World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(REVIEW ARTICLE)

Check for updates

Protein secondary structure prediction using deep neural network and particle swarm optimization algorithms

Aka Patience Abimiku [1,*], Gilbert Imuetinyan Osaze Aimufua [1] and Muhammad Umar Abdullahi [2]

[1] *Department of Computer science, Nasarawa State University Keffi, Nigeria.*
[2] *Department of Computer Science, Federal University of Technology, Owerri, Nigeria.*

World Journal of Advanced Research and Reviews, 2022, 16(03), 219–230

Publication history: Received on 18 August 2022; revised on 30 November 2022; accepted on 03 December 2022

Article DOI: https://doi.org/10.30574/wjarr.2022.16.3.0862

Abstract

Protein secondary structure prediction from its amino acids is purposely used to evaluate and improve the accuracy of performance as well as drug design and cell functionality. Various approaches for predicting protein secondary structure have been used, each with varying accuracy, vulnerabilities, and strengths. In view of this, this paper is aimed at training a deep neural network with particle swarm optimization and comparing the results with the state of accuracy. Also, the methodology used is basic particle swarm optimization for training a 20-15-15-15-3 deep neural network. The Java programming language and the Spring Boot framework were employed to implement the various application programming interfaces of the model. The dataset acquired after the training of JPred Server 1.2, which included 1349 training sets and 149 test sets, was used in training the model. Following the training, it was discovered that the model had a highest accuracy of 53.18 percent on epoch 140, indicating that this model is not a best fit or an alternative to the current state of the art for the prediction of protein secondary structure.

Keyword: Protein; Deep Neural Network; Particle Swarm Optimization; Algorithms; Model; Drug Design

1. Introduction

Proteins carry out the majority of biological or chemical features in a cell, whereas the function of a protein is solely determined by its structure. (Beck & Alonsa, 2008) This could be the main reason why structural bioinformatics has become a vital area of research. Proteins are built from the twenty (20) naturally occurring amino acids (AA). These AA are small molecules composed of free amino groups (NH2) and free carboxyl groups (COOH) (Beck, 2008). These two groups are then linked to a central carbon (C) attached to hydrogen and a side chain group (R). These groups are illustratively shown in Figure 1.

As seen in Figure 1, what differentiate each amino acid is the sidechain R-group. Interestingly, out of the twenty naturally occurring amino acids, there are particular interest in two which are glycine and proline (Walsh.,2013).

Glycine is considered the smallest of the amino acids group, even though it has atom of hydrogen similar to that of R-group. It can adopt more flexible conformations that other amino acids cannot. Proline's R-group, on the other hand, has a cyclic bond with its backbone amino group. The cyclic nature of Proline makes it very rigid and unable to occupy many of the main chain conformation adopted by the other amino acids (Cormanich,2013).

* Corresponding author: Aka Patience Abimiku
Department of Computer science, Nasarawa State University Keffi, Nigeria.

These amino acids are actually gotten from the translation of Ribonucleic acids (RNA) which are gotten from the transcription of Deoxyribonucleic acids (DNA). The process of this formation is referred to as the central dogma of bioinformatics (Kumar and Khan, 2017). Figure 2 depicts the dogma of bioinformatics formation process.

Central Dogma: DNA → RNA → Protein

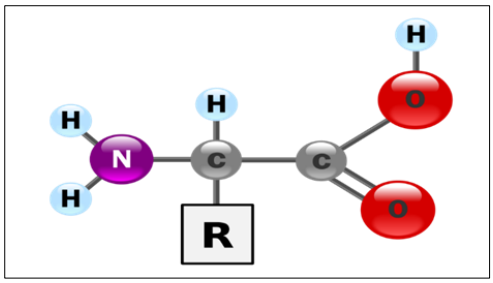
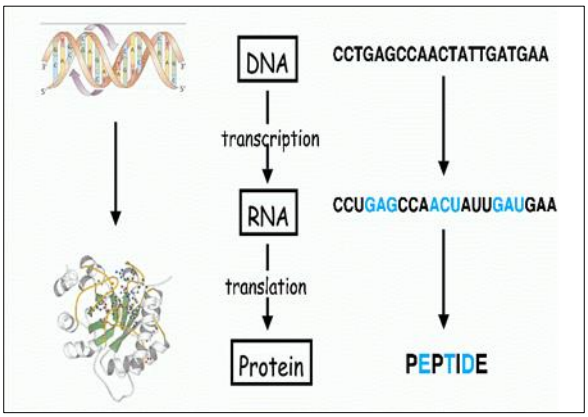


Figure 1 Amino Acid Structure (Bioinformatics, 2019a)



Source: Bioinfor, 2019a

Figure 1 Bioinformatics Central Dogma

The structure of proteins can be determined experimentally using various methods. Out of these various methods for determining a protein structure, there are two procedures that are notably used which are Nuclear Magnetic Resonance (NMR) Spectroscopy and X-ray Crystallography which are believed to be very effective (Cormanich et al., 2013). These methods are actually very expensive, time consuming and are not robust enough (Walsh et al.,2013). Thus, the issue of predicting and studying the secondary proteins structure from amino acids (Primary structure) in a more accurate and optimal way using deep neural network and particle swarm optimization appeared to be the point of interest of this paper. Thus, the objectives of this paper are to:

- Train Deep Neural Network with much data using position specific scoring matrix as the input
- Implement deep neural Network and Particles Swarm Optimization Algorithms using java programming Language.
- Compare the performance of the system with the existing model.

2. Literature survey

2.1. Protein Structure Prediction using Artificial Neural Network

This work employed three artificial neural networks. The first is used to classify the amino acid into their various BCD code, the second then classify them into their various primary structure while the third is used to classify the output of the second ANN into the secondary structure. Some problems, however, were observed due to the larger data sets which the ANN classifier was given to handle. It generates certain computational constrains. Moreover, the ANN classifier in its present form finds it hard to make differentiation between the amino acids in the sequences. Also, the classifier faced some difficulties while detecting the starting and ending point between two amino acids. These shall be removed in our

work and extend it for the prediction using PSO. This work claims to have 100% accuracy on training but neglected the importance of providing the test rating which this study seeks to address. (Bordoloi and Sarma, 2019).

2.2. Protein Secondary Structure Prediction Using Deep Multi-Scale Convolutional Neural Networks and Next-Step Conditioning

This work used the CullPDB and CB513 dataset, employed two models which are the Fully-Connected model and the Convolutional model. The Fully Connected model used a deep neural network of five feed forward layers. The accepted a window size of seventeen (17) for the input layer and used an error rate of 0.00004 which is reduced by 50% on every 35,000-training iteration by the ADAM's optimizer. The convolutional model is then built upon the Deep Neural Network to improve the fixed size window input. The current work contributes two state-of-the-art results to the eight-class secondary structure prediction problem. By analyzing the impact of convolutional variants developed for recognition of natural images, we develop a modified multi-scale and residual convolutional architecture which outperforms previous deep learning approaches on the same benchmark dataset, the idea of using conditioning on past structure labels to boost accuracy was introduced. While there is much future work to be done to exploit conditioning by mitigating the overfitting, pushing state-of-the-art Q8 accuracy on the CB513 data further by 0.9% relative to the highest previously reported result. anticipating that this approach will suffer less from overfitting induced by conditioning, since exact copying of consecutive output values will no longer yield good accuracy during training. This work predicted the 8-state protein structure with an accuracy of 71.4%. which is expected to rate more. (Busia, A., Collins, J., and Jaitly, N., 2016).

2.3. Protein Secondary Structure Prediction Using Deep Convolutional Neural Fields

This work used the DeepCNF in the prediction of both 3-state and 8-state protein secondary structure prediction. DeepCNF combined the advantages of both the Conditional Neural Fields (CNF) and the Deep Convolutional Neural Network (DCNN). The work used the five publicly available datasets from CullPDB, CB513, CASP10 and CASP11. This method was found to have outperformed other methods including PSIPRED. This method is said to have obtained an accuracy of 83.8% on 8-state and 71.9% on 3-state. Nonetheless, this method has a drawback on proteins with very sparse sequence profile. This makes it very challenging to predict from the primary sequence instead of sequence profile using this method. (Wang *et al*., 2016).

2.4. Cuckoo Search Algorithm and its Application for Secondary Protein Structure Prediction

This work investigated the use of cuckoo search algorithm which is a global optimization algorithm inspired by the behavior of the cuckoo species such as ani and guiro and the foraging patterns of animals. At the end of the experiment, they concluded that the Cuckoo search algorithm was able to solve the local convergence trapping problem experienced by other Evolutionary Algorithms in the prediction of protein structure but they never published the accuracy of the applied method. (Hoijat., 2016).

2.5. Protein Secondary Structure Using Long Short-Term Memory Networks

This work argued that the use of Neural Networks and Support Vector machines for the prediction of protein secondary structure is not idea due to the fact the primary structure of proteins cannot naturally be represented as a vector of fixed dimensionality. The work further proved that the sliding window is used to get around this problem yet the sliding window is not capable of learning the general dependency in the structure. However, the work used the long short-term memory networks without peepholes because it has been revealed in recent papers that the model works better without the peepholes. The result from the work is said to have a correct classification rate of 0.674 which is better than the 0.511 success rate in the Bi-direction Recurrent Neural Network used in the SSpro8. (Sonderby and Winther, 2014).

3. Methodology

Various computational methods such as HMM (Hidden Markov Model), ANN (Artificial Neural Network), SVM (Support Vector Machine), etc. have been used in the prediction of protein secondary structure problem. In this paper, the Particle Swarm Optimization Algorithm was used to train a Deep Neural Network for the prediction of the Protein Secondary Structure.

3.1. Principles of Deep Neural Network

Neural network has existed for a long time now but it's applications of recent has produced a high performance and state of art results in many fields. They are used in image recognition, natural language processing and in many more problems.

The neural network simulates the operations of the human brain. The human brain is primarily composed of nerve cells called neurons as shown in Figure 3. These neurons are connected via a strand of fiber called dendrites. These dendrites are used to accept information from other neurons and then pass out electrical spikes (information) through the axons.

The axon then splits into various branches which have a structure called synapse at the tip. These synapses convert the information from the axons into electrical effect that triggers or excites activity in the connected neuron.

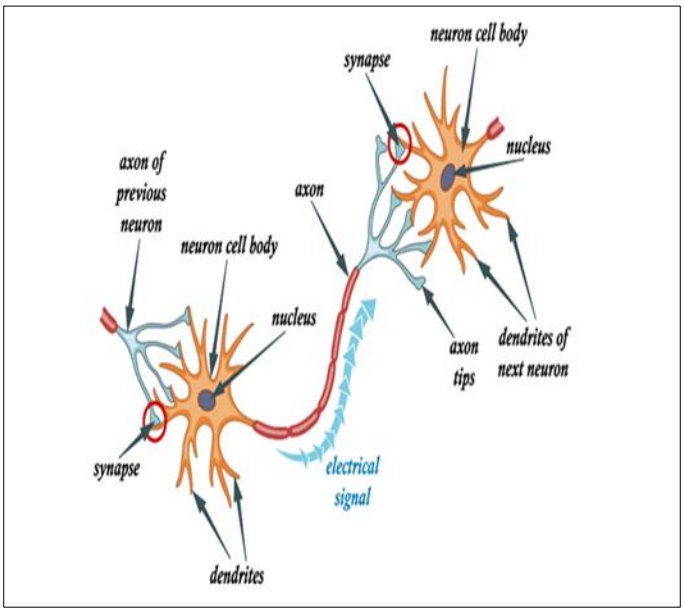


Figure 3 Human Brain Network (Yegnanarayana, 2009)

The concept of these neuron activities is emulated in the Artificial Neural Network (ANN). ANN is composed of various layers (depends on how it is designed) and these layers consist of neurons as illustrated in Figure 4. The neurons in this layer are connected to the next layer with set weights. These weights can be seen as signal strength that is used to calculate the value of the next neuron with the help of some mathematical functions called the activation function.

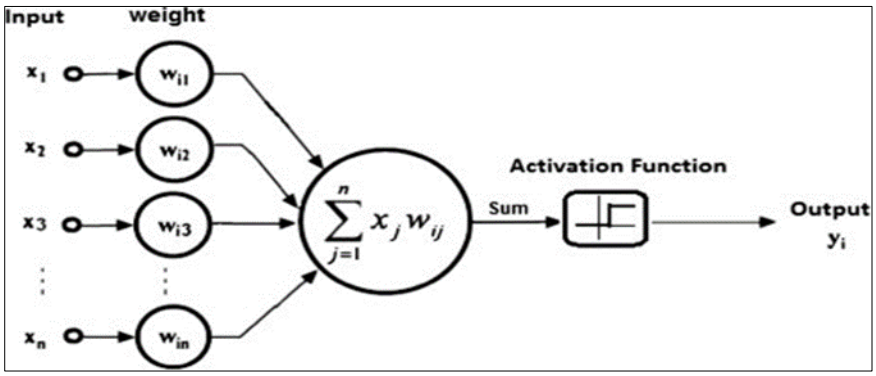


Figure 4 Artificial Neural Networks (Yegnanarayana, 2009)

In a multi-layered neural network, the output generated from the activation function is then used as the input to the next layer. There are various variants of ANN but the most popularly used is the feed-forward neural network. This neural network is composed of an input layer, zero or more hidden layer(s), and an output layer. A neural network with multiple hidden layers is referred to as a deep neural network. Figure 5 depicts a deep neural network.

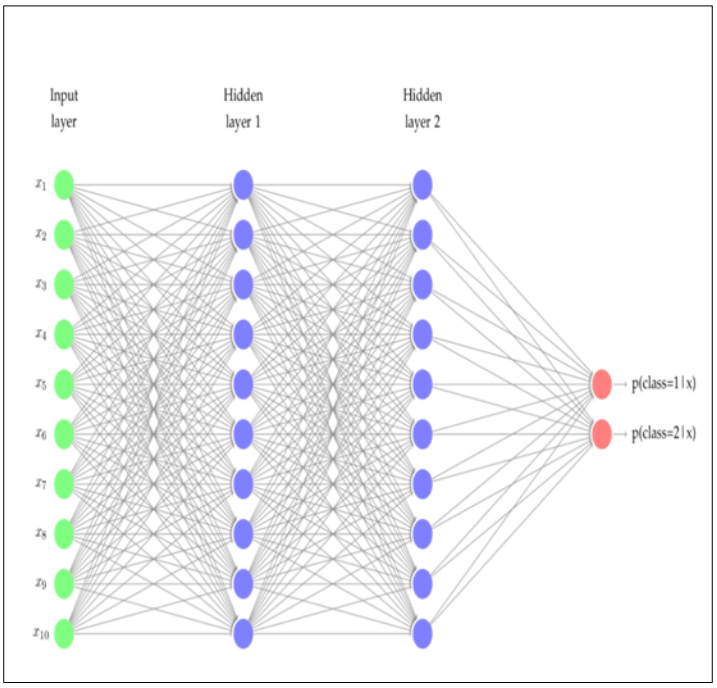


Figure 5 Deep Neural Network (Canziani., 2016)

Popularly, neural networks are trained with the stochastic Gradient Descent and Backward propagation.

3.2. Backward Propagation

Backward propagation is a supervised learning algorithm of artificial neural networks using the gradient descent. This algorithm is fully called "backward propagation of errors" because the error of the output is calculated using the weights with a given error function and then propagated back through the network layers. Based on the error derivative gotten, the weights of the network are then updated and the output regenerated.

The most popularly used error (loss) function used in this training is the Mean Squared Error (MSE). In the MSE, we have a vector of target value that we are training the model to learn and then a vector of outputs from the output layer. We calculate the error as follow:

$$MSE = \frac{1}{n}\sum(y(x) - t(x))^2$$

y(x): output value
t(x): target value

The derivative of this error function is then used in updating the weights of the network.

Source: https://www.guru99.com/

3.3. System Science Methodology

The system science methodology was adopted using the particle Swarm Optimization Algorithm to train deep neural network for the prediction of the protein secondary Structure. The methodology as shown in Figure 6 begins with the stage of data collection down to classification stage. Each square box represents a process while the arrows indicate movement from one process to another.

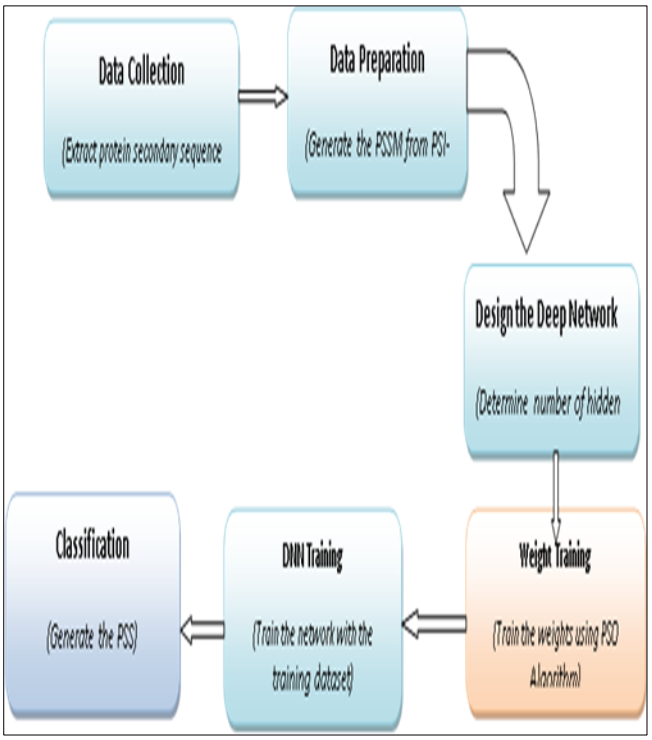


Figure 6 Training a deep neural network with particle swarm optimization algorithm System Model adopted and modified (Bredaet *al*. (2007)

3.4 System Model

The model used for this paper is shown in Figure 7.

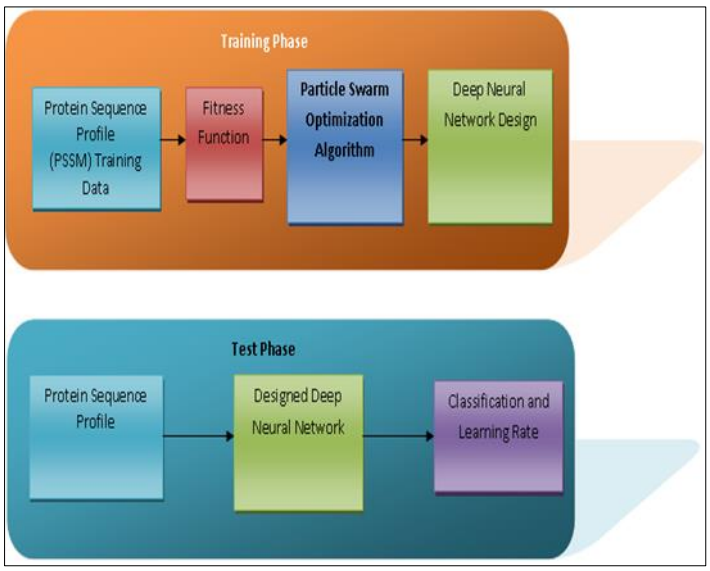


Figure 7 System Model System Model adopted and modified (Bredaet *al*. (2007)

The output of the network is coded as follows:

$Coil(-) = 0,0,1$

$Helix(H) = 0,1,0$

$Sheet(E) = 1,0,0$

The protein Sequence profile is a (n x 20) vector. Each row calculates the probability of the availability of a particular amino acid in the general twenty amino acids thereby the residue is coded to be a 20-dimensional vector which serves as the input to the deep neural network.

The layers of this network are then connected with a set of weight and bias and the output of each layer is then calculated using an activation function. The sigmoid function is adapted for this project which is defined as follows:

$$S(x) = \frac{1}{1 + e^{-x}}$$

Here, x is the sum output generated from the weights, bias and the value of the previous neuron given by:

$$x_n = \theta + \sum w_{kn} + o_k$$

Thereby,

$$S(x_n) = \frac{1}{1 + e^{-(\theta + \sum w_{kn} + o_k)}}$$

After these outputs are generated, a fitness function (Mean Squared Error) is used to calculate the deviation of the result from the target and then training begins. The smaller the error, the best the solution.

Particle swarm optimization (PSO)

The algorithm of PSO is a meta-heuristic and originally developed by an American social psychologist Kennedy (Kennedy and Eberhart, 2001). In this paper, one of the problems faced with a number of non -linear problems and, in order to find the correct solution, the PSO method has been developed and widely used. The PSO algorithm was stimulated to locate the best possible food route for bird and fish intelligence. Here, birds are the particles and try to find a solution to the problem. Particles are always tried to find out best possible solution to a problem through n -dimensional space, in which n represents each and every problem's different parameter [Kennedy et al., 2001].

Optimization of position and velocity is the basic principle of each particle.

Therefore, let us say,

$$xit = (xi1t, xi2t \ldots, xint) \; and \; vit = (vi1t, vi2t \ldots, vint$$

are the position and velocity of changing position designed for ith particle in tth iteration accordingly? The following equations are used for the ith particle's position and velocity in (t+1) th iteration.

$$vit{+}1 = \omega.vit + c1.r1.(pit - xit) + c2.r2.(git - xit) \; with - vmax \leq vit{+}1 \leq vmax \quad \ldots\ldots\ldots\ldots\ldots\ldots(1)$$

$$xit{+}1 = (xit + vit{+}1) \quad \ldots\ldots\ldots\ldots\ldots\ldots(2)$$

Where, xit represents previous ith position; pit represents most excellent found position; git represents particle's best position; $r1$ and $r2$ represent random numbers within 0 and 1; ω is weights of inertia; $c1$ is coefficient and $c2$ represents social coefficient Therefore, it is supposed to be believed that the concentration of all particle swarms in a point and space has been achieved when problem to be solved. The intelligence based PSO algorithm has been widely used in high efficiency swarm paralleling and optimization property. Beside this, by using multi-objective fitness function, PSO determines the quality of several features in a dataset.

For our model, the weights and bias of the system is randomly generated at first and the poured into the swarm by assigning this generated weights and bias to the global best position (solution) and the MSE of the random generated weights and bias as the global best error. The swarm then generates various particles (potential solutions) and then train and modifies the global best based on the number of epochs.

3.5 Algorithm

- Start
- Randomly assign weights(w) and bias(b) to the network
- for each neuron in the network
- computes the output o_i
- end for
- Calculate the Error (E) of the system with the present weights using Mean Squared Error
- Initialize the Global Best Position (Solution) of the Swarm to the weights and bias generated
- Initialize the Global Best Error of the Swarm to the Error Calculated
- Create a swarm with the appropriate number of particles
- Initialize the epoch
- Initialize each of the Swarm particle to a random state (position, velocity, error, personal best error, personal best position)
- Loop until done (until epoch exhausted)
  - For each particle in swarm,
    - ✓ Compute new particle velocity
    - ✓ Use new velocity to compute new position,
    - ✓ Compute error of new position
    - ✓ If new error better than personal best error, then
    - ✓ Set personal best error equals new error
    - ✓ If new error better than global best error
    - ✓ Set global best error equals to new error
    - ✓ Set global best position equals to new position
  - End for
- End loop
- Update the weights and bias with global best position
- Test the system with the test dataset
- End

4. Results and discussions

4.1. System Implementation

This model was implemented using Java Programming Language alongside Spring Boot framework. Due to the huge size of processing resources and computations required in this system, the system implements a console application that is been interfaced with web services (API). The initiative of the web services is to make this system available to other systems that may want to implement the algorithm. The API accepts a sequence profile and returns the predicted secondary structure of the protein. This system was implemented with three hidden layers and each hidden layer has a neuron size of 15. The System was also trained with 1349 protein sequences (281,421 residues) and the test was carried out with 149 (22,734 residues) protein sequences. This system has twenty neurons on the input layer to accommodate the twenty amino acids and three neurons in the output layer to code for the 3-state DSSP classification (Helix, Sheet and Coil). To calculate the accuracy of the system, we made a count of correct predictions and the wrong predictions and then applied the formula below:

$$percentage_{correct} = \frac{totalcorrectpredictions}{totalcorrectpredictions + totalwrongpredictions} * 100$$

4.2. Parameter Setting, Tweaking and Training

One of the crucial aspects of this prediction is setting the correct parameter to yield maximum accuracy. There are some fixed parameters for training the PSO which are said to give optimum performance based on research. These parameters include inertial weight, social weight and the cognitive weight. Inertial weight is said to be optimum when it is between 0.4 and 0.9. It is better to start from 0.9 and then moderate as need be. This system started from 0.9 but we experienced a better accuracy when we set inertial weight (ω) to 0.5. We also set the cognitive weight (c_1) and the social weight (c_2) to same value (1.4212) to reduce the chances of the system converging at the local minima.

Other parameters that are changed frequently as training commenced such as the number of epochs, number of layers and their various sizes, and number of particles were accepted via the Train API (Figure 8).

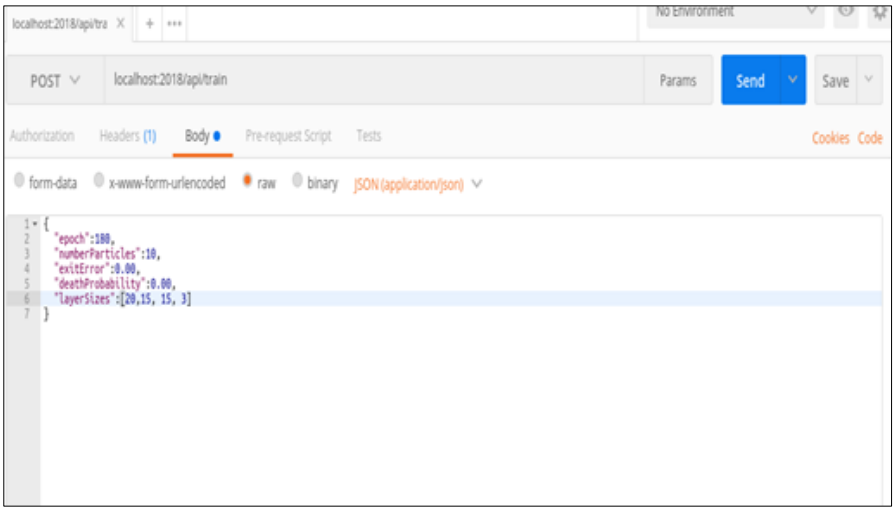


Figure 8 Training parameter API interfaces

In order to examine the best activation function for the system, a training using Hyperbolic Tangent and then sigmoid function. A better output layer was obtained using the SoftMax activation function from the output layer.

Table 4.1 shows the level of accuracy attained at different epoch using the Hyperbolic Tangent activation function and SoftMax while Table 4.2 shows the level of accuracy attained at different epoch using the sigmoid function and SoftMax.

$$Sigmoid(x_n) = \frac{1}{1 + e^{-(\theta + \sum w_{kn} + o_k)}}$$

$$hyperTan(x_n) = \max(0, x_n)$$

$$softmax(x_n) = \frac{e^{x_n}}{\sum e^{x_n}}$$

These trainings were carried out at different epoch and accuracy measured

Table 1 Accuracy with Hyperbolic Tangent and SoftMax

Epoch	20	40	80	100	120	140	160	180	200
Accuracy (%)	37.11	33.94	35.67	38.04	36.69	36.84	37.53	35.29	36.97

Table 2 Accuracy with Sigmoid and SoftMax

Epoch	20	40	80	100	120	140	160	180	200
Accuracy (%)	43.16	41.01	52.04	52.63	52.93	53.18	53.04	53.18	51.94

Comparing the predicted result, in relation to others like ANN which claim closely 100% accuracy but never published. Deep multi- scale convolution neural network and Next-step conditioning which predicted the 8-state of protein with accuracy of 71.4%

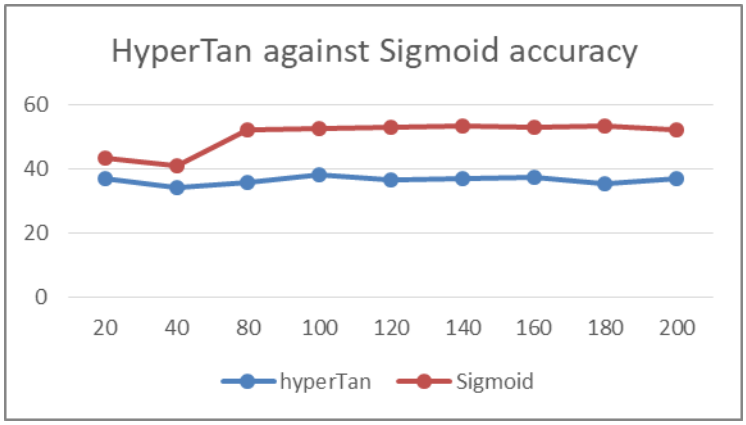


Figure 9 Accuracy of hyperbolic tangent against sigmoid function based on number of epochs After training, our system attained a maximum accuracy of 53.18% on epoch 180

5. Conclusion

Protein Secondary Structure is very essential in determining the function of proteins. These functions are essentials in medicine (drug design) and also in biotechnology (novel enzyme design) which makes it one of the major researches trending in bioinformatics and chemistry. This paper tries to solve this problem by training a deep neural network with basic particle swarm optimization but at the end of the research, it was acquired that the stated algorithm is not the best fit for this problem as the accuracy is lower compare to the prediction accuracy level experienced in JPred.

Recommendation

After carrying out the research, it is necessary to recommend that anybody who want to do any work on prediction of protein secondary structure using Deep neural network and particle swarm optimization algorithm should use other algorithm because comparing our result with others, our is lower in accuracies. Protein secondary structure prediction using deep neural network and particle swarm optimization algorithms is not a best fit.

Future work

One of the major challenges faced in this research is the basic PSO algorithm early convergence at a local minimal; studies revealed that the use of CLPSO (Commutative Learning Particle Swarm Optimization) algorithm which is an improvement over the basic PSO solves this problem considerately. This should also be explored in solving the problem of Protein Secondary structure prediction as the Swarm intelligence is showing some promising benefit

Compliance with ethical standards

Acknowledgments

The authors thank the reviewers for their insightful suggestion.

Disclosure of conflict of interest

On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

[1] Afonin, A. V., and Vashchenko, A. V. (2019). Benchmark calculations of intramolecular hydrogen bond energy based on molecular tailoring and function-based approaches: Developing hybrid approach. International Journal of Quantum Chemistry, 119(21), e26001.

[2] Akosua, B., Jasmin, C. andNavdeep, J. (2016). Protein secondary structure prediction using deep multi-scale convolutional neural networks and next-step conditioning. arXiv preprint:1611.01503 Available at: https://arxiv.org/abs/1611.01503v1.

[3] Barr, R. G., Bluemke, D. A., Ahmed, F. S., Carr, J. J., Enright, P. L., Hoffman, E. A., ... and Watson, K. E. (2010). Percent emphysema, airflow obstruction, and impaired left ventricular filling. New England Journal of Medicine, 362(3), 217-227.

[4] Beck, D. A., Alonso, D. O., Inoyama, D., and Daggett, V. (2008). The intrinsic conformational propensities of the 20 naturally occurring amino acids and reflection of these propensities in proteins. Proceedings of the National Academy of Sciences, 105(34), 12259-12264.

[5] Bioinfor, (2019a): Bioinformatics, Online Lectures on www.bioinfor.org.cn/lectures/index-91.html. Retrieved on 2nd of February, 2019.

[6] Bioinfor,(2019b): IUPAC Code, Online Lectures on www.bioinfor.org.cn/lectures/index-91.html. Retrieved on 2nd of February, 2019.

[7] Blake, C. C. F., Geisow, M. J., Oatley, S. J., Rerat, B., and Rerat, C. (1978). Structure of prealbumin: secondary, tertiary and quaternary interactions determined by Fourier refinement at 1.8 Å. Journal of molecular biology, 121(3), 339-356.

[8] Blundell, T. L., and Patel, S. (2004). High-throughput X-ray crystallography for drug discovery. Current opinion in pharmacology, 4(5), 490-496.

[9] Bordoloi, H. andSarma, K. K. (2014). Protein Structure Prediction using Artificial Neural Network. Available at: https://www.researchgate.net/publication/252067753_Protein_Structure_Prediction_using_Artificial_Neural_Network. Access on: 4th October, 2019.

[10] Boyken, S. E., Chen, Z., Groves, B., Langan, R. A., Oberdorfer, G., Ford, A., ... and Baker, D. (2016). De novo design of protein homo-oligomers with modular hydrogen-bond network–mediated specificity. Science, 352(6286), 680-687.

[11] Breda, A., Valadares, N. F., de Souza, O. N., and Garratt, R. C. (2007).Protein structure, modelling and applications Available at: https://www.ncbi.nlm.nih.gov/books/NBK6824/pdf/Bookshelf_NBK6824.pdf. Accessed on: 13thAug. 2019

[12] Busia, A., Collins, J., and Jaitly, N. (2016). Protein Secondary Structure Prediction Using Deep Multi-scale Convolutional Neural Networks and Next-Step Conditioning. arXiv preprint arXiv:1611.01503.

[13] Canziani, A., Paszke, A., and Culurciello, E. (2016). An analysis of deep neural network models for practical applications. arXiv preprint arXiv:1605.07678.

[14] Cormanich, R. A., Ducati, L. C., Tormena, C. F., and Rittner, R. (2013). A theoretical investigation of the dictating forces in small amino acid conformational preferences: The case of glycine, sarcosine and N, N-dimethylglycine. Chemical Physics, 421, 32-38.

[15] Diener, M., Adamcik, J., Sánchez-Ferrer, A., Jaedig, F., Schefer, L., and Mezzenga, R. (2019). Primary, secondary, tertiary and quaternary structure levels in linear polysaccharides: From random coil, to single helix to supramolecular assembly. Biomacromolecules, 20(4), 1731-1739.

[16] Dytfeld, D., Rosebeck, S., Kandarpa, M., Mayampurath, A., Mellacheruvu, D., Alonge, M. M., ... and Jakubowiak, A. J. (2015). Proteomic profiling of naïve multiple myeloma patient plasma cells identifies pathways associated with favourable response to bortezomib-based treatment regimens. British journal of haematology, 170(1), 66-79.

[17] Hoijat, R., Amin, R. andEffat, D. (2016).Cuckoo Search Algorithm and Its Application for Secondary Structure Protein Structure Prediction. Journal of Informatics and Computer Engineering (JICE) Vol. 2(3), pp. 134 – 139. DOI: 649123/220124. Accessed on: 17th September, 2019.

[18] Howles, C. M. (1996). Genetic engineering of human FSH (Gonal-F®). Human Reproduction Update, 2(2), 172-191.

[19] Hutchison, C. A., Chuang, R. Y., Noskov, V. N., Assad-Garcia, N., Deerinck, T. J., Ellisman, M. H., ... and Venter, J. C. (2016). Design and synthesis of a minimal bacterial genome. Science, 351(6280).

[20] Islam, MdNasrul (2015). A Balanced Secondary Structure Predictor.University of New Orleans Teses and Dissertations. Paper 1995 Available at: http://scholarworks.uno.edu/cgi/viewcontent.cgi?article=3100andcontext=td. Accessed on: 29th March., 2020

[21] Kabsch W, Sander C., (2018). Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers; 22(12):2577-637.

[22] Kumar, L. S., and Khan, S. F. (2017). Unit-19 Proteins and Nucleic Acids.

[23] Kurgan, L., and Miri Disfani, F. (2011). Structural protein descriptors in 1-dimension and their sequence-based predictions. Current Protein and Peptide Science, 12(6), 470-489.

[24] Kumar, W. K., Gupta, C. D., Bordoloi, J. and Kumar, Y. (2014).MeeteiMayek Unicode Modeling Using Swarm Intelligence and Neural Networks.International Journal of Engineering Research and Applications, 4(4): 216-224.

[25] Pauling L, Corey R. B, and Branson H. R. (1951). The Structure of Proteins: two hydrogen-bonded helical configurations of the polypeptide chain. Proc. Natl. Acad. Sci. U.S.A. 37, 205-211

[26] Rost, B. (2001). Protein secondary structure prediction continues to rise. Journal of structural biology, 134(2-3), 204-218.

[27] Sevendaysperweek, (2019). STPM Biology Biological Molecules (Part15), sevendaysperweek.blogspot.com.ng/2015/10/stpm-biology-biological-molecules-part_4.html, Retrieved on the 2nd February, 2019.

[28] Sonderby S. K andWinther O (2014).Protein Secondary Structure Prediction with Long Short Term Memory Networks.arXiv: 1412.7828 [q-bio.QM] Available at: https://arxiv.org/pdf/1412.7828.pdf. Accessed on: 5th October, 2019.

[29] Vorgelegt V, Quoc-Vu H. N, Bonn A (2008). Protein Structure and Enzyme Catalysis: Knowledge-Based Protein Loop Prediction and Initio Equilibrium Constant Estimation.

[30] Walsh, C. T., O'Brien, R. V., and Khosla, C. (2013). Nonproteinogenic amino acid building blocks for nonribosomal peptide and hybrid polyketide scaffolds. AngewandteChemie International Edition, 52(28), 7098-7124.

[31] Wang, J., Sanmamed, M. F., Datar, I., Su, T. T., Ji, L., Sun, J.and Chen, L. (2019). Fibrinogen-like protein 1 is a major immune inhibitory ligand of LAG-3. Cell, 176(1-2), 334-347.

[32] Wang, S., Peng, J., Ma, J., and Xu, J. (2016). Protein secondary structure prediction using deep convolutional neural fields. Scientific reports, 6(1), 1-11.

[33] Wang, S., Peng, J., Ma, J., andXu, J. (2016). Protein Secondary Structure Prediction Using Deep Convolutional Neural Fields. Scientific Reports, 6, 18962.http://doi.org/10.1038/srep18962. Accessed on: 5th October, 2020

[34] Wang, Z., Gerstein, M., and Snyder, M. (2009). RNA-Seq: a revolutionary tool for transcriptomics. Nature reviews genetics, 10(1), 57-63.

[35] Wright, L. K., Fisk, J. N., and Newman, D. L. (2014). DNA→ RNA: What do students think the arrow means? CBE—Life Sciences Education, 13(2), 338-348.

[36] Yan, R., Song, J., Cai, W. and Zhang, Z. (2015) A Short Review on Protein Secondary Structure Prediction Methods, in Pattern Recognition in Computational Molecular Biology: Techniques and Approaches (eds M. Elloumi, C. S. Iliopoulos, J. T. L. Wang and A. Y. Zomaya), John Wiley and Sons, Inc, Hoboken, NJ, USA. doi: 10.1002/9781119078845.ch6. Available: http://onlinelibrary.wiley.com/doi/10.1002/9781119078845.ch6/pdf. Accessed on: 17th September, 2019.

[37] Yatskevich, S., Rhodes, J., and Nasmyth, K. (2019). Organization of chromosomal DNA by SMC complexes. Annual Review of Genetics, 53, 445-482.

[38] Yegnanarayana, B. (2009). Artificial neural networks. PHI Learning Pvt. Ltd..4.

[39] Zhang, B., Li, J., and Lü, Q. (2018). Prediction of 8-state protein secondary structures by a novel deep learning architecture. BMC bioinformatics, 19(1), 1-13.