A novel optimized deep learning method for protein-protein prediction in bioinformatics

Preeti Thareja, Rajender Singh Chhillar
Department of Computer Science and Applications, Maharshi Dayanand University, Rohtak, India

Article Info

Article history:
Received Jun 9, 2023
Revised Sep 14, 2023
Accepted Oct 9, 2023

Keywords:
Aquila optimization
Long short-term memory
Protein-protein interaction prediction
Restricted Boltzmann machines
Semantic similarity
Shark nose optimization

ABSTRACT

Proteins have been shown to perform critical activities in cellular processes and are required for the organism's existence and proliferation. On complicated protein-protein interaction (PPI) networks, conventional centrality approaches perform poorly. Machine learning algorithms based on enormous amounts of data do not make use of biological information's temporal and spatial dimensions. As a result, we developed a sequence-dependent PPI prediction model using an Aquila and shark noses-based hybrid prediction technique. This model operates in two stages: feature extraction and prediction. The features are acquired using the semantic similarity technique for good results. The acquired features are utilized to predict the PPI using hybrid deep networks long short-term memory (LSTM) networks and restricted Boltzmann machines (RBMs). The weighting parameters of these neural networks (NNs) were changed using a novel optimization approach hybrid of aquila and shark noses (ASN), and the results revealed that our proposed ASN-based PPI prediction is more accurate and efficient than other existing techniques.

1. INTRODUCTION

Protein-protein interactions (PPIs) can be utilized to look into the mechanisms underlying many biological processes, such as deoxyribonucleic acid (DNA) replication, protein modification, and signal transmission. Due to their accurate understanding and analysis, which can reveal numerous roles at the molecular and proteome levels, PPIs have been a research focus [1], [2]. On the other hand, there are problems with incomplete and imprecise prediction using web-lab identification methods [3], [4]. Alternately, low-cost candidates for future experimental validation could be obtained by applying precise bioinformatics methods for PPI prediction [5], [6].

Using advanced techniques to calculate PPI is not only laborious and costly, but it also produces an excessive number of false positives and false negatives [7], [8]. As a result, computational tools that can aid in the process of discovering genuine protein interactions are required. This problem can be viewed as a categorical classifying problem from the standpoint of machine learning, and it can be tackled using supervised learning methods [9], [10]. With the accelerated growth of deep learning techniques and neural network infrastructure, certain machine intelligence-based and sequence-based models for PPI prediction have been developed. Table 1 shows a summary of the state-of-art methods.

Li et al. [11] proposed DeepCellEss which is a methodology for easy-to-interpret deep learning (DL) based on sequences and cell-line-specific key protein predictions. To extract minute and prolonged-range
hidden features from protein sequences, DeepCellEss uses a convolutional network and bidirectional long short-term memory (LSTM). Additionally, to enable the residue-level point process, a multi-head self-attention technique is adopted. Numerous computer studies show that DeepCellEss beats previous sequence-based approaches as well as network-based clear implications and provides effective prediction results for distinct cell lines.

Hou et al. [12] created a method for identifying PPI sites based on an ensemble deep learning method for protein-protein interaction (EDLMPPI). This would aid in solving the issue of modeling the properties of amino acid (AA) sequences for PPI bindings by directly encoding them into distributed vector representations. Additionally, their performance could stand to be improved when AA sequences are directly encoded into distributed vector model to categorize PPI binding events because the experiment numbers for detected PPI sites are significantly less than the number of PPIs or protein domains in protein complexes.

Gao et al. [13] offered the HIGH-PPI two-side learning hierarchical graph network to forecast PPIs and deduce the relevant chemical information. A vertex in the graph (top outer view) is a protein graph in this model's hierarchical graph (bottom inside-of-protein view). To effectively depict the quality of support of the protein, a set of chemically pertinent descriptors rather than protein sequences are used in the bottom view. To create a solid machine understanding of PPIs, HIGH-PPI investigates the human interactome’s inside and outside of protein components. In terms of forecasting PPIs, this model has good accuracy and durability.

Yue et al. [14] introduced a deep learning framework for identifying important proteins. Their research focused on three main objectives: investigating the significance of each element’s value in model prediction, improving the handling of unbalanced datasets, and assessing the model’s accuracy in predicting important proteins. They used node2vec for feature representation and depth-wise separable convolution for gene expression profiles. Results on Saccharomyces cerevisiae (S. cerevisiae) data demonstrated their model’s superiority over traditional deep learning methods.

In their 2022 study, Díaz-Eufracio and Medina-Franco [15] developed ensemble models utilizing support vector machine (SVM), logistic regression (LR), and random forest (RF) algorithms, employing an extended connectivity fingerprint radius of 2. Their primary objective was to validate newly generated PPI inhibitors from apothecary sources. The significance of their research lies in the predictive models they have created, which will empower future initiatives in designing PPI inhibitors to make informed, data-driven choices.

<table>
<thead>
<tr>
<th>Reference</th>
<th>Techniques used</th>
<th>Features</th>
<th>Issues</th>
<th>Dataset used</th>
</tr>
</thead>
<tbody>
<tr>
<td>[11]</td>
<td>CNN, bidirectional long short-term memory neural network (BiLSTM) and capsule network</td>
<td>Attention scores play a vital role in enabling the identification of essential sequence regions for predicting outcomes specific to various cell lines. They facilitate in-depth research and comparisons for critical cell line-specific proteins.</td>
<td>Does not reflect the relationships between several cell lines within the same tissue or cancer type.</td>
<td>Nucleotide, Protein Sequences</td>
</tr>
<tr>
<td>[12]</td>
<td>BiLSTM and capsule network</td>
<td>Work directly with AA sequences.</td>
<td>Need to add more dynamic word embedding models to the model and modify them to address further pertinent protein-identifying issues.</td>
<td>Dset_448, Dset_72, and Dset_164</td>
</tr>
<tr>
<td>[13]</td>
<td>Graph convolutional network</td>
<td>Its capacity to recognize residue significance for PPI is a positive sign of great interpretability.</td>
<td>Protein-level annotations weren’t fully explored, and memory needs increase with more views in a hierarchical graph.</td>
<td>PPI Sequences</td>
</tr>
<tr>
<td>[14]</td>
<td>1D convolution</td>
<td>On gene expression profiles, the notion of depth wise separable convolution is applied to extract attributes.</td>
<td>Using a long vector to represent subcellular localization demands significant processing resources.</td>
<td>S. cerevisiae</td>
</tr>
<tr>
<td>[15]</td>
<td>RF, LR, SVM</td>
<td>Assess ML models for classifying new inhibitors by chemists and maintain the PPI inhibitors database regularly.</td>
<td>For classification, new models and challenges can be applied.</td>
<td>PPI Inhibitors</td>
</tr>
</tbody>
</table>

Deep learning (DL) methodologies, as documented in references [16], [17], encompass a range of techniques, including support vector machines (SVM) [18], artificial neural networks (ANN) [19], and others. These approaches offer indispensable tools for the secure prediction of PPI by extracting essential peptide information from amino acid sequences [20]. This research demonstrates that deep learning frameworks [21]...
A novel optimized deep learning method for protein-protein prediction in bioinformatics (Preeti Thareja)
covariance (AC). This transformation ensures that all proteins, regardless of their varying amino acid content, are represented by matrices of the same length.

The $i^{th}$ physicochemical property scale’s auto covariance $AC_{i,g}$ is provided by (1) and (2):

$$AC_{i,g} = \frac{1}{L-g} \sum_{m=1}^{L-g} (P_{i,m} - \gamma_i) \times (P_{i,m+g} - \gamma_i) \quad (1)$$

$$\gamma_i = \frac{1}{L} \sum_{m=1}^{L} P_{i,m} \quad (2)$$

where $g$ denotes the predefined gap, $L$ indicates the protein $P$‘s length, while $\gamma_i$ indicates the mean of protein $P$‘s $i^{th}$ physicochemical scale values. By fixing the greatest spacing to $G(g = 1, 2, ..., G)$, every protein may be initialized of $k \times G$ elements, where $k$ seems to be the physicochemical property scales count.

### 2.2.2. Semantic similarity-based feature extraction

The semantic similarity identification technique has been utilized to compute the degree of similarity. To compute the resemblance, every source is represented as a vector. In particular, the vector model faces issues such as word impropriety (e.g., disregard synonymy) as well as lacking semantic data. The point word recognition (FWR) technique incorporates semantic meaning into the vector model, hence removing vector semantic problems. Throughout this application, the species sensitivity distributions (SSD) approach is primarily concerned with determining the associations between every pair of resources by using a cosine similarity metric. Syntax, as well as semantic similarity, are integrated into SSD cosine similarity as can be expressed with (3).

$$\text{Sem}_{sim}(R_{a'}, R_b) = \frac{\rho_{a'b}}{|\rho_{a'b}| |\rho_{a'a}|} = \frac{\sum_{i=1}^{m}(\omega_{a'i} \times \omega_{b'i})}{\sqrt{\sum_{i=1}^{m}(\omega_{a'i})^2 \times \sum_{i=1}^{m}(\omega_{b'i})^2}} \quad (3)$$

### 2.2. Optimal trained hybrid classifier

The extracted features are subjected to the prediction model where a hybrid model that combines improved LSTM and the RBMs classifiers is used. The hybrid concept is as follows: initially, the features are passed to both the individual classifiers, and finally the mean of the classifiers’ output will be considered as the outcome. Here, to enhance the performance of prediction results, the training of both classifiers is carried out by the proposed ASN via tuning the optimal weights.

### 2.2.1. LSTM networks

The most popular type of recurrent neural networks (RNNs) are LSTM networks. The memory cell and the gates are the two essential parts of the LSTM. The input gates and forget gates alter the internal elements of the memory cell.

LSTM networks rely on four essential gates. The forget gate ($f$), responsible for determining what information from the previous state should be remembered or discarded. The input gate ($i$) comes into play, deciding which incoming data should be integrated into the current state. The input modulation gate ($g$), often considered as part of the input gate, alters incoming data to ensure its appropriateness for updating the internal state. Finally, the output gate ($o$) combines various outputs, including the previous state, to generate the current state. Together, these four gates orchestrate the flow of information, control state updates, and contribute to the network’s output.

In our work, we have employed the tanh activation function, denoted in (4). The use of the tanh activation function is pivotal in our neural network framework, introducing essential non-linearity that aids in capturing intricate data relationships. This function’s significance lies in its widespread use across various neural network architectures, contributing to tasks like feature transformation and classification.

$$H_t = \tanh(W_{Ht}H_{t-1} + W_{xt}X_t) \quad (4)$$

To reduce the model’s loss, we employed the following cross-entropy loss function in our work, as in (5).

$$cross\_\text{Ent} = \frac{1}{N} \sum_{i=1}^{N} [t_i \log(\text{sigmoid}(\gamma_i)) + (1 - t_i) \log(\text{sigmoid}(1 - \gamma_i))] \quad (5)$$

### 2.2.2. Restricted Boltzmann machines

RBM layers, which were used for pre-training, were transformed into a feed-forward network to enable weight fine-tuning by using a new strategy. A SoftMax layer was added to the top layer during the fine-
A novel optimized deep learning method for protein-protein prediction in bioinformatics (Preeti Thareja)

2.3. ASN optimizer

The proposed ASN is the combination of Aquila optimizer and shark nose smell optimization. Aquila update is influenced by the Shark Nose algorithm. Normally, the hybrid concept of optimization ensures better convergence rate and speed rather than executing as the individual algorithms. The objective function of our proposed ASN-based PPI prediction model is provided in further subsections.

2.3.1. Objective function

The input for our proposed ASN-based method is visually represented in Figure 2. This figure serves as a pivotal element in conveying the data and information that are crucial for the successful implementation of our approach. Figure 2 provides a clear and concise visualization of the solution input, which can include various data sources, parameters, or components, depending on the context of the method.

\[
\begin{array}{cccccc}
W1 & W2 & \ldots & WN & U1 & U2 & \ldots & UN \\
\text{LSTM} & & & & \text{RBM}s & \\
\end{array} \Rightarrow Y
\]

Figure 2. Solution encoding of proposed ASN-based PPI prediction technique

The primary objective of this work was centered around the minimization of the mean square error, as described in (6). Mean square error (MSE) is a fundamental metric used in various fields, particularly in the context of optimization problems and statistical analysis. In this work, (6) serves as the key representation of the objective function, which outlines the specific mathematical criterion for minimizing the discrepancies between predicted and observed values.

\[
Obj = \min (MSE) \tag{6}
\]

Where, the MSE denotes the mean square error.

2.3.2. Aquila optimizer

The Aquila optimizer is a nature-inspired algorithm based on Aquila bird hunting behavior, primarily used for optimization tasks. It may exhibit slower convergence and suboptimal results in complex optimization tasks. Aquila birds, like true eagles, build their nests in high places, use speed and talons for hunting, with ground squirrels being their common prey.

The four hunting methods used by the bird are described as: ii) expanded exploration, in which the target is pursued (high soar with vertical scoop); ii) narrowed exploration, which is the preferred technique for pursuing ground creatures like snakes and squirrels (contour flight with short glide attack); iii) expanded exploitation, which is the technique for pursuing slow prey (low flight with a slow descending attack); and iv) narrowed exploitation is a technique for hunting huge animals (walking and grabbing the target). The mathematical expressions for the methods are expressed in (7)-(11).

\[
X_1(t+1) = X_{best}(t) \times \left( 1 - \frac{1}{t} \right) + (X_M(t) - X_{best}(t) \times rand) \tag{7}
\]

\[
X_M(t) = \frac{1}{N} \sum_{i=1}^{N} X_i(t), \forall j = 1,2,\ldots,Dim \tag{8}
\]

\[
X_2(t+1) = X_{best}(t) \times Levy(D) + (X_R(t) + (y - x) \times rand) \tag{9}
\]

\[
X_3(t+1) = (X_{best}(t) \times X_M(t)) \times a - rand + ((UB - LB) \times rand + LB) \times \delta \tag{10}
\]

\[
X_4(t+1) = QF \times X_{best}(t) - (G_1 \times X(t) \times rand) - G_2 \times Levy(D) + rand \times G_1 \tag{11}
\]

\[
QF(t) = \frac{2 \times rand (\cdot) - 1}{(1 - \gamma)^2} \tag{12}
\]

\[
G_1 = 2 \times rand (\cdot) - 1 \tag{13}
\]
\[ G_2 = 2 \times (1 - \frac{t}{T}) \]  

(14)

where \( X_1, X_2, \) and \( X_3 \) represent the new solution for methods 1, 2, and 3, \( X_{best} \) is the best solution, \( t \) is the current iteration, \( T \) is the maximum iteration, \( N \) is the population size, \( Dim \) is the variable size, \( rand \) is the random value in the range 0 to 1, and \( X_m \) is the local mean value, as in (8). Levy (\( D \)) is Levy’s flight distribution, \( UB \) is the upper bound, \( LB \) is the lower bound, \( \alpha, \delta \) are exploitation parameters, and \( QF, G_1, \) and \( G_2 \) are quality factors as shown in (12)-(14).

2.3.3. Shark nose optimizer

Shark nose optimization algorithm is a population-based metaheuristic optimization algorithm. Shark nose optimization algorithm is inspired by the Shark food foraging behavior. The entire algorithm is based on calculating the shark’s position based on the movements of the shark which are: i) forward movement and ii) rotational movement. The mathematical expression for the movements is expressed in (15)-(17).

\[
Y_{i}^{k+1} = X_{i}^{k} + V_{i}^{k} \times \Delta t_k, i = 1,2, ..., newposition, k = 1,2, ..., k_{max} \]  

(15)

\[
Z_{i}^{k+1,m} = Y_{i}^{k+1} + R3 \times Y_{i}^{k+1}, m = 1,2, ..., M, i = 1,2, ..., newposition, k = 1,2, ..., k_{max} \]  

(16)

The shark’s new position is determined using the expression as shown in (17).

\[
X_{i}^{k+1} = \text{argmax}(of\( Y_{i}^{k+1} \), of\( Z_{i}^{k+1,1} \), ..., of\( Z_{i}^{k+1,m} \)), i = 1,2, ..., newposition \]  

(17)

Gauss mutation was also performed in our work to provide an accurate and reliable optimization. To make a new generation, Gaussian mutation simply adds a random value from a Gaussian distribution to every member of an individual’s vector. The pseudocode for the proposed algorithm is described in Table 2.

<table>
<thead>
<tr>
<th>Step Number</th>
<th>Step Name</th>
<th>Step procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Initialization</td>
<td>Set the attributes new position, ( k_{max}, \alpha, \delta ). Create an initial population. Create every decision randomly within the acceptable range. Initializing the stage counter ( k = 1 ) for ( k = 1 ) to ( k_{max} )</td>
</tr>
<tr>
<td>2</td>
<td>Forward movement</td>
<td>Compute velocity vector using for every element. Acquire a new location of the shark depending on its forward movement, using the Aquila updating function.</td>
</tr>
<tr>
<td>3</td>
<td>Rotational movement</td>
<td>Depending on the rotational movement, acquire the new location of the shark. Depending on the two moves, choose the shark's upcoming location.</td>
</tr>
<tr>
<td>4</td>
<td>Gaussian mutation</td>
<td>Apply Gaussian mutation to increase the local search ability. End for ( k ) Set ( k = k + 1 ) Choose the shark position with the greatest value in the final stage.</td>
</tr>
</tbody>
</table>

3. RESULTS AND DISCUSSION

3.1. Simulation setup

The proposed work has been implemented in the MATLAB tool. The datasets are typical UniProt proteins with experimental gene ontology (GO) annotation and structure models predicted by I-TASSER. Performance matrices of our proposed ASN PPI prediction technique were evaluated and compared with conventional techniques such as Aquila, cat swarm optimization, hunger games search, poor rich optimization, and shark nose optimization.

3.2. Error analysis

The error analysis in this study encompasses several performance metrics to evaluate the model’s accuracy. These metrics include mean absolute error (MAE), measuring the absolute size of discrepancies between actual and predicted values, root mean square error (RMSE), assessing the overall magnitude of errors, mean absolute relative error (MARE), evaluating prediction accuracy in relation to relative errors, and mean squared error (MSE), which calculates the average of squared differences between predicted values and the overall mean, offering insights into prediction variability. These metrics collectively provide a comprehensive assessment of the model’s predictive capabilities and its ability to minimize errors across a range of contexts.
A novel optimized deep learning method for protein-protein prediction in bioinformatics (Preeti Thareja)

Our proposed ASN approach was compared to traditional optimization techniques, including cat swarm optimization (CSO), hunger games search (HGS), and poor rich optimization (PRO), using various evaluation metrics. For dataset-1, our approach achieved a lower MAE of 0.013 at 60% learning percentage (LP) compared to PRO (0.017) and CSO (0.014). Additionally, our method demonstrated a MARE value of 1 for 60% and 70% LPs, highlighting its effectiveness. In contrast, HGS resulted in higher MSE and RMSE values of 0.043 and 0.21, respectively, at 60% LP, indicating that our ASN approach is more reliable and outperforms traditional methods. Figure 3 shows the comparison for ASN PPI prediction model with traditional models when applied to dataset 1 giving results for MAE in Figure 3(a), MSE in Figure 3(b), MARE in Figure 3(c) and RMSE in Figure 3(d).

For dataset-2, we compared our ASN approach to traditional optimization algorithms, assessing metrics like MAE, MARE, MASE, and RMSE. Figure 4 shows the comparison for ASN PPI prediction model with traditional models when applied to dataset 2 giving results for MAE in Figure 4(a), MSE in Figure 4(b), MARE in Figure 4(c) and RMSE in Figure 4(d). Notably, at 60-90% LPs, our approach achieves lower MAE values (0.17, 0.18, 0.19, and 0.17) compared to CSO (0.180, 0.183, 0.194, and 0.195). Similarly, our MARE values for dataset-2 are consistently lower (1.7, 1.3, 1.8, and 1.5) across LPs, showcasing the effectiveness of our ASN technique for PPI prediction. In contrast, both HGS and PRO techniques yield higher MAE and MARE values, underlining the superior performance of our proposed prediction strategy over traditional methods.

Figure 5 serves as a visual representation of the performance comparison of the proposed ASN-based prediction strategy with other alternative networks across multiple cases and datasets. The figure provides a clear and concise summary of the evaluation results for dataset-1 and dataset-2. It is divided into two sub-figures, Figures 5(a) and 5(b), each focusing on a specific dataset.

In Figure 5(a), the performance results for dataset-1 are presented. The key performance metric, MAE, is highlighted, showing that the ASN-based strategy achieves a low MAE of 0.135. This is contrasted with LSTM, CNN, and SVM, which exhibit significantly higher MARE values of 2.44, 2.99, and 1.96, respectively. The results emphasize the superior performance of the proposed ASN-based strategy in dataset-1.

Figure 5(b) shifts the focus to dataset-2 and provides a comprehensive examination of performance metrics, including MAE, RMSE, MARE, and MSE. The ASN-based approach in dataset-2 demonstrates MAE and RMSE values of 0.173 and 0.228, respectively. In contrast, alternative networks like LSTM, CNN, and...
SVM yield higher MAE and RMSE values, further highlighting the superior performance of the ASN-based strategy in this dataset.

![Graphs comparing MAE, MSE, MARE, and RMSE for different learning percentages](image_url)

**Figure 4.** Comparing results of (a) MAE, (b) MSE, (c) MARE, and (d) RMSE of proposed ASN PPI prediction model with standard optimization algorithms for dataset 2.

![Graphs comparing MAE, MSE, MARE, and RMSE for datasets 1 and 2](image_url)

**Figure 5.** Comparing results of MAE, MSE, MARE, and RMSE of the proposed ASN model with standard optimization algorithms for (a) dataset 1 and (b) dataset 2.

### 3.3. Accuracy analysis

The performance of the projected model is evaluated for dataset 1 and dataset 2 by considering various learning percentages such as 60, 70, 80, and 90 respectively. As per the obtained results, the projected model has attained the highest accuracy over the conventional models for different learning percentages. The obtained results are illustrated in Tables 3 and 4.

At a 60% learning percentage in dataset 1, the developed model achieves an impressive accuracy of approximately 87.37%, surpassing traditional methods such as CSO, HGS, and PRO. Additionally, in dataset 2, at a 70% learning percentage, the developed model consistently attains the highest accuracy among the alternatives. These results underscore the model’s robust performance and its superiority over traditional methods in delivering accurate outcomes across various datasets and learning percentages.
Table 3. Comparison of accuracy of the proposed ASN approach with traditional optimization algorithms for dataset 1

<table>
<thead>
<tr>
<th>Method</th>
<th>60%</th>
<th>70%</th>
<th>80%</th>
<th>90%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cat Swarm</td>
<td>84</td>
<td>85.6</td>
<td>84</td>
<td>83.9</td>
</tr>
<tr>
<td>Hunger Games</td>
<td>86.4</td>
<td>85.8</td>
<td>85</td>
<td>84</td>
</tr>
<tr>
<td>Poor Rich</td>
<td>84.9</td>
<td>85.8</td>
<td>86</td>
<td>84.5</td>
</tr>
<tr>
<td>Proposed ASN</td>
<td>87.37</td>
<td>86.6</td>
<td>86.5</td>
<td>87</td>
</tr>
</tbody>
</table>

Table 4. Comparison of accuracy of the proposed ASN approach with traditional optimization algorithms for dataset 2

<table>
<thead>
<tr>
<th>Method</th>
<th>60%</th>
<th>70%</th>
<th>80%</th>
<th>90%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cat Swarm</td>
<td>84</td>
<td>85.4</td>
<td>83.9</td>
<td>83.8</td>
</tr>
<tr>
<td>Hunger Games</td>
<td>85</td>
<td>86</td>
<td>85</td>
<td>84</td>
</tr>
<tr>
<td>Poor Rich</td>
<td>85.9</td>
<td>86</td>
<td>85.9</td>
<td>84.9</td>
</tr>
<tr>
<td>Proposed ASN</td>
<td>86.7</td>
<td>87.5</td>
<td>86.5</td>
<td>87</td>
</tr>
</tbody>
</table>

4. CONCLUSION

The current research work has emphasized predicting the protein-to-protein interaction by using sequence-based features and optimized classifiers. Different physicochemical properties have different effects on the classification of AAs in protein sequences. The classification criteria for AAs based on their physicochemical properties is difficult to choose. This is also the direction of our efforts in the future. In addition, the proposed machine learning approach has distinctive inherent biases, including representation biases and process biases, which affect their learning behaviors and performances significantly even in the same learning task. In the future, we will develop an ensemble meta-learning strategy to overcome these issues and it will extendable to other domains. And also, we would employ another sequence-based model with an advanced deep-learning concept. In addition, the most effective optimization approach can be developed for extending the current method.

REFERENCES


A novel optimized deep learning method for protein-protein prediction in bioinformatics (Preeti Thareja)


BIOGRAPHIES OF AUTHORS

Preeti Thareja is a computer science research scholar at Maharshi Dayanand University in Rohtak, Haryana, India. Data mining, artificial intelligence, soft computing, and deep learning are among her research interests. Over the last few years, she has published 3 journal papers, 4 conference papers, and 1 book chapter, as well as two books about Python and soft computing. She can be contacted on preetithareja10@gmail.com.

Rajender Singh Chhillar is a computer science professor at Maharshi Dayanand University in Rohtak, Haryana, India. He was also the head of the Department of Computer Science, the chairman of a board of studies, and a member of the executive and academic councils. Software engineering, software testing, software metrics, web metrics, biometrics, data warehouse and data mining, computer networking, and software design are among his research interests. Over the last several years, he has produced over 91 journal papers and 65 conference papers, as well as two books about software engineering and information technology. He is a director of the CMAI Asia Association in New Delhi, as well as a senior member of the IACSIT in Singapore and a member of the Computer Society of India. He can be contacted on r.chhillar@mdurohtak.ac.in.