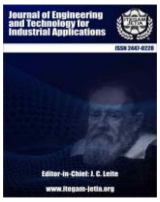
Journal of Engineering and Technology for Industrial Applications

ISSN ONLINE: 2447-0228

ITEGAM-JETIA

Manaus, v.7 n.27, p. 49-56. Jan/Feb, 2021

DOI: https://doi.org/10.5935/jetia.v7i27.732



RESEARCH ARTICLE OPEN ACCESS

PREDICTING PROTEIN SECONDARY STRUCTURE BASED ON ENSEMBLE NEURAL NETWORK

Emmanuel Gbenga Dada*[1], David Opeoluwa Oyewola[2], Joseph Hurcha Yakubu[3] and Ayotunde Alaba Fadele[4]

[1] Department of Mathematical Sciences, Faculty of Science, University of Maiduguri, Maiduguri, Nigeria.
[2] Department of Mathematics and Computer Science, Federal University Kashere, Gombe, Nigeria.
[3] Department of Mathematical Sciences, University of Maiduguri, Maiduguri, Nigeria.
[4] Department of Computer Science, Federal College of Education, Zaria, Kaduna State, Nigeria.

[1] http://orcid.org/0000-0002-1132-5447, [2] http://orcid.org/0000-0001-9638-8764, [3] http://orcid.org/0000-0003-2168-9882,
[4] http://orcid.org/0000-0002-1125-0780

Email: *gbengadada@unimaid.edu.ng, davidakaprof01@yahoo.com, hurch65@gmail.com, ayotundefadele@yahoo.com

ARTICLE INFO

Article History
Received: December 14th, 2020
Accepted: February 03th, 2021
Published: February 15th, 2021

Keywords:
Protein secondary structure prediction,
Ensemble Neural Network,
Feed Forward Neural Network (FFNN),
Recurrent Neural Network (RNN),
Cascade Forward Network (CFN).

ABSTRACT

Protein structure prediction is very vital to innovative process of discovering new medications based on the knowledge of a biological target. It is also useful for scientifically exposing the biological basis of convoluted diseases and drug effects. Despite its usefulness, protein structure is very complex, thereby making its prediction to be arduous, timewasting and costly. These drawbacks necessitated the need to develop more effective techniques with high prediction capability. Conventional techniques for predicting protein structure are ineffective, perform poorly, expensive and slow. The reasons for these are due to the vague dissimilar sequences among protein structures, meaningless protein data, high dimensional data, and having to deal with highly imbalanced classification task. We proposed an Ensemble Neural Network learning model that consists of some Neural Network algorithms such as Feed Forward Neural Network (FFNN), Recurrent Neural Network (RNN), Cascade Forward Network (CFN) and Non-linear Autoregressive Network with Exogenous (NARX) models. These models were trained using training algorithms such as Levenberg-Marquardt (LM), Resilient Back Propagation (RBP) and Scaled Conjugate Gradient (SCG) to improve the performance. Experimental results show that our proposed model has superior performance compared to the other models compared.

I. INTRODUCTION

It Proteins are sophisticated molecular structures that universally execute the cell to cell routines that are essential to support life. Though the organic proteins have been very playing very remarkable roles, it is only a small percentage of the entire probable amino acid structures that become visible in nature. Bioinformatics which is a collaboration between biology and computer science provides an avenue that allows more comprehensive exploration of protein's sequence space to develop artificial proteins with enhanced robustness and greater usefulness in comparison with their natural equivalents. It is clear from [1, 2] that several protein functions are facilitated by protein–protein interactions (PPIs). Therefore, restructuring their interfaces to improve or fine-tune the connecting attraction and connecting mode of PPIs is a useful technique to improve the functions of proteins [3]. This method has been used productively to remodel different protein systems [4-8], and has an enormous capacity for the design and implementation of an innovative curative, globular protein and other beneficial proteins.

There are two basic computational techniques for predicting protein secondary structure: template-based approach and machine learning approach. The drawbacks of the template-based approach are that it has lower accuracy compared to a machine learning approach, and performs poorly on non-homologous proteins [9]. Machine learning (ML) usually predict protein structure by first extracting essential features from protein sequences. The shortcoming of this approach is that the extracted features might not contain all the information that is in a protein sequence. This means that some important information might have been lost [10-12].

On the other hand, from the viewpoint of biology, protein sequence has very important information that enables it to take on specific structures [9, 12, 13]. Even though it is very interesting to predict such structures, serious problems are arising from it. To start with, the complexity of the connection between a sequence and its subsequent structure is a huge disadvantage [14, 15]. Moreover, the chosen features can considerably influence the efficiency of the learner [12]. Another issue is the problem of noise in the training data as well as protein sequences and their associated structures. Finally, there is the problem of class inequality in amino-acid samples as the classes are not evenly distributed [16].

Artificial Neural Networks (ANN) is a major recent advance in machine learning. It possesses the ability to automatically learn to appropriately represent primary data, detect attributes that are above average, enhance effectiveness over conventional models. ANN also improve the understanding of the significance of data and offers further understanding on the composition of biotic data [17]. Motivated by the characteristics of ANN.

This paper proposed an ensemble Neural Network based approach for prediction of protein secondary structure. Our main contributions in this paper are as stated below:

- We developed an ensemble Neural Network learning model that can process hidden contexts of input protein sequences and accurately predict their secondary structures.
- We used four (4) different neural network classifiers and jointly combined the classification results to represent the final results.
- Better classification accuracy of protein secondary structure was achieved through aggregation of ensemble results of Feed forward neural network (FFN), Cascade forward neural network, Recurrent neural network, neural network, and Non-linear autoregressive network with exogenous (NARX).

II. RELATED WORKS

The Several works have been done in the field of protein structure prediction. The ML based approach are one of the prevalent methods in use for the purpose. There are three popular sets of ML models that have found application for protein sequence-structure mapping. They include: artificial neural networks (ANN), deep learning techniques, and ensemble learners. ANN is the earliest type of ML algorithms that are applied for predicting protein secondary structure. In reality, using an ingenious ANN can accurately predict the boundaries of a class. Experimental results of various research have shown that recurrent neural networks (RNN) [18-20] are very effective for processing protein sequence data. A variant of RNN called bidirectional recurrent neural networks is known to exploit the data of the complete structure. Whereas time is a complex entity that plays a very important role in information retention [21-23]. It has been demonstrated that long temporary RNN possesses the ability to keep data over a long time span [24]. The Convolutional neural network (CNN) is a type of supervised Deep learning algorithm. The CNN is beneficial for predicting protein structure due to its dependable, robust, concurrent processing and self-learning proficiency. The CNN uses additional sequence information in it is learning process and makes allowances for mutuality of the neighbouring frame of reference [25]. Recently, research in both surface [26, 27] and DNN [28], aggregate the predictions of different networks in an ensemble manner.

An ensemble NN having three deep learning algorithms was developed by [28]. Though neural networks have several advantages, they are characterised with several weaknesses such as difficulty in choosing optima values for parameters like neurons, layers, and activation functions. The implication of this is the negative impact it has on the prediction result. There is also the drawback of the ANN algorithm getting stuck in the local minima.

To overcome these challenges associated with the prediction of the protein secondary structures, we employed an Ensemble Neural Network model that comprises of the following Feed forward neural network, Cascade FFN and RNN. We compared the performance of our proposed model with that of Pattern recognition neural network, NARX and Multilayer Perceptron neural network.

III. MATERIALS AND METHODS

III.1 NEURAL NETWORK ALGORITHMS

III.1.1 Feed Forward Neural Network (FFNN)

The FFNN is a very popular neural networks. It was developed as a result of the need to develop more efficient artificial neural network that will overcome weaknesses associated with back propagation learning algorithm. The FFNN feeds data from the inputs layers via the hidden layers to the outputs. The basic reason why it is referred to as feed forward network is that it uses forward propagation. The major strength of the FFNN is their ease of implementation and management. This makes them suitable for approximating any type of input and output representation [29]. The effectiveness of FFNN deeply hangs on the tuning of the weights of the nodes. The discrepancy between the result produced by the FFNN and the forecasted output is finalized after every iteration. As the neural network is separated with regard to their nodes, the training process becomes more controllable. The structure and operation of the FFN is in Figure 1.

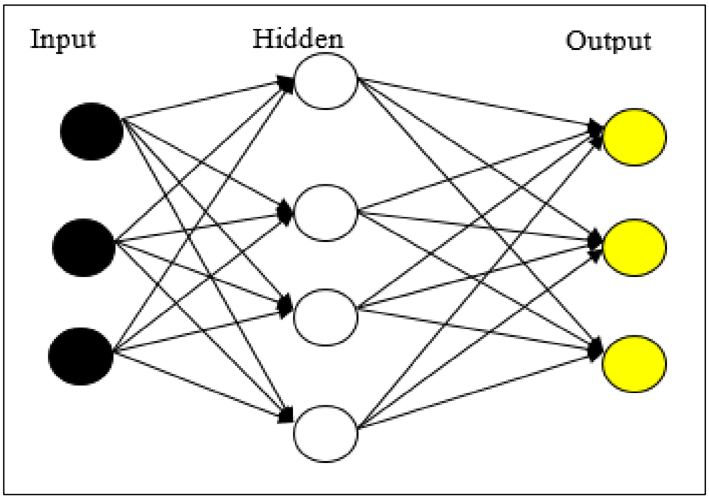


Figure 1: Feed Forward Neural Network.
Source: [29].

The activation function equation of the i^{th} hidden neuron is given in Equation 1:

$$h_i = f(u_i) = f(\sum_{k=0}^{K} w_{ki} x_k) \quad (1)$$

Where h_i is the i^{th} hidden neuron, $f(u_i)$ is the connection function that ensures that the outcome does not change in proportion to a change in any of the inputs among input and hidden layers, w_{ki} is the weight in the ki^{th} entry in a $(K\times N)$ weight matrix, x_k is the K input value.

$$y_j = f(u_j^1) = f(\textstyle\sum_{i=1}^{N} w_{ij} h_i) \qquad (2)$$

Where y_j is the j^{th} output value.

III.1.2 Cascade Forward Network

The cascade forward network (CFN) neural networks have some similarities with the FFNN because they also use back propagation algorithm for updating weights. The main difference is that they are made up of a weighted connection to the input of individual level as well as across one level to the successive levels [30]. It was opined that some cascade forward back propagation network can have superior performance compared to FFNN in many cases [31]. One of the striking characteristics of this network is that individual layer of neurons is connected to the entire preceding layer of neurons [32]. The pictorial depiction of CFN is in Figure 2.

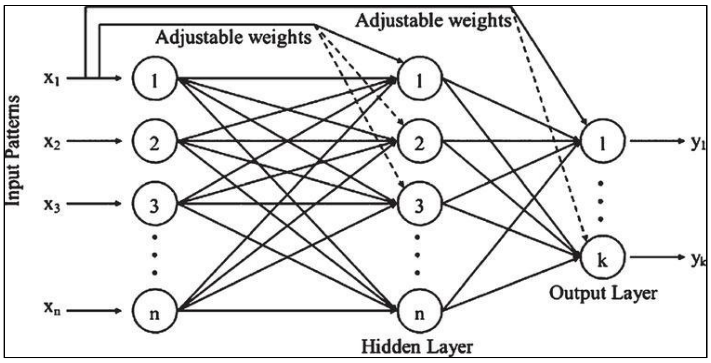


Figure 2: Cascade Forward Network.
Source: [32].

The mathematical equation for CFN is stated as:

$$y = \textstyle\sum_{i=1}^{n} f^i(w_i x_i) + f^o(\sum_{j=1}^{K} w_j^o f_j^h(\sum_{i=1}^{n} w_{ji}^h x_i)) \qquad (3)$$

The activation function from the input layer to the output layer is represented as f^i, while the weight from the input layer to the output layer is w_i^i. In a situation where bias is combined with the input layer, the activation function of each neuron in the hidden layer is represented as f^h so equation (3) can be expressed as:

$$y = \textstyle\sum_{i=1}^{n} f^i(w_i x_i) + f^o\left(w^b + \sum_{j=1}^{K} w_j^o f^h\left(w_j^b + \sum_{i=1}^{n} w_{ji}^h x_i\right)\right) \ (4)$$

III.1.3 Recurrent Neural Network (RNN)

RNN is distinctive for its extra set of responses from the generated result of layer that is concealed in between input and output layers. This layer constitutes the context layer that preserves data among observations [33]. The result of processing in a preceding phase can be carried over and used in the current period phase. This important attribute of the RNN offers a tremendously significant advantage, especially in real-time applications. RNN can have an unrestricted memory level and can therefore learn connections through time in addition to learning via the all current possible inputs [34, 35]. The RNN is depicted in Figure 3.

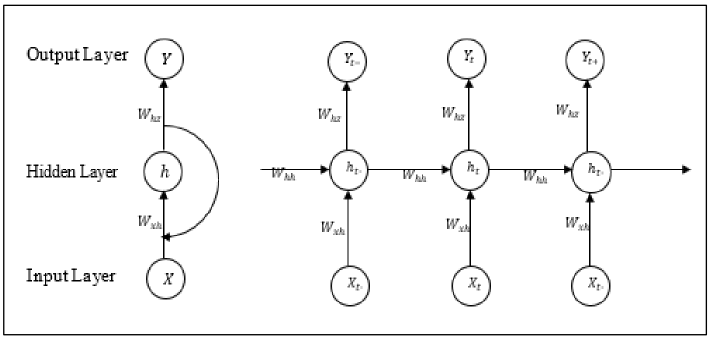


Figure 3: RNN Architecture.
Source: [33].

Input to hidden layer is expressed as in Equation 5 as:

$$h_t = g_n(W_{xh}X_t + W_{hh}h_{t-1} + b_h) \qquad (5)$$

Here h_t is the hidden layer at the instance t^{th}, moreover, g_n is the function, W_{xh} is the input to hidden layer of weight matrix, and X_t is the input at instance t^{th}. Also, h_{t-1} is the hidden layer at instance $t-1$, and the bias or threshold value is represented by b_h. Equation 6 which represents the hidden to output layer is stated as:

$$Z_t = g_n(W_{hz}h_t + b_z) \qquad (6)$$

Where the output vector is represented as Z_t, the hidden to output layer weight matrix is W_{hz}, and b_z is the bias or threshold.

III.1.4 Multi-Layer Perceptron Neural Networks

Multi-layer perceptron (MLP) It is made up of many perceptrons. MLP comprises of an input layer that accepts data, an output layer which generates results or prediction as regards the input, and in the middle of those two, there exist a set of random hidden layers that serves as the real computational engine of the MLP. The MLPs with one hidden layer is capable of estimating any function that is continuous at every value in an interval. Equation 7 expresses the rule used in updating the parameters (w_n,b_n):

$$w_{n+1} = w_n + cx \qquad b_{n+1} = b_n + c \qquad (7)$$

The condition for halting the algorithm is that a function that accepts a dataset as input and produces a decision as output that correctly categorizes the whole training datasets to various classes must be found [37]. Figure 4 below depicts a diagrammatical representation of MLP Neural Networks.

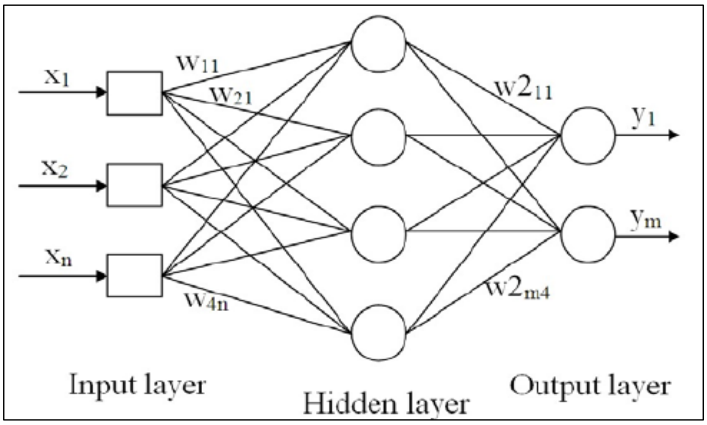


Figure 4: MLP Neural Networks Architecture.
Source: [37].

III.1.5 Non-linear Autoregressive (NARX) Network with Exogenous

Non-linear autoregressive (NARX) neural network is regarded as a classic time series predictor [38-40]. The idea behind NARX is a nonlinear overview of the Autoregressive Exogenous (ARX) that is categorized as a yardstick tool in linear system detection where the most important thing is fitting the data irrespective of the mathematical makeup of the model [41]. One of the major strength of the NARX models is their ability to model a wide-ranging array of nonlinear dynamic systems. They have used for solving many time-series modelling problems [42]. They are considered as a recurrent dynamic neural network with feedback connections that encircle many layers of the network [43]. It is necessary to fully exploit the NARX neural network memory capacity using the previous values of forecasted or actual time series. This will make the NARX neural network to have its optimal performance. Figure 5 depicts the architecture of the NARX.

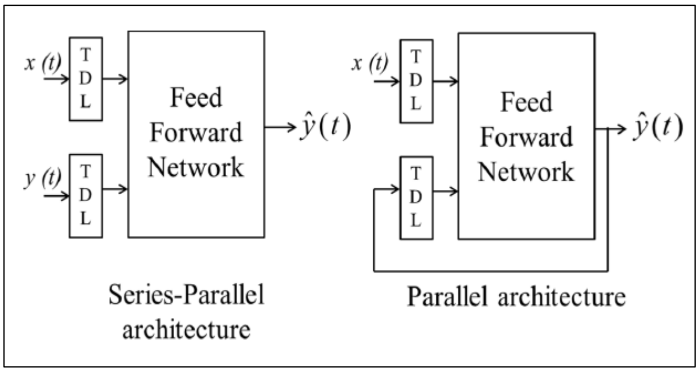


Figure 5: NARX neural network Architecture.
Source: [39].

III.2 NEURAL NETWORK TRAINING ALGORITHMS

In this paper three training algorithms namely: LM, SCG, and RBP were used. Description of these algorithms is done in this section. The Training Algorithms is used for weighting adjustments of RNN, CFN and FFN model used in this research.

III.2.1 Levenberg Marquardt (LM)

LM is a highly effective technique for weights adjustment. It is the fusion of the gradient descendent rule and the Gauss-Newton technique. LM determines the step size using a parameter that accepts big values for the initial iterations (the same as that of the Gradient Descent algorithm), and small values in the later phases (like what is obtainable in Gauss-Newton technique). The LM is a fusion of the strengths of both techniques. This makes it start converging from any early-stage like the Gradient Descent method. It also has quick convergence close to the neighbourhood of the least error the same way the Gauss-Newton method behaves. The LM technique however overcomes the weaknesses that are exhibited by both the Gradient Descent algorithm and Gauss-Newton technique [44, 45]. The Levenberg-Marquardt algorithm with vector of unknown parameters which are decided during step $K + 1$ is represented by Equation 8:

$$x_{k+1} = x_k^T - [J^T(x_k,t)J(x_k,t) + \mu_k I]^{-1} J^T(x_k,t) y(x_k,t) \quad (8)$$

LM with error is denoted by Equation 9:

$$I_2 = \int_0^T y^2(x_k,t)dt \quad (9)$$

Where:

$$y(x_k,t) = \int_0^t k(t-\tau)u(\tau)d\tau \quad (10)$$

$$J(a_k,t) = \begin{bmatrix} \frac{\partial y(x_k,t_1)}{\partial x_1} & \frac{\partial y(x_k,t_1)}{\partial x_2} & \cdots & \frac{\partial y(x,t_1)}{\partial x_m} \\ \vdots & & \ddots & \vdots \\ \frac{\partial y(x_k,t_n)}{\partial x_1} & & \cdots & \frac{\partial y(x_k,t_n)}{\partial x_m} \end{bmatrix} \quad (11)$$

Assuming:

$k = 1,2, \dots p$; p represents the number of iteration loops; $J_{n\times m}$ is the Jacobian matrix; $I_{m\times n}$ is the unit matrix; μ_k is the scalar and its value changes during iteration; $x = [x_1, x_2, \dots, x_m]$ is the model parameters searched for.

In situations where the parameters of the vector are not the best ones, and the value of error (8) is not at the lowest level. In such circumstance:

$$J^T(x_k,t)J(x_k,t) \ll \mu_k I \quad (12)$$

Can be taken and this results in the Gradient Descent technique which results in equation 15:

$$x_{k+1} = x_k^T - \frac{1}{\mu_k} J^T(x_k,t) y(x_k,t) \quad (13)$$

If the value of coefficient μ_k is small, it denotes that the values of the parameters of vector x are close to the best solution. At this instant:

$$J^T(x_k,t)J(x_k,t) \gg \mu_k I \quad (14)$$

Indicated that the LM algorithm is condensed to the Gauss-Newton method:

$$x_{k+1} = x_k^T - [J^T(x_k,t)J(x_k,t)]^{-1} J^T(x_k,t) y(x_k,t) \quad (15)$$

Computation of the Jacobian J regarding the weight and bias variables x is done using Backpropagation. The tuning of the variables is done according to Levenberg-Marquardt.

III.2.2 Resilient Back Propagation (RBP)

The RBP was first proposed by [46]. It is a supervised learning method that learns from the entire training dataset at once in FFN. The primary goal of the RBP is to eradicate the negative outcome of the volume of the partial derivative on the weight step. The aftermath of this is that the sign of the derivative is the only factor taken into account to specify the path through which the weight will be updated [46].

When the weight update is done using back propagation, the weight update is decided through the partial derivative expressed in equation 16:

$$\Delta_{wij}(t) = \alpha \times x_i(t) \times \delta_j(t) \quad (16)$$

Assuming α is the learning rate, $x_i(t)$ depict the propagation of inputs backwards to the i^{th}neuron at time step t, and δ is the equivalent error gradient. Unlike the RBP, Resilient propagation computes a distinctive delta Δ_{ij} for every connection. This plays a determining factor in choosing the magnitude of the weight update. The reader can consult Riedmiller [45] for a detailed explanation on the RBP algorithm.

III.2.3 Scaled Conjugate Gradient (SCG)

SCG algorithm was proposed by [47, 48]. SCG is a variant of conjugate gradient technique developed to lower the running time by employing LM algorithm to increase the step size of the line search for each learning iteration [44]. Through the step size scaling system, this technique circumvents the timewasting and inefficient line-search that characterized each learning iteration. SCG can effectively handle wide-ranging problems. It has proven over time to be very effective at handling training Feed Forward Neural networks and other outsized networks. According to Martin [50], SCG algorithm was derived from quadratic reduction of objective function E within N iterations to the lowest possible level.

III.3 PROPOSED ENSEMBLE NEURAL NETWORK TECHNIQUE

We employed four (4) variants of ANN classifiers (FFNN, RNN, CFN and NARX) as base learners and Random Forest (RF) was used as the top layer of our proposed model. Due to the complexity of protein structure, we integrate all the optimal outputs produced by FFNN, RNN, CFN and NARX after they were trained using LM, RBP and SCG. Figure 6 shows the architecture of the proposed Ensemble ANN.

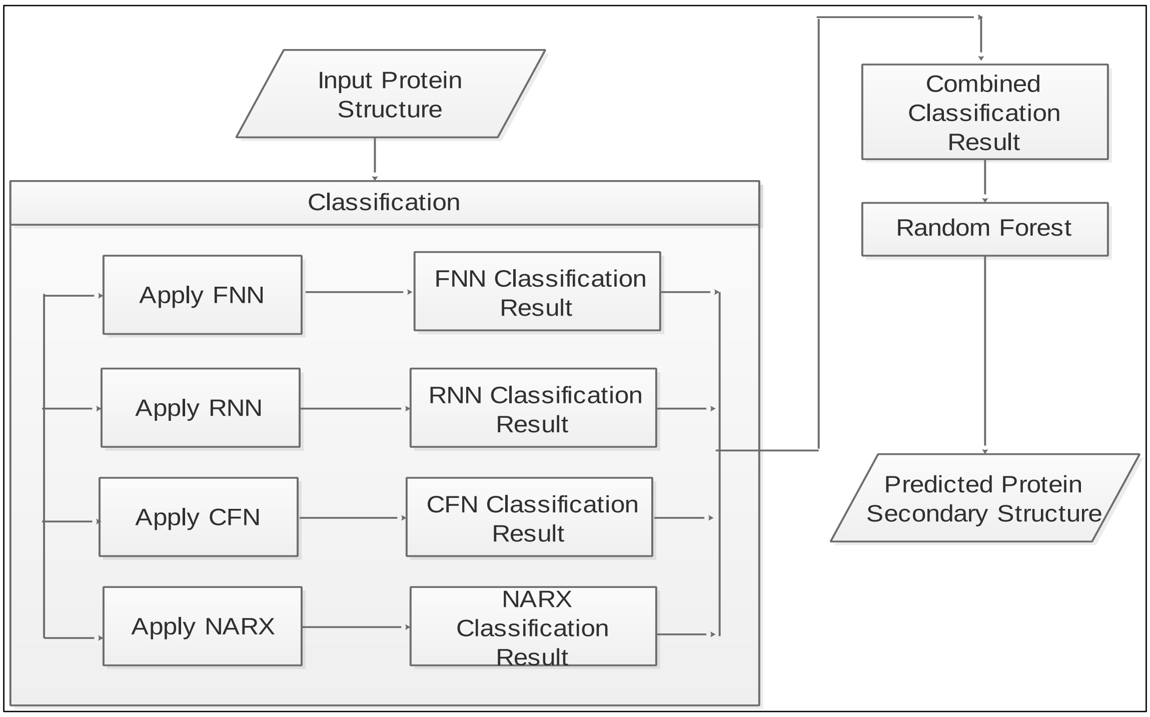

Input Protein Structure
Classification
Apply FNN
FNN Classification Result
Apply RNN
RNN Classification Result
Apply CFN
CFN Classification Result
Apply NARX
NARX Classification Result
Combined Classification Result
Random Forest
Predicted Protein Secondary Structure

Figure 6: Proposed Ensemble ANN.
Source: Authors, (2020).

In the proposed ensemble technique, the protein sequence is fed into each of the Neural Network algorithms in the learning stage. The output is regarded as the aggregated classification result of all the learning Neural Networks (FNN, RNN, CFN and NARX) which is used for the final prediction.

IV. RESULTS AND DISCUSSIONS

IV.1 EXPERIMENTAL SETUP

The dataset used for the experiments conducted in this work was obtained from the repository of Research Collaboratory for Structural Bioinformatics (RCSB) Protein Data Bank (PDB). The dataset is made up of several kinds of macromolecules of biological importance. A great part of the dataset is protein. Since the antecedent of DNA is RNA that can be converted, it therefore means that proteins are the biomolecules that are immediately interacting in biological routes and progressions. The repository contains over 400,000 annotated protein structures sequences which are publicly available at https://github.com/iamdebanjangoswami/Predictive-Protein-classification--Naive-Bayes-Classifier. All the simulation for this work was done using MATLAB 2018 version.

IV.2 PERFORMANCE METRICS

We evaluate the prediction performance of the ensemble ANN using three metrics: Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and Mean Absolute Scaled Error (MASE).

IV.2.1 Mean Absolute Error (MAE)

MAE is defined as the average of the difference between predicted and actual values in the test.

IV.2.2 Root Mean Square Error (RMSE)

RMSE is defined as the standard deviation of prediction errors in a test.

IV.2.3 Mean Absolute Scaled Error (MASE)

MASE is defined as is a measure of the accuracy of predictions. It is a scale-free error metric that gives each error as a percentage in comparison to a standard mean error.

IV.3 RESULT AND DISCUSSION

This section describes the statistical results of our simulations. The ensemble ANN was employed to classify and predict the protein sequence structure in the dataset. The dataset was divided into a training set (60%) and test set (40%). The individual classifiers were used to train the dataset. The proposed ensemble ANN produced a classification accuracy of 99.48%. Table 1 shows the simulation results of our proposed model in comparison with some models.

Table 1: Performance evaluation of models based on MAE, RMSE, MASE and Percentage Accuracy.

Model	**MAE**	**RMSE**	**MASE**	**% Accuracy**
FFN	0.04416	0.14632	1.1998	97.7454
CFN	0.03494	0.09252	0.9493	98.0573
RNN	0.04459	0.14593	1.2115	97.7447
PRNN	0.04302	0.14646	1.1688	97.7440
NARX	0.04902	0.14447	1.3318	97.7454
MLP	0.02254	0.15012	0.6124	97.7400
Ensemble NN	0.01352	0.0108	0.4108	99.4800

Source: Authors, (2020).

The comparison of accuracy for the different models considered in this paper is depicted in Figure 7 while the accuracy is presented in percentage in Figure 8.

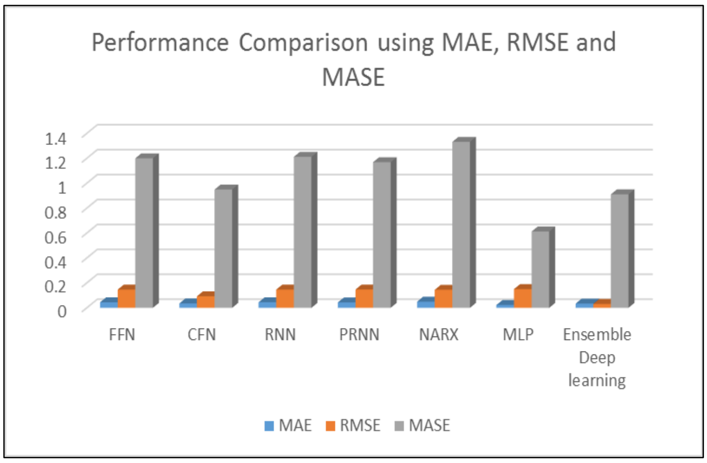


Figure 7: Comparison of models based on Classification Accuracy.
Source: Authors, (2020).

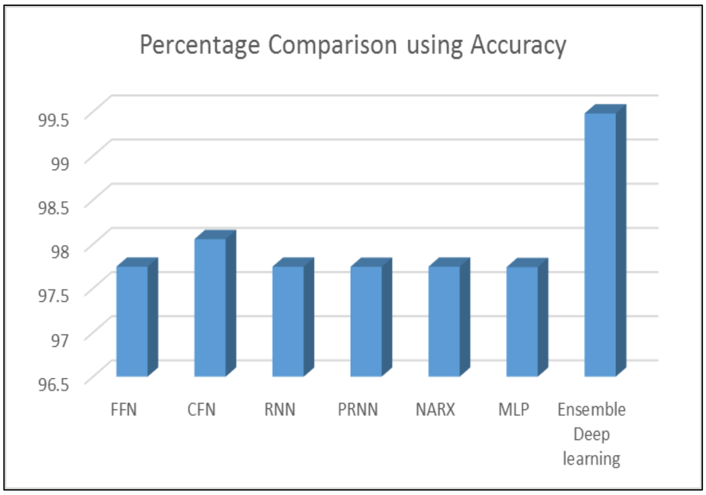


Figure 8: Performance Comparison of models using Percentage Accuracy.
Source: Authors, (2020).

Depicted in Table 2 is the confusion matrix of the various algorithms under consideration.

Table 2: Confusion Matrix.

Model	**DNA**	**Protein**	**Class**
FFN	0	0	DNA
	3188	138213	Protein
CFN	1036	595	DNA
	2152	137618	Protein
RNN	0	1	DNA
	3188	138212	Protein
PRNN	0	2	DNA
	3188	138211	Protein
NARX	0	0	DNA
	3188	138213	Protein
MLP	1	1	DNA
	3187	138212	Protein
Ensemble NN	2872	422	DNA
	315	137791	Protein

Source: Authors, (2020).

From Table 2, it is easy to draw a comparison between the actual class and predicted results . Ensemble accurately predicts 140663 instances out of 141400 instances (2872 DNA instances that are truly DNA and 137791 protein instance that are protein). And 737 instances wrongly predicted (422 instances of DNA class predicted as protein and 315 instances of protein class predicted as DNA). This explains why Ensemble produced superior prediction accuracy compared to other Neural Network models under consideration. From our experiments, it is obvious that Ensemble have superior performance in term of effectiveness and efficiency considering its classification accuracy and MASE.

V. CONCLUSIONS

An Ensemble ANN model for predicting protein secondary structure is proposed in this paper. The proposed model integrated different Neural Network algorithms for an enhanced predictive accuracy. The three ANN used are FFNN, RNN and CFN. Our statistical results show clearly that our model produced superior results compared to other six models compared. It can therefore be deduced that it is better to predict protein secondary structure by means of the fusion of different ANN rather than using the models alone. In the future, we hope to perform experiment with deep learning architectures and compared it is performance with the ensemble algorithm propose in this work. Also we intend to Extend this research by using moth flame optimisation, particle swarm optimization, grey wolf optimisation, genetic algorithms and others.

VI. AUTHOR'S CONTRIBUTION

Conceptualization: Emmanuel Gbenga Dada.
Methodology: Emmanuel Gbenga Dada, David Opeoluwa Oyewola and Joseph Hurcha Yakubu.
Investigation: Emmanuel Gbenga Dada and Ayotunde Alaba Fadele.
Discussion of results: Emmanuel Gbenga Dada, David Opeoluwa Oyewola and Joseph Hurcha Yakubu.
Writing – Original Draft: Emmanuel Gbenga Dada and David Opeoluwa Oyewola.
Writing – Review and Editing: Emmanuel Gbenga Dada, David Opeoluwa Oyewola, Joseph Hurcha Yakubu and Ayotunde Alaba Fadele.
Resources: Emmanuel Gbenga Dada and David Opeoluwa Oyewola.

Supervision: Emmanuel Gbenga Dada and David Opeoluwa Oyewola.

Approval of the final text: Emmanuel Gbenga Dada, David Opeoluwa Oyewola, Joseph Hurcha Yakubu and Ayotunde Alaba Fadele.

VIII. REFERENCES

[1] H. Jeong, S.P. Mason, A.L. Barabási and , Z.N. Oltvai. "Lethality and centrality in protein networks". Nature, 411(6833), 2001, pp.41-42.

[2] A. Szilagyi, V. Grimm, A.K. Arakaki and J. Skolnick. "Prediction of physical protein–protein interactions." Physical biology, 2(2), 2005, p.S1.

[3] J. Karanicolas, and B. Kuhlman, "Computational design of affinity and specificity at protein–protein interfaces," Curr. Opin. Struct. Biol. 19 (2009), 458–463.

[4] A. Chevalier, D.A. Silva, G.J. Rocklin, D.R. Hicks, R. Vergara, P. Murapa, S.M. Bernard, L. Zhang, K.H. Lam, G. Yao and C.D. Bahl, "Massively parallel de novo protein design for targeted therapeutics." Nature, 550 (7674), 2017, pp.74-79.

[5] G. Grigoryan, A.W. Reinke, A.E. Keating, "Design of protein interaction specificity gives selective bZIP-binding peptides", Nature 458, 2009, 859–864.

[6] S.J. Fleishman, T.A. Whitehead, D.C. Ekiert, C. Dreyfus, J.E. Corn, E.M. Strauch, I.A. Wilson and D. Baker. "Computational design of proteins targeting the conserved stem region of influenza hemagglutinin". Science, 332(6031), 2011, pp.816-821.

[7] N.P. King, W. Sheffler, M.R. Sawaya, B.S. Vollmar, J.P. Sumida, I. André, T. Gonen, T.O. Yeates and D. Baker. "Computational design of self-assembling protein nanomaterials with atomic level accuracy". Science, 336(6085), 2012, pp.1171-1174.

[8] D. Shultis, P. Mitra, X. Huang, J. Johnson, N.A. Khattak, F. Gray, C. Piper, J. Czajka, L. Hansen, B. Wan and K. Chinnaswamy. "Changing the apoptosis pathway through evolutionary protein design". Journal of molecular biology, 431(4), 2019, pp.825-841.

[9] H. N. Lin, T.Y. Sung, S.Y. Ho and W.L. Hsu, "Improving protein secondary structure prediction based on short subsequences with local structure similarity", Bmc Genomics, vol. 11, 2010, p. S4 BioMed Central.

[10] Y. Chen, "Long sequence feature extraction based on deep learning neural network for protein secondary structure prediction", Information Technology and Mechatronics Engineering Conference (ITOEC), IEEE, 2017, pp. 843–847 2017 IEEE 3rd.

[11] Z. Li, J. Wang, S. Zhang, Q. Zhang, and W. Wu, "A new hybrid coding for protein secondary structure prediction based on primary structure similarity", Gene, 618, 2017, pp. 8–13.

[12] T. Liu, X. Zheng and J. Wang, "Prediction of protein structural class using a complexity based distance measure", Amino Acids, 38 (3), 2010, pp. 721–728.

[13] Y. Liu, J. Cheng, Y. Ma and Y. Chen, "Protein secondary structure prediction based on two dimensional deep convolutional neural networks", 3rd IEEE International Conference on Computer and Communications (ICCC), IEEE, 2017, pp. 1995–1999.

[14] E. Krissinel, "On the relationship between sequence and structure similarities in proteomics", Bioinformatics, 23 (6), 2007, pp. 717–723.

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

[16] M. Alirezaee, A. Dehzangi and E. Mansoori, "Ensemble of neural networks to solve class imbalance problem of protein secondary structure prediction", International Journal of Artificial Intelligence & Applications, 3 (6), 2012, pp. 9.

[17] E.G. Dada, J.S. Bassi, H., Chiroma, M.A. Shafi'i, A.O. Adetunmbi, and O.E. Ajibuwa, "Machine learning for email spam filtering: review, approaches and open research problems". Heliyon, 5(6), 2019, p.e01802.

[18] S. Babaei, A. Geranmayeh, S.A. Seyyedsalehi, "Towards designing modular recurrent neural networks in learning protein secondary structures", Expert Syst. Appl. 39 (6), 2012, pp. 6263–6274

[19] F. Masulli, S. Mitra, "Natural computing methods in bioinformatics: a survey", Inf. Fusion 10 (3), 2009, pp. 211–216.

[20] G. Pollastri, A.J.M. Martin, C. Mooney, A. Vullo, "Accurate prediction of protein secondary structure and solvent accessibility by consensus combiners of sequence and structure information", BMC Bioinf., 8 (1), 2007, p. 201.

[21] S. Hosseini, H. Yin, N.M. Cheung, K.P. Leng, Y. Elovici and X. Zhou. "Exploiting reshaping subgraphs from bilateral propagation graphs." In International Conference on Database Systems for Advanced Applications, 2018, (pp. 342-351). Springer, Cham.

[22] S. Hosseini, H. Yin, M. Zhang, X. Zhou and S. Sadiq, "Jointly modeling heterogeneous temporal properties in location recommendation", International Conference on Database Systems for Advanced Applications, Springer, 2017, pp. 490–506.

[23] S. Hosseini, H. Yin, X. Zhou, S. Sadiq, M.R. Kangavari, N.M. Cheung, "Mining subgraphs from propagation networks through temporal dynamic analysis." In 2018 19th IEEE International Conference on Mobile Data Management (MDM) 2018, (pp. 66-75). IEEE.

[24] K. Paliwal, J. Lyons, R. Heffernan, "A short review of deep learning neural networks in protein structure prediction problems", Advanced Techniques in Biology & Medicine, 2015, pp. 1–2.

[25] J. Zhou and O.G. Troyanskaya "Deep supervised and convolutional generative stochastic network for protein secondary structure prediction." 2014, arXiv preprint arXiv:1403.1347.

[26] L. Khalatbari and M.R. Kangavari, "Protein secondary structure prediction: a literature review with focus on machine learning approaches." Journal of Computer & Robotics, 8(1), 2015, pp.9-24.

[27] G. Wang, Y. Zhao, and D. Wang, "A protein secondary structure prediction framework based on the extreme learning machine", Neurocomputing 72 (1–3), 2008, pp. 262–268.

[28] M. Spencer, J. Eickholt and J. Cheng, "A deep learning network approach to ab initio protein secondary structure prediction", IEEE ACM Trans. Comput. Biol. Bioinform 12 (1), 2015, pp. 103–112.

[29] K., M. Hornik Stinchcombe and H. White. "Multilayer feedforward networks are universal approximators." Neural Networks, 2(5), 1989, pp. 359-366.

[30] M. Lashkarbolooki and Z.S. Shafipour, "Trainable cascade-forward back-propagation network modeling of spearmint oil extraction in a packed bed using SC-CO2". The Journal of Supercritical Fluids, 73, 2012, pp. 108-115.

[31] U.B. Filik and M. Kurban. "A new approach for the short-term load forecasting with autoregressive and artificial neural network models." International Journal of Computational Intelligence Research, 3(1), 2007, pp. 66-71.

[32] H. Demuth, M. Beale and M. Hagan. "Neural Network Toolbox User‟s Guide". The MathWorks, Inc., Natrick, USA. 2009.

[33] M. Lashkarbolooki, B. Vaferi, A. Shariati, and A.Z. Hezave. "Investigating vapor–liquid equilibria of binary mixtures containing supercritical or near-critical carbon dioxide and a cyclic compound using cascade neural network". Fluid Phase Equilibria, 343, 2013, pp.24-29.

[34] B.K. Chauhan, A. Sharma and M. Hanmandlu. "August. Neuro-Fuzzy Approach Based Short Term Electric Load Forecastig". In 2005 IEEE/PES Transmission & Distribution Conference & Exposition: Asia and Pacific, 2005, (pp. 1-5). IEEE.

[35] H. Demuth, M. Beale and M.Hagan (2009). "Neural Network Toolbox User‟s Guide". The MathWorks, Inc., Natrick, USA. 2009.

[36] J.L. Elman "Finding structure in time". Cognitive Science, 14(2), 1990, pp. 179-211.

[37] S. Haykin. "Neural Networks: A Comprehensive Foundation". 2nd ed. New Jersey: Prentice Hall., 1999, p. 823.

[38] H. Jia "Investigation into the effectiveness of long short term memory networks for stock price prediction". arXiv preprint arXiv , 2016, 1603.07893

[39] M. Awad and M. Foqaha. "Email Spam Classification Using Hybrid Approach of RBF Neural Network and Particle Swarm Optimization." International Journal of Network Security and its Applications, Vol. 8, no. 4, 2016, pp. 17-28.

[40] O.A. Carpinteiro, I. Lima, J.M. Assis, A.C.Z. de Souza, E.M. Moreira, and C.A. Pinheiro, "A neural model in anti-spam systems". In International Conference on Artificial Neural Networks, 2006, (pp. 847-855). Springer, Berlin, Heidelberg.

[41] S. Mohanty, P.K. Patra, and S.S. Sahoo. "Prediction of global solar radiation using nonlinear autoregressive network win exogenous inputs (narx)". In Proceedings of the 39th National System Conference (NSC), Noida, India, 14–16 December 2015.

[42] E. Pisoni, M. Farina, C. Carnevale, and L. Piroddi, "Forecasting peak air pollution levels using NARX models." Eng. Appl. Artif. Intell. 2009, 22, pp. 593–602.

[43] L.G.B. Ruiz, M.P. Cuéllar, M.D. Calvo-Flores, and M.D.C.P. Jiménez, "An Application of Non-Linear Autoregressive Neural Networks to Predict Energy Consumption in Public Buildings". Energies, 2016, 9, 684.

[44] K. Levenberg, "A Method for the Solution of Certain Non-Linear Problems in Least Squares." The Quarterly of Applied Mathematics. 2, 1944, pp. 164-168.

[45] D. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters", SIAM J. Appl. Math., Vol. 11, 1963, pp. 431–441.

[46] M. Riedmiller, "Untersuchungen zu konvergenz und generalisierungsverhalten uberwachter lernverfahren mit dem SNNS". In Proceedings of the SNNS 1993 workshop.

[47] M.F. Møller, A scaled conjugate gradient algorithm for fast supervised learning. Aarhus University, 1990, Computer Science Department.

[48] C. Tran, A. Abraham and L. Jain "Decision support systems using hybrid neurocomputing". Neurocomputing, 61, 2004, pp.85-97.

[49] M. Riedmiller and H. Braun. "Rprop: A fast adaptive learning algorithm." In Proc. of the Int. Symposium on Computer and Information Science VII. 1992.

[50] J.T. Wei, Z. Zhang, S.D. Barnhill, K.R. Madyastha, H. Zhang, and J.E. Oesterling. "Understanding artificial neural networks and exploring their potential applications for the practicing urologist". Urology, 52(2), 1998, pp.161-172.