter α, the original input feature map x is multiplied by $(1 - \alpha)$ and these two multiplicands are summed to synthesize the final output y.

$$y_i = (\alpha) g_i + (1 - \alpha) x_i \quad (7)$$

where y_i is the ith output and α is a learnable scalar. By introducing weighed sum of g_i and x_i, we give our model the freedom to choose how much weight should be given to each of the features maps, g_i and x_i while generating the output y_i. The optimal value of the parameter α is learnt through back propagation along with the rest of the model.

3 Results and Discussion

3.1 Results on benchmark dataset

The comparison of SAINT with the state-of-the-art Q8 structure prediction methods on TEST2016, TEST2018, CASP13, CASP12 and CASP-FM is shown in Table 1. To train SAINT and tune necessary hyper-parameters, we have used the same training and validation sets that were used by SPOT-1D. Notably, SPOT- 1D is an ensemble of nine models where each single model uses predicted contact map in addition to other features. SAINT, on the other hand, is an ensemble of only four models, three of which take advantage of predicted contact map with different windows sizes. Experimental results show that SAINT outperforms all other methods across all the test sets. It is worth mentioning that SAINT's accuracy on the valida-

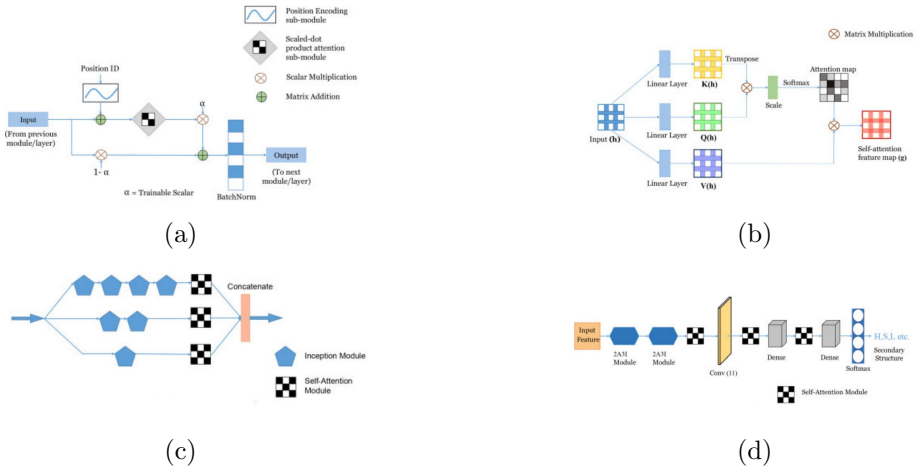

Figure 1: Schematic diagrams of SAINT and its various components. (a) Architecture of the self-attention module used in SAINT. (b) Architecture of the scaled dot-product attention sub-module. (c) Architecture of our proposed 2A3I module by augmenting self-attention within the 3I network. (d) A schematic diagram of the overall architecture of SAINT, which comprises two 2A3I modules, three self-attention modules, convolutional layers with window size 11 and 2 dense layers

tion set (78.18%) was also better than that of SPOT-1D (77.60%). SPOT-1D's base model,which does not require contact maps as features is also an ensemble of nine models,whereas SAINT- base model is a single model. Despite being a single model, SAINT-base consistently outperformed SPOT-1D base in TEST2016 and TEST2018. We could not evaluate SPOT-1D base on CASP12, CASP13 and CASP-FM as it is not publicly available. From Table 1, it is also evident that SAINT is substantially better than the other recent methods, namely, NetSurfP-2.0 and MUFOLD-SS. Even the base model of SAINT consistently outperformed both NetSurfP-2.0 and MUFOLD-SS. The remarkably large improvement of SAINT over MUFOLD-SS across all the dataset suggests the advantage of augmenting our proposed self-attention mechanism in the Deep3I network used in MUFOLD-SS. Statistical tests (see Table 1) suggests that these improvements of SAINT over other methods are statistically significant ($P << 0:05$).

Table 1: Statistical significance of the Q8 accuracy between SAINT and other state-of-the-art methods

Method	TEST2016 (1213)	TEST2018 (250)	CASP13 (31)	CASP12 (49)	CASP-FM (56)
SPOT-1D	$8.168e^{-27}$	$3.893e^{-5}$	0.101	0.0345	0.0791
NetSurfP-2.0	$2.607e^{-57}$	$3.258e^{-18}$	0.179	$1.55e^{-6}$	0.0001
MUFOLD-SS	$1.531e^{-88}$	$3.145e^{-21}$	0.179	$6.51e^{-5}$	0.005

Note: The numbers of protein chains or domains in these datasets are shown in parentheses. We show the P-values using a Wilcoxon signed-rank test

In addition to the model accuracy, we also investigated the *precision, recall* and F1-score to obtain better insights on the performances of various methods. Precision, also known as predictivity,denotes the confidence that can be imposed on a prediction. Recall signifies how accurately an algorithm can predict a sample from a particular class. Sometimes an algorithm tends to over-classify which results into high recall but low precision. On the other hand, some algorithms tend to under-classify, preserving the precision at the cost of recall. In order to get an unbiased evaluation of the performance, F1-score is considered to be an appropriate measure and has been being used for over 25 years in various domains [16] Tables 3–5 show the precision, recall and F1-score on each of the Q8 obtained by SAINT and other methods. These results suggest that SAINT achieves better F1-score than other methods on five states (out of Q8), showing that SAINT produced more balanced and meaningful results than other methods. SAINT substantially outperforms other methods on the non-ordinary states[20], such as I,G, S and T. However, MUFOLD-SS and SPOT-1D achieved slightly better F1-score for the 'B' and 'E' states, respectively. State 'I' (π helix) is extremely rare which comprises 7 or more residues and is present in 15% of all known protein structures [11]. They are very difficult to predict, but mostly found at functionally important regions, such as ligand- and ion-binding sites [11].Therefore, specialized predictors, such as PiPred [11], are also available that only predicts the π-helix structures. SAINT significantly outperforms SPOT-1D,NetSurfP-2.0 and MUFOLD-SS in predicting π-helix in TEST2016 dataset by correctly predicting 21 out of 47 'I' states and thus achieving a recall of 0.45 for this structure. SAINT's precision for πhelix, on the other hand, is 1. This is remarkable considering the fact that the π-helix specific predictor, PiPred, reports precision

and recall of 0.48 and 0.46, respectively, on a different dataset which they analyzed in [11]. However, this comparison should be taken with a grain of salt since the reported values are on different test sets.
We analyzed the CASP-FM dataset comprising the FM targets in the CASP dataset to demonstrate the performance of models on proteins with previously unseen folds. SAINT achieved the best accuracy on CASP-FM, suggesting SAINT's superiority in predicting structures of proteins having unseen folds.
While the advantage of utilizing our proposed self-attention mechanism in the Deep3I framework of MUFOLD-SS is evident from the significant improvement of SAINT over MUFOLD-SS across all the dataset analyzed in this study, we further investigated the efficacy of our proposed attention mechanism in capturing the long-range interactions. We computed the number of non-local interactions per residue for each of the 1213 proteins in TEST2016, and sorted them in an ascending order. Two residues at sequence position i and j are considered to have non-local interaction if they are at least 20 residues apart ($|i - j| \geq 20$), but $< 8A^{\circ}$ away in terms of their atomic distances between $C\alpha$ atoms [8]. Next, we put them in six equal sized bins $b_1, b_2, ..., b_6$ (each containing 202 proteins except for b_6, which contains 203 proteins), where b_1 contains the proteins with the lowest level of non-local interactions and b_6 represents the model condition with the highest level of non-local interactions. We show the Q8 accuracy of SAINT base and MUFOLD-SS on these model conditions in Table 6 and Figure 2. Note that, instead of our ensemble model, which is more accurate than our base model, we deliberately show the results for our single base model, which uses the same feature set as MUFOLD-SS, and the only difference between them is the self-attention modules introduced in our architecture.
These results show that the difference in predictive performance between SAINT-base and MUFOLD-SS significantly increases with increasing levels of non-local interactions. There is no statistically significant difference between them on b_1, but as we increase the level of non-local interactions, SAINT becomes significantly more accurate than MUFOLD-SS and attains the highest level of improvement on b_6. This clearly indicates that capturing non-local interactions by self-attention is the key factor in the improvement. We also performed the same analyses on other methods (see Figure 3). The results in Figure 3 show that the differences among of these methods are not that substantial on the model conditions with low levels of long-range interactions, but the differences become notable as we increase the non-local interactions. SAINT not only achieved the best accuracy, its improvement over other methods increases with increasing amount of long-range interactions as well—suggesting the superiority of our proposed self-attention mechanism compared to $CNN +$

$LSTM$ (used in NetSurfP-2.0) and $CNN(ResNet) + BRLSTM$ (used in SPOT- 1D) in terms of capturing the non-local interactions. In order to demonstrate the efficacy of SAINT and other methods in capturing the continuous structure of a protein, we show the 1D map of the annotated and predicted SS of a representative protein 5M2PA in TEST2016.

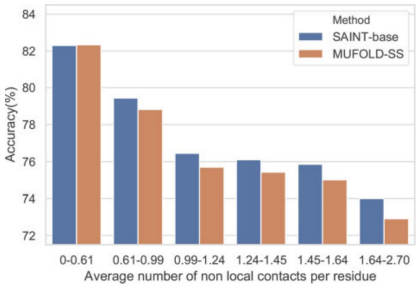

Figure 2: Accuracy of SAINT-base and MUFOLD-SS under various levels of non-local interactions. We show the results on the TEST2016 test set using six bins of proteins as shown in Table 6

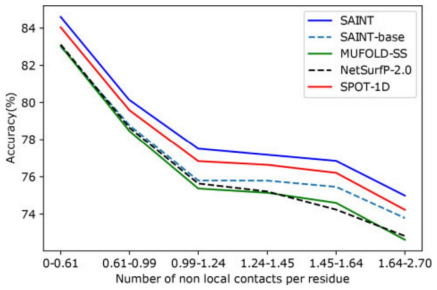

Figure 3: Accuracy of SAINT, SPOT-1D, NetSurfP-2.0 and MUFOLD-SS as a function of the average number of non-local interactions per residue. We show the results on the six bins as shown in Table 6

3.2 Running time

SAINT is much faster than the best alternate method SPOT-1D. For generating the structures of 1213 protein chains in TEST2016, given the necessary input files, SAINT took $360 \pm 5s$ whereas SPOT-1D took $2485 \pm 5s$ on our local machine [Intel core i7-7700 CPU (4 cores), 16 GB RAM, NVIDIA GeForce GTX 1070 GPU]. Under the same settings, SAINT took $197 \pm 5s$ to generate SSs for the 250 proteins in TEST2018, whereas SPOT-1D took $668 \pm 5s$. Since both these methods use the same input files for feature generation, this substantial difference in running time can be attributed to the efficiency of our attention based method over the LSTM network-based model used in SPOT-1D.

Acknowledgement

The authors thank the anonymous reviewers for their insightful comments and suggestions, and the authors of SPOT-1D for providing the PSSMs,HMMs and contact maps of the proteins in the training dataset.

References

[1] "Altschul,S.F. et al. (1997) Gapped BLAST and PSI BLAST: a new generation of protein database search programs. Nucleic Acids Res., 25, 3389–3402". In: (1997).

[2] "Anfinsen,C.B. (1973) Principles that govern the folding of protein chains.Science, 181, 223–230." In: (1973).

[3] "Bahdanau,D. et al. (2014) Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv: 1409.0473". In: (2014).

[4] "Cheng,J. et al. (2016) Long short-term memory- networks for machine reading.Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 551–561. Austin, Texas, USA." In: (2016).

[5] "Fang,C. et al. (2018) MUFOLD-SS: new deep inception- inside-inception networks for protein secondary structure prediction. Proteins, 86, 592–598". In: (2018).

[6] "Graves,A. et al. (2014) Neural turing machines. arXi preprint arXiv:1410.5401." In: (2014).

[7] "Hanson,J. et al. (2016) Improving protein disorder prediction by deep bidirectional long short-term memory recurrent neural networks. Bioinformatics, 33, 685–692." In: (2016).

[8] "Heffernan,R. et al. (2017) Capturing non-local interactions by long short-term memory bidirectional recurrent neural networks for improving prediction of protein secondary structure, backbone angles, contact numbers and solvent accessibility. Bioinformatics, 33, 2842 2849". In: (2017).

[9] "Ioffe,S. and Szegedy,C. (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd International Conference on International Conference on Machine Learning. Vol. 37. pp. 448–456. JMLR. Lille, France." In: (2015).

[10] "Jiang, Q. et al. (2017) Protein secondary structure prediction: a survey of the state of the art. J. Mol. Graph. Model., 76, 379–402." In: (2017).

[11] "Ludwiczak,J. et al. (2019) PiPred–a deep-learning method for prediction of π-helices in protein sequences. Sci. Rep., 9, 6888." In: (2019).

[12] "Luong,M.-T. et al. (2015) Effective approaches to attention-based neural machine translation. arXiv preprint arXiv: 1508.04025". In: (2015).

[13] “Meiler,J. et al. (2001) Generation and evaluation of dimension-reduced amino acid parameter representations by artificial neural networks. Mol. Model. Annual, 7, 360–369”. In: (2001).

[14] “Pauling,L. et al. (1951) The structure of proteins: two hydrogen-bonded helical configurations of the polypeptide chain. Proc. Natl. Acad. Sci. USA, 37,205–211.” In: (1951).

[15] “Remmert,M. et al. (2012) HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. Nat. Methods, 9, 173–175”. In: (2012).

[16] “Sasaki,Y. et al. (2007) The truth of the F-measure. Teach. Tutor. Mater., 1, 1–5”. In: (2007).

[17] “Sordoni,A. et al. (2016) Iterative alternating neural attention for machine reading. arXiv preprint arXiv: 1606.02245”. In: (2016).

[18] “UniProt Consortium (2007) The universal protein resource (UniProt). Nucleic Acids Res., 36, D190–D195”. In: (2007).

[19] “Vaswani,A. et al. (2017) Attention is all you need. In: Advances in Neural Information Processing System. pp. 5998–6008. Long Beach, California, USA”. In: (2017).

[20] “Wang,S. et al. (2016a) Protein secondary structure prediction using deep convolutional neural fields. Sci. Rep., 6, 18962”. In: (2016).