Send Orders for Reprints to reprints@benthamscience.net

RESEARCH ARTICLE

HumDLoc: Human Protein Subcellular Localization Prediction Using Deep Neural Network

Rahul Semwal[1] and Pritish Kumar Varadwaj[2,*]

[1]*Department of Information Technology (Bioinformatics), Indian Institute of Information Technology-Allahabad, Jhalwa, Prayagraj, India;* [2]*Department of Bioinformatics and Applied Science, Indian Institute of Information Technology-Allahabad, Jhalwa, Prayagraj, India*

Abstract: ***Aims*****:** To develop a tool that can annotate subcellular localization of human proteins.

Background**:** With the progression of high throughput human proteomics projects, an enormous amount of protein sequence data has been discovered in the recent past. All these raw sequence data require precise mapping and annotation for their respective biological role and functional attributes. The functional characteristics of protein molecules are highly dependent on the subcellular localization/compartment. Therefore, a fully automated and reliable protein subcellular localization prediction system would be very useful for current proteomic research.

Objective**:** To develop a machine learning-based predictive model that can annotate the subcellular localization of human proteins with high accuracy and precision.

Methods**:** In this study, we used the PSI-CD-HIT homology criterion and utilized the sequence-based features of protein sequences to develop a powerful subcellular localization predictive model. The dataset used to train the HumDLoc model was extracted from a reliable data source, Uniprot knowledge base, which helps the model to generalize on the unseen dataset.

Results**:** The proposed model, HumDLoc, was compared with two of the most widely used techniques: CELLO and DeepLoc, and other machine learning-based tools. The result demonstrated promising predictive performance of HumDLoc model based on various machine learning parameters such as accuracy (≥97.00%), precision (≥0.86), recall (≥0.89), MCC score (≥0.86), ROC curve (0.98 square unit), and precision-recall curve (0.93 square unit).

Conclusion**:** In conclusion, HumDLoc was able to outperform several alternative tools for correctly predicting subcellular localization of human proteins. The HumDLoc has been hosted as a web-based tool at https://bioserver.iiita.ac.in/HumDLoc/.

ARTICLE HISTORY

Received: January 10, 2020
Revised: March 27, 2020
Accepted: March 30, 2020

DOI:
10.2174/1389202921999200528160534

Keywords: Bioinformatics, subcellular localization, machine learning, human protein, deep learning, deep neural network.

1. INTRODUCTION

The biological cell is a complex structural unit with various functionally distinct subcellular compartments/ locations. These subcellular compartments include the cell membrane, cytoplasm, nucleus, endoplasmic reticulum, golgi-apparatus, mitochondria, and extracellular region, each with a defined set of roles. The major role of subcellular localization is to provide a functional environment for proteins [1]. They also affect the function of proteins by controlling the availability and access of partner molecules across different localizations [2]. Since the eukaryotic cell synthesizes more than ten thousand different types of proteins, and each protein performs its optimal function in specific subcellular localization; hence, translocation of these proteins into their respective subcellular locations is very important [3]. It has been reported that, if a protein fails to shift into its respective subcellular location, serious disorders or functional loss can occur [4-9]. Hence, the accurate identification of subcellular localization of a protein is a crucial step for its functional annotation and to decide its role in underlying complex biological processes.

The traditional approach to determine the subcellular localization of protein depends on biochemical experiments such as fluorescence microscopy, electronic microscopy, and cell separation methods [10]. However, for a single protein, these methods are very labor-intensive and often time-consuming. In today's post-genomic era, given the rate at which protein data is generated, a reliable automated method is required that can precisely predict the subcellular localization of protein molecules [11]. Automating this process with higher accuracy remains a challenging task in computational biology.

*Address correspondence to this author at the Department of Bioinformatics and Applied Science, Indian Institute of Information Technology-Allahabad, Jhalwa, Prayagraj, India; Tel/Fax: +91-945264599;
E-mail: pritish@iiita.ac.in

There are several computational approaches for predicting the subcellular localization of proteins; the techniques could be categorized as follows:

Composition based method: These methods use classical machine learning approaches, such as Artificial Neural Network (ANN) [8, 12, 13], and Support Vector Machine (SVM) [14-21] to predict the subcellular localization of protein. The methods included in this category are CELLO [19, 20], and P-CLASSIFIER [22], which use amino-acid composition based features to predict subcellular localization of proteins.

Integrated method: These methods also make use of machine learning approaches, but include several other structural and functional attributes of protein as their feature space to predict the subcellular localization. The methods included in this category are PSLpred [22], PSORTb [23, 24] and, EvoStruct-Sub [25], which integrates various analytical protein characteristics as their feature space, demonstrating that integrated approaches perform better than individual feature approaches.

Homology based method: These methods are based on the assumption that there exist several conserved traits among protein sequences belonging to specific subcellular localization. These techniques make an effort to address the relationship between the subcellular localization and evolutionary information by using a sequence-based similarity profile. The methods included in this category are homology-based method [26], domain projection [27-29], and phylogenetic profiling [30, 31].

However, there exist various challenges in such subcellular localization prediction models, which can be summarized as follows: i) the composition-based and homology-based methods degrade their performance if homologous sequences are not captured while training the model parameters. ii) there is a possibility that highly-homologous sequences share the same structure or function; however, it is not necessary that they belong to the same subcellular localization. This results in wrong classifier training, preventing it from correctly annotating the subcellular localization of proteins. iii) composition-based approach is further limited to amino acid composition based features, which are essentially not capable of capturing other important features for the prediction. Meanwhile, the integrated-based method tries to incorporate various features but suffers from overfitting problems. It is also difficult to determine all the crucial set of features related to a specific subcellular compartment. Finally, the above said subcellular localization prediction models use redundant training sets, which overestimate the prediction performance. It often results in poor performance with a significantly lower accuracy score, while redundant sequences were removed from the training set. Furthermore, due to unconscious bias in feature selection as well as due to human error, the classifier was splashed into local minima and mis-predicted the query point. This makes the machine learning model less reliable among the biologists. However, the situation changed after the emergence of deep neural networks [32, 33]. These networks are capable of adjusting the connection weights based on feature importance for classification [11].

In this study, a DNN based approach, HumDLoc, is introduced to predict the subcellular localization of proteins. Utilizing DNN with several hidden layers as compared to conventional ANN enables HumDLoc to capture robust features from the training dataset, *i.e*., the fully connected initial layers learn simplex features and then forward these features to subsequent layers to learn complex features for classification [34]. HumDLoc was also compared with other classification approaches such as K-nearest neighbor (K-NN), Gaussian Naive-Bayes (GNB), SVM, Random Forests (RF), and other existing tools. The comparison result suggests that HumDLoc outperforms the other approaches.

To develop a reliable classification model, HumDLoc followed Chou's 5-step rules [35]. The five key points summarized by Chou to develop a reliable model are as follows: i) Design a valid dataset from a reliable source. ii) Formulate the dataset sample in such a way that it can truly represent a correlation with a predicted target. iii) Develop an algorithm that can automate the prediction process. iv) Perform Cross-fold validation to evaluate the performance of the developed model. v) Provide a user-friendly server to the public. In our study, the five steps corresponding to Chou's five-step rules are as follows: i) we design dataset from UniProt, which is a reliable database for protein sequences. ii) To represent sequence information in the form of machine learning trainable units, we convert them into a feature vector, which correlates with predicted targets. iii) To perform a reliable classification, we develop a DNN based HumDLoc model. iv) Cross-validation is performed to evaluate the performance of the model. v) To demonstrate our findings, the HumDLoc tool was implemented and hosted as a user friendly and publicly accessible web prediction server, which is available on server interface https://bioserver.iiita.ac.in/HumDLoc/.

2. MATERIALS AND METHODS

2.1. Dataset Collection and Preprocessing

The prediction quality of any machine learning model is highly dependent on the reliability of the training dataset. In our study, protein sequences were extracted from Uniprot Knowledgebase, release 2019_03 [36] for training the HumDLoc model. The Uniprot Knowledgebase consists of two sections: UniProtKB/Swiss-Prot, which contains reviewed and manually annotated protein subcellular localization entries, and UniProtKB/TrEMBL, which contains non-reviewed and automatically annotated protein subcellular localization entries. To extract reliable protein entries from Uniprot knowledgebase, the following filtering criteria were used: Organism: Homo sapiens, Sequence: not a fragment (C-terminal or N-terminal should be absent), longer than 30 amino acids, does not contain non-amino acid character, manually annotated and reviewed. To increase the number of protein sequences in each localization/compartment, similar subcellular localization or subclasses of the same subcellular localization were mapped to 7 main subcellular compartments. Moreover, protein sequences were labeled as a membrane or soluble if they were found either in the membrane or the lumen of the organelle; if no information was available for the protein sequence, then they were filtered out. Protein sequences with more than one subcellular localization were also removed. After the filtering process, a total of 4,418 human protein sequences with unique subcellular localizations were obtained. The number of protein sequences and the mapped subclass of each main subcellular location is summarized in Table **1**.

Table 1. Summary of human protein subcellular localization.

S. No.	Subcellular Localization/Compartment	Number of Proteins	Subclass
1	Nucleus	1251	Nucleus, Nucleus Matrix, Nucleolus, Nucleus lamina, Nucleus envelope, Nucleus speckle, Nucleus inner membrane, Nucleus Outer Membrane, Nucleus Membrane, Chromosome.
2	Cell Membrane	1045	Cell Membrane, Apical cell membrane, Basal cell membrane, Basolateral cell membrane, Lateral cell membrane, Cell Projection, lamellipodium, axon, dendrite, filopodium, Phagocytic cup.
3	Cytoplasm	764	Cytoplasm, Microtubule, Stress fiber, Spindle, myofibril, Spindle Pole, Centrosome, Cytoskeleton, Cytosol, sarcomere, A band, M line, H zone, Z line, I band, microtubule organizing center.
4	Extracellular	578	Extracellular, Secreted, Extracellular Space, Extracellular Matrix, Basement membrane, surface film, Interphotoreceptor matrix.
5	Mitochondrion	456	Mitochondrion, Mitochondrion outer membrane, Mitochondrion Matrix, mitochondrion nucleoid, Mitochondrion Membrane, Mitochondrion Inner Membrane, Mitochondrion intermembrane space.
6	Endoplasmic Reticulum	230	Endoplasmic Reticulum, Endoplasmic reticulum Membrane, Sarcoplasmic Reticulum, Microsome, Endoplasmic Reticulum Lumen, Microsome membrane, Rough endoplasmic reticulum, Rough endoplasmic reticulum lumen, Smooth endoplasmic reticulum membrane.
7	Golgi Apparatus	94	Golgi Apparatus, Golgi Network, Golgi Apparatus Lumen, Golgi apparatus Membrane, Cis Golgi network, cis-Golgi network membrane, Golgi stack membrane, trans-Golgi network membrane, Golgi stack trans-Golgi network.

After the mapping process, a stringent homology partition was performed on said above-mapped dataset, which will ensure that HumDLoc generalizes on the new dataset. To do this, PSI-CD-HIT [37] tool was used, which cluster homologous proteins based on certain constraints. The constraints used in this study can be summarized as follows: if proteins are at least 30% identical, and if alignment covers at least 80% of the shorter sequence, then proteins are mapped onto the same cluster, otherwise on different clusters. This clustering process produced 3079 clusters for the whole dataset. Following this step, proteins from each cluster were mapped to one of three folds (train fold, validate fold, and test fold) in such a way that all folds contain distinct sets of protein.

2.2. Feature Calculation

Protein sequences were converted into numerical feature vectors by "*protr"* [38], a library package, from the R software environment [39]. The calculated features include conjoint triad, pseudo amino acid composition, CTD (Composition, Transition, and Distribution), quasi sequence order features, autocorrelation, and amino acid composition. The descriptions related to features are summarized in Table **2**.

2.3. Supervised Machine Learning Algorithm

2.3.1. HumDLoc

HumDLoc is based on DNN with multiple hidden layers for classification, unlike conventional ANN [40]. Fig. (**1**) depicts the architecture of the HumDLoc training kernel. The initial input layer contains 600 self-learning units called neurons. The initial layer receives protein as an input, represented by [1 X 9920] feature vectors, and projects them onto 600 neurons of the first hidden layer to learn the simplex features for classification. These simplex/learned features are used as input for the batch normalization layer, which avoids the problems caused due to the internal covariate shift [41]. However, since every feature is not important for classification, a dropout layer is used to randomly drop some neuron connection before transferring the learned simplex feature to the next layer. The same procedure was used for subsequent hidden layers with 400 and 200 neurons, respectively. Finally, the last layer contains seven output neurons, which corresponds to seven subcellular localization.

Table 2. List of protein sequence features calculated using 'protr'.

Sl. No.	Feature Group	Feature Name	Dimension
1	Conjoint Triad	Conjoint Triad	343
2	Pseudo-Amino Acid Composition	Pseudo-Amino Acid Composition	50
		Amphiphilic Pseudo-Amino Acid Composition	80
3	CTD	Composition	21
		Transition	21
		Distribution	105
4	Quasi-Sequence-Order	Sequence-Order-Coupling Number	60
		Quasi-Sequence-Order Descriptors	100
5	Autocorrelation	Moran Autocorrelation	240
		Geary Autocorrelation	240
		Normalized Moreau-Broto Autocorrelation	240
6	Amino Acid Composition	Amino Acid Composition	20
		Dipeptide Composition	400
		Tripeptide Composition	8000

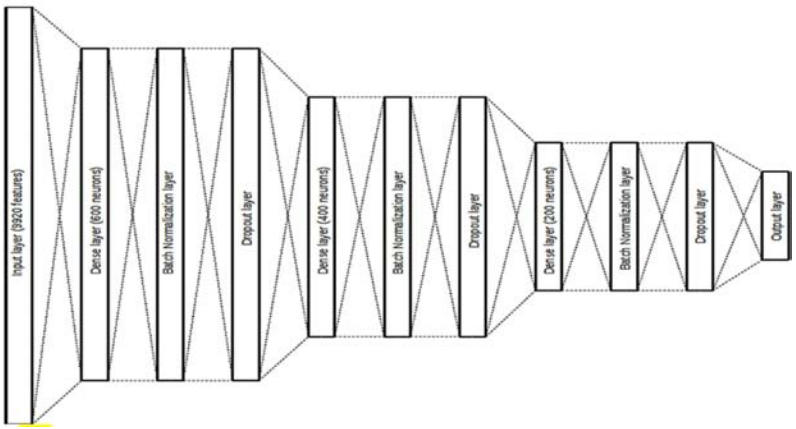

Fig. (1). Architectural view of the HumDLoc model. The vertical rectangular bars represent the layer of HumDLoc model, and dotted lines between the two layers represent a fully connected connection.

A deep neural network model for predicting the subcellular localization can be defined by equation (1).

$$\hat{y} = f_p(X) \quad (1)$$

Where $\hat{y}$ represents the predicted subcellular localization, f_p represents learned function with hyperparameters p, and X represents the feature vector of protein under consideration.

In general, single learnable units or neurons in the hidden layers are fully connected to the previous layer of neurons. However, in a large network, these fully connected neurons create problems of co-adaptation, *i.e*., if all neurons try to learn their weights together, some neurons had more predictive power than others. To avoid such problems, dropout layers were introduced in the proposed DNN model. Also, with the dropout layers, neurons in each hidden layer become more robust and learn useful features on their own without relying on other neurons [42]. Dropout refers to the act of randomly dropping some neurons, along with its incoming and outgoing connection from the network during the training phase of the model based on a dropout rate (hyperparameter), and constructs a new network on which forward and backpropagation is applied. Thus, training a dropout neural network is similar to training a collection of 2^m neural networks, where m is the number of neurons. But, during the testing phase, the idea is to use single neural networks without the dropout layer. To do so, if a neuron is present with probability p during the training phase, the outgoing weight of that neuron is multiplied with probability p in the testing phase. This ensures that each individual neuron will learn its own useful features for classification and avoid the problem of overfitting of the training dataset.

To overcome the problem related to internal covariate shift, a batch normalization layer was introduced in a deep neural network. The internal covariate shift creates a problem in classification during the training phase of DNN, due to the differential distribution of training dataset in hidden layer neurons. The batch normalization layer receives the output of the previous layer and normalizes it before forwarding it to the next layer. The other parameters used to train HumDLoc, known as "adam" optimizer (adaptive learning rate optimizer) [43], was used to minimize the loss function. Although several optimizers have been proposed in the literature [44-46] due to different effect of chosen scalar products [47], we used "adam" optimizer, since the purpose of this optimizer is to provide different learning rate for neurons based on the classification outcome. At the output layer, the softmax activation function was used, which maps the output between 0 and 1. For other layers, the "relu" activation function was used. However, the drawback of this activation function has been identified in some recent works [48, 49], but the quantitative evaluations have shown the high performance of the "relu" function in the proposed network model with our datasets. To train the HumDLoc model, early stopping criteria was used, with the final model being the one with the lowest validation loss. As the graphical figures provide useful information than the textual information [50], traces of training and validation loss (categorical cross-entropy) and accuracy of HumDLoc are shown in Fig. (**2**).

2.3.2. Other Supervised Techniques

K-NN is an iterative technique in which a query point is assigned to a particular category/class based on the majority of K (hyperparameter) nearest neighbors (minimum Euclidean distance). To select hyperparameter K, different values of K were used (1-99), and the ones with the highest performance were selected. GNB uses a probabilistic approach to decide the category/class of a query point based on the probabilistic value determined through Bayes theorem. The underlying assumptions of GNB are Gaussian distribution of features and feature independence, where a smoothing factor (hyperparameter), α, is used to avoid the zero probability problem during class probability calculation. The smoothing factor (α) is a portion of the largest variance of all features that is added to the variance for calculation stability. To select hyperparameter, α, different values of α were used (1e−20 to 1e−1), and the one with the highest performance was selected. SVM uses maximal margin hyperplane and support vectors (the nearest training data points form maximal margin hyperplane) to predict the fate of a query point. For this, SVM kernels, such as the linear kernel, polynomial kernel, sigmoid kernel, and radial basis kernel, were used. RF is an ensemble classification model that uses multiple decision trees for classification. The hyperparameter in RF is the number of decision trees or base models. To select the hyperparameter, *i.e*., the number of the base model (n), different values of n (10 to 200) were used, and the ones with the highest performance were selected.

2.4. Evaluating Criteria

To evaluate the performance of each classifier, different statistical scores were used, as described in [11, 51-54]. In fact, in a typical supervised binary classification problem, each query point from the test sets have their own true class label. However, during the evaluation process, the classifier maps the query points onto one of the following categories: True Positive (TP), True Negative (TN), False Positive (FP), and false negative (FN). To achieve such categories for each class, the multiclass classification uses one *versus* rest approach. In this approach, the query point belongs to a particular class considered a positive or negative point. Based on this, TP, TN, FP, and FN is calculated for each class, then the following statistical scores are used to evaluate the performance of classifier correspond to each class:

a) Accuracy (ACC): It is the measure of correct prediction out of total predictions.

$$ACC = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (2)$$

b) Precision (PPV): It can be defined as the ability of a classifier to correctly predict only relevant data, and is calculated as the ratio between predicted true positive (TP) to all predicted positive observations (TP+FP).

$$PPV = \frac{TP}{(TP + FP)} \quad (3)$$

c) Recall/Sensitivity (SEN): It can be defined as the ability of a classifier to correctly predict all relevant data, and can be calculated as the ratio between predicted true positive (TP) to all positive observations (TP+FN).

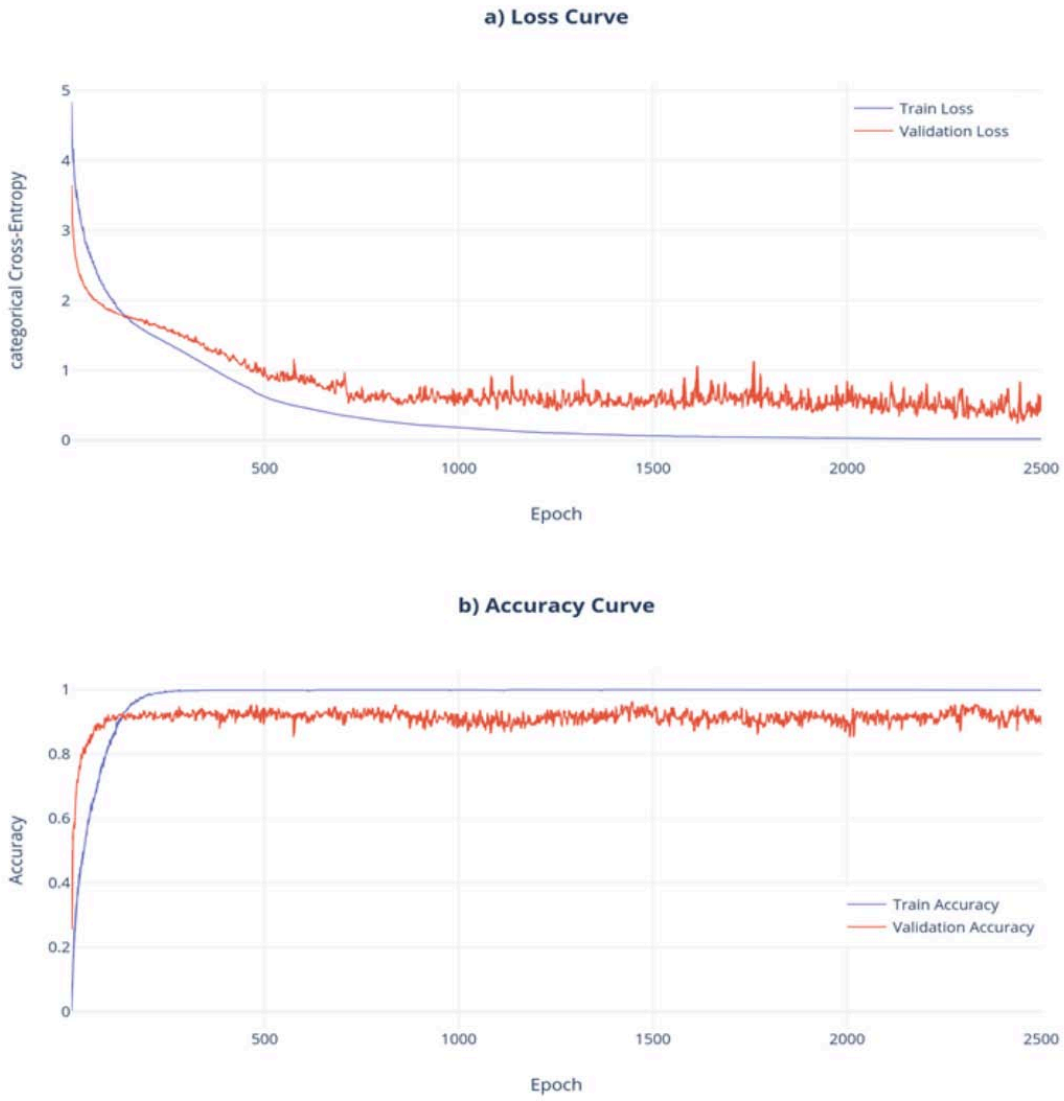

Fig. (2). a) Tracing of training and validation loss **b**) Tracing of training and validation accuracy during the training of HumDLoc. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

$$SEN = \frac{(TP)}{(TP + FN)} \quad (4)$$

d) F1-score (F1): The F1-score uses only three categories (TP, FP, and FN) to evaluate the performance of the classifier. It is the weighted average of PPV and SEN, and takes values between 0 and 1, where zero value represents the worst classifier, and the value one represents the best classifier.

$$F_1 = \frac{2 * (PPV * SEN)}{(PPV + SEN)} \quad (5)$$

e) Matthew's correlation coefficient (MCC): It can be defined as the correlation between the observed and predicted values. The reason behind calculating MCC is that the ACC and F1scores sometimes overestimate the performance of the classifier [55]. Also, the MCC score is considered as balanced statistics to measure classifier performance as it does not effected by class imbalance problems. To calculate MCC, all four categories (TP, TN, FP, and FN) were used, in which the classifier predicts the fate of the query point [11]. The MCC value +1 represents the best prediction, 0 represents random prediction, and -1 represents the disagreement between true class and predicted class.

$$MCC = \frac{(TP + TN) - (FP + FN)}{\sqrt{(TP + FN) * (TP + FP) * (TN + FP) * (TN + FN))}} \quad (6)$$

3. RESULTS AND DISCUSSION

3.1. HumDLoc Result Analysis

To evaluate the performance of the HumDLoc classifier, along with the above-specified evaluation criteria, ROC and Precision-recall analysis of each class were also performed. (Table **3**) was used to represent the different statistical scores of the HumDLoc classifier corresponding to each subcellular compartment. The result showed that all compartments had an accuracy of more than 93 percent, while the accuracy of the golgi-apparatus compartment was 100 percent. However, accuracy alone is not a good measure of classifier perfor-

Table 3. HumDLoc statistical scores correspond to each subcellular localization/compartment.

Performance Measures / Subcellular Localization	Accuracy	Precision	Recall	F1-Score	MCC
Cell Membrane	0.9862	0.99	0.98	0.99	+0.9912
Cytoplasm	0.9358	0.71	0.81	0.76	+0.7104
Endoplasmic Reticulum	0.9862	0.77	1.00	0.87	+0.7717
Golgi-Apparatus	1.00	1.00	1.00	1.00	+1.000
Mitochondria	0.9817	0.87	0.87	0.87	+0.8765
Nucleus	0.9771	0.88	0.92	0.90	+0.8801
Extracellular	0.9679	1.00	0.82	0.90	+0.9810

mance because even a classifier with zero predictive power can get high accuracy, which is known as an accuracy paradox [55-57]. According to this, when the number of false positives are greater than the number of true positives, the accuracy will always increase, where the classifier rule always gives a negative category as the output for all test cases. The same holds true when the number of false negatives are greater than the number of true negatives; in this case, the classifier rule always gives positive class as the output. This implies that a high accuracy model doesn't always mean high performance of the classifier. To avoid such problems, F1-score is used as a less misleading measure [58]. The F1-score of all compartments were greater than 86 percent, which implied that the classifier had high precision and recall. However, to include all four categories (TP, TN, FP, and FN) for evaluating the performance, Matthew's correlation coefficient (MCC) was used. The MCC score of all components were greater than +0.71, while the MCC score of Golgi apparatus was +1.

3.1.1. ROC Analysis (Receiver Operating Characteristics)

ROC is a graphical plot that is used to describe the performance of a system, where a false-positive rate (FPR or 1—specificity) is plotted against the x-axis and true-positive rate (TPR or sensitivity) is plotted against the y-axis [59]. TPR is defined in equation (3), while FPR can be defined as the ratio of the number of negative data points predicted as positive, out of the total negative data points. (Fig. **3a**) was used to represent the plot for the ROC curve of each subcellular compartment, generated using sklearn [60] package 'roc-auc' [61] with different thresholds. The ideal situation for the ROC curve can be depicted by coordinate values (0, 1), corresponding to FPR and TPR, respectively. This implies that tests have sensitivity and specificity equal to 100%. This situation is known as a perfect classification [59]. The diagonal of the ROC plot (coordinate (0, 0) to (0,1)) represents the random classification with sensitivity and specificity equal to 50%. If the ROC curve of a classifier is above the diagonal, then it is considered as a good classifier, *i.e*., more the area under the curve, better a classifier is. Since the problem at hand deals with multiclass classification, the micro average [58] ROC curve for the HumDLoc classifier was also plotted, which was used to represent the average performance of the classifier (ROC-AUC= 0.98 square unit). Micro average ROC was plotted against micro average TPR (TPRμ) and micro average FPR (FPRμ), where micro average TPR and FPR are individual contributions of each subcellular compartment to compute average matrix and is defined in equation (7), and equation (8):

$$TPR_{\mu} = \frac{\sum_{i=1}^{c} TP_i}{\sum_{i=1}^{c}(TP_i + FN_i)} \quad (7)$$

$$FPR_{\mu} = \frac{\sum_{i=1}^{c} FN_i}{\sum_{i=1}^{c}(TP_i + FN_i)} \quad (8)$$

Where *c* represents the total number of subcellular compartments. The area under the curve of each subcellular compartment: cell membrane, cytoplasm, endoplasmic reticulum, golgi-apparatus, mitochondria, nucleus, and extracellular was 0.99 square unit, 0.96 square unit, 0.99 square unit, 1.0 square unit, 0.96square unit, 0.97 square unit, and 0.99 square unit respectively.

3.1.2. Precision-Recall (PR) Analysis

Precision recall curve is another measure used to evaluate the performance of classifiers, where precision is plotted against the y-axis, and recall is plotted against the x-axis. (Fig. **3b**) represents the precision-recall plot of the HumDLoc classifier corresponding to each subcellular compartment. The plot of PR-curve is generated using sklearn package 'precision-recall curve' with different thresholds.

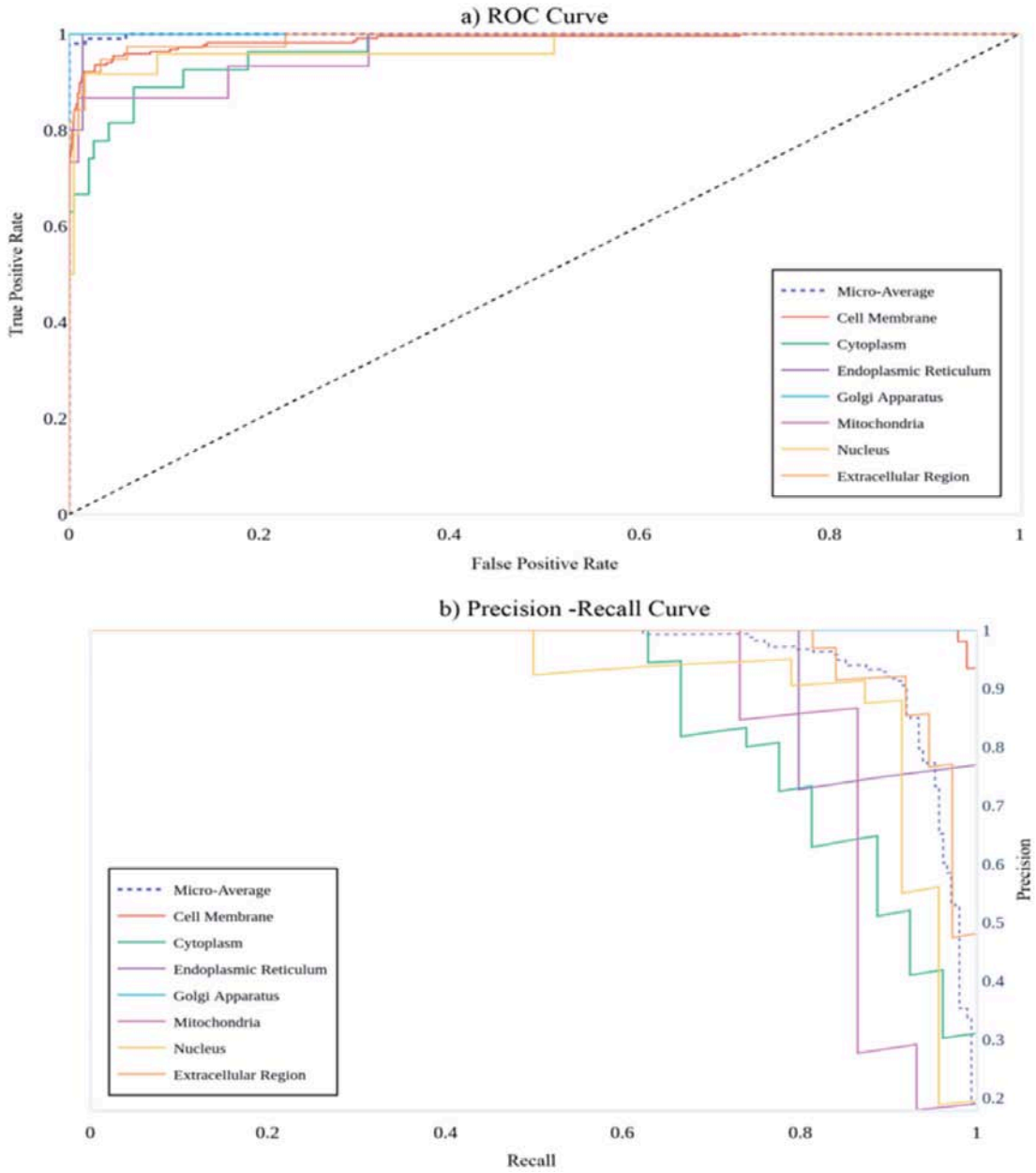

Fig. (3). Performance analysis of HumDLoc against seven subcellular compartments. **a**) Represents the ROC analysis of HumDLoc. **b**) Represents the Precision-Recall analysis of HumDLoc. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

Precision-recall does not consider the true negative samples in its calculation and helps in analyzing the true positive quality of the classifier. In an ideal situation, the value of precision and recall equal to 1; this is the top right corner of the precision-recall curve. This implies the ability of the classifier to perfectly predict all true positive samples without any false positive prediction. In this situation, the area under the precision-recall curve is 1 square unit. However, when the precision is low, and recall is high, then it shows that the classifier is able to predict most of the true samples, as well as, it also predicts false samples as true samples. Similarly, when the precision is high, and recall is low, this means classifiers are able to predict some of the true samples out of all true samples with a less false positive value. In practice, for good classification, there must be a good trade-off between precision and recall. To achieve this criterion, the area under the precision-recall curve must be close to 1 square unit. In Fig. (**3b**), the area under the curve of each subcellular compartment were found to be: cell membrane, cytoplasm, endoplasmic reticulum, golgi-apparatus, mitochondria, nucleus, and extracellular is 0.99 square unit, 0.87 square unit, 0.95 square unit,1.0 square unit, 0.88 square unit,0.91 square unit, and 0.96 square unit respectively.

3.2. Comparison with Other Classification Techniques and Existing Tools

To perform a comparison between HumDLoc, other classification techniques, and existing tools, the above-specified

evaluation criteria were aggregated to get the overall performance of classifiers. For comparison, the two most popular and recently reported tools: DeepLoc [62] and CELLO [19, 63] were selected. DeepLoc uses a combination of convolution neural networks and recurrent neural networks, while CELLO uses the SVM classifier in its core to predict subcellular localization. Both of these methods are used to predict eukaryotic protein subcellular localization. The detailed evaluation criteria of each classifier correspond to the class specified in the Supplementary material (**S1-S9**). The micro averaged evaluation criteria of each classifier were represented in Table **4**.

In terms of micro average accuracy, the performance of HumDLoc (97.64%) was comparable to that of SVM with linear (SVML with accuracy 96.46%), and rbf kernel (SVMR with accuracy 96.07%); whereas the micro-average accuracy of HumDLoc was much higher compared to other classifiers and existing tools. This suggested that the training dataset could be separated by almost linear hyperplanes. However, in multiclass classification, due to the accuracy paradox, one can not only rely on accuracy parameters to evaluate the performance of the prediction system. The micro-average precision of HumDLoc (92%) was lower than the Random Forest classifier (RF with precision 97%) but was much higher than other machine learning techniques and existing tools. This showed that the false positive prediction quality of HumDLoc was slightly higher than that of RF, but much lower than other techniques and tools. However, the micro-average recall of HumDLoc (92%) was exceptionally high when compared to other classification methods. This indicated that the true positive prediction quality of HumDLoc was much better than other tools when applied on the total available dataset. The micro-average F1-score for HumDLoc was quite high (0.92), while other classifiers suffered from lower F1-scores (0.86). For a given dataset, the F1-score represents the ability of HumDLoc to make a clear distinction between true positive samples from false positive and negative ones. Furthermore, the MCC score of HumDLoc was much higher (+0.8873) while compared to other techniques (+0.855).

To perform a unbiased comparison (dataset independent comparison) between HumDLoc and other existing tools (*i.e* CELLO, and DeepLoc), a standard benchmark database repository LocDB [64] was used. The LocDB is an expert-curated protein subcellular localization database, containing localization information related to Homo sapiens (human) and Arabidopsis thaliana (Weed). To perform a comparison between tools, we extracted the data related to Homo sapiens from LocDB. Fig. (**4**) represents the comparison result between HumDLoc and existing tools (*i.e*., CELLO and DeepLoc). In all four evaluation metrics criteria, the HumDLoc tool outperformed the existing tools. The accuracy of HumDLoc was 97% while the accuracy of DeepLoc

Table 4. Micro-averaged comparative statistical scores of classifiers.

Performance Measures / Machine Learning Model	Accuracy	Precision	Recall	F1-Score	MCC
HumDCLoc	**0.9764**	0.92	**0.92**	**0.92**	**+0.8873**
K-NN[a]	0.9004	0.65	0.65	0.65	+0.6500
Naive-Bayes	0.9043	0.67	0.67	0.67	+0.6093
SVML[b]	0.9646	0.88	0.88	0.88	+0.8555
SVMP[c]	0.9581	0.85	0.85	0.85	+0.8287
SVMR[d]	0.9607	0.86	0.86	0.86	+0.8394
SVMS[e]	0.9541	0.84	0.84	0.84	+0.8400
RF[f]	0.9227	0.97	0.67	0.80	+0.6843
DeepLoc	0.9521	0.81	0.81	0.78	+0.7631
CELLO	0.9498	0.80	0.80	0.80	+0.7704

[a]K-nearest neighbour, [b]SVM with linear kernel, [c]SVM with polynomial kernel, [d]SVM with radial basis kernel, [e]SVM with sigmoid kernel, [f]Random Forests.

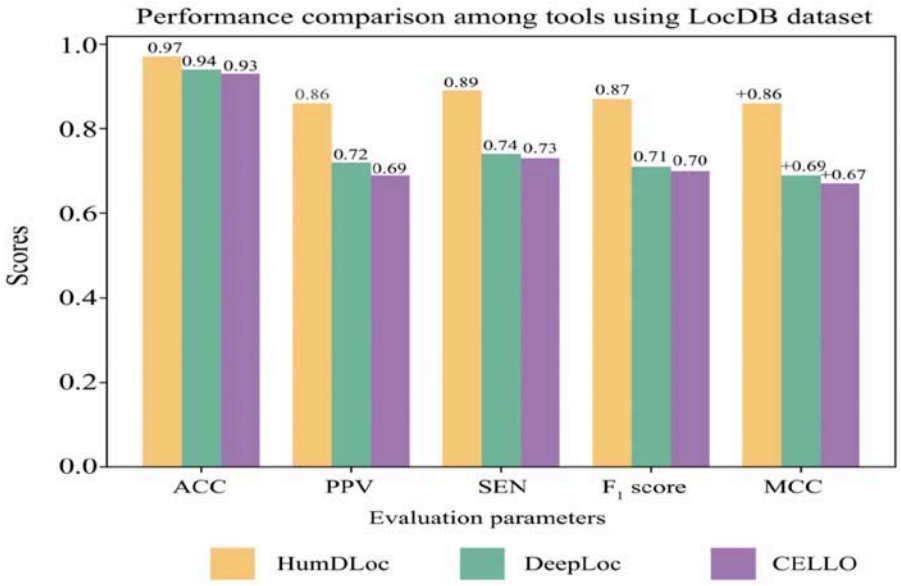

Fig. (4). Performance comparison among HumDLoc, DeepLoc, and CELLO using LocDB dataset. (*A higher resolution / colour version of this figure is available in the electronic copy of the article*).

and CELLO was 94% and 93%respectively. The precision of HumDLoc to capture true data out of all positive predicted data was much higher (86%) than that of DeepLoc (72%) and CELLO (69%). The sensitivity of HumDLoc to predict positive data out of all actual positive data was also much higher (89%) than DeepLoc (74%) and CELLO (73%). Similarly, the F1-score of HumDLoc was much higher (87%) than DeepLoc (71%) and CELLO (70%), showing the ability of the classifier to make clear distinctions between a true positive sample from false positive and negative ones. The MCC score of the HumDLoc classifier was much better (86%) than the DeepLoc (69%) and CELLO (67%), which showed that HumDLoc performed better classification even if the distribution among classes was non-uniform. The detailed evaluation criteria of each classifier correspond to the class specified in the Supplementary material (**S10-S12**).

CONCLUSION

With the advancement in protein sequence discovery and the abundance of raw protein data, there is a requirement for automated tools that can predict subcellular compartmentalization of proteins with high precision and accuracy. Knowledge of subcellular localization helps in deciphering the functional aspect of proteins. In this study, a machine learning-based prediction system, HumDLoc, was developed. This tool can be used to predict human protein subcellular localization into seven major subcellular compartments: cell membrane, cytoplasm, endoplasmic reticulum, golgi-apparatus, mitochondria, nucleus, and extracellular. To predict subcellular localization of the protein, HumDLoc uses various sequence-related features, such as CTD, PseAAc, AAC, *etc*. HumDLoc was compared with other machine learning techniques and existing tools (DeepLoc and CELLO). The average accuracy (97.64%), precision (0.92), recall (0.92), and Matthew's correlation coefficient (+0.92) of HumDLoc was much higher than other machine learning techniques, as well as existing tools. Also, to make an unbiased comparison, a benchmark dataset LocDB had been taken to evaluate the performance of HumDLoc and existing tools (*i.e*., DeepLoc and CELLO). HumDLoc outperformed DeepLoc and CELLO in terms of average accuracy (97.42%), precision (0.86), recall (0.89), and Matthew's correlation coefficient (+0.86). In conclusion, HumDLoc was able to outperform several alternative tools for correctly classifying large datasets obtained from Uniprot Knowledgebase with higher accuracy and precision.

AUTHORS' CONTRIBUTIONS

R.S. and P.K.V conceived and designed the experiment. R.S. performed all computational analyses. All authors wrote the manuscript and approved the final version.

ETHICS APPROVAL AND CONSENT TO PARTICIPATE

Not applicable.

HUMAN AND ANIMAL RIGHTS

No animals/humans were used for studies that are the basis of this research.

CONSENT FOR PUBLICATION

Not applicable.

AVAILABILITY OF DATA AND MATERIALS

The data supporting the findings of the article is available at Uniprot (URL: https://www.uniprot.org/), reference number [Manuscript Ref: 36].

FUNDING

None.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

The authors acknowledge the Department of Bioinformatics & Applied Sciences, Central Computing Facility of IIITA, and Indian Institute of Information Technology-Allahabad for providing the computing facility. The authors are also grateful to Imlimaong Aier for assistance with figure production.

SUPPLEMENTARY MATERIAL

Supplementary material is available on the publisher's website along with the published article.

REFERENCES

[1] Popgeorgiev, N.; Jabbour, L.; Gillet, G. Subcellular localization and dynamics of the Bcl-2 family of proteins. *Front. Cell Dev. Biol.,* **2018**, *6*, 13. http://dx.doi.org/10.3389/fcell.2018.00013 PMID: 29497611

[2] Scott, M.S.; Calafell, S.J.; Thomas, D.Y.; Hallett, M.T. Refining protein subcellular localization. *PLOS Comput. Biol.,* **2005**, *1*(6), e66. http://dx.doi.org/10.1371/journal.pcbi.0010066 PMID: 16322766

[3] D??nnes, P.; H??glund, A. Predicting protein subcellular localization: past, present, and future. *Genomics Proteomics Bioinformatics,* **2004**, *2*(4), 209-215. http://dx.doi.org/10.1016/S1672-0229(04)02027-3 PMID: 15901249

[4] LaQuaglia, M.J.; Grijalva, J.L.; Mueller, K.A.; Perez-Atayde, A.R.; Kim, H.B.; Sadri-Vakili, G.; Vakili, K. YAP subcellular localization and hippo pathway transcriptome analysis in pediatric hepatocellular carcinoma. *Sci. Rep.,* **2016**, *6*, 30238. http://dx.doi.org/10.1038/srep30238 PMID: 27605415

[5] Shurety, W.; Merino-Trigo, A.; Brown, D.; Hume, D. A.; Stow, J. L. Localization and post-Golgi trafficking of tumor necrosis factor-alpha in macrophages. *J. Interferon Cytokine Res.,* **2000**, *20*(4), 427-438. http://dx.doi.org/10.1089/107999000312379

[6] Bryant, D.M.; Stow, J.L. The ins and outs of E-cadherin trafficking. *Trends in Cell Biol.,* **2004**, *14*(8), 427-434.

[7] Cheng, X.; Xiao, X.; Chou, K-C. pLoc-mGneg: Predict subcellular localization of Gram-negative bacterial proteins by deep gene ontology learning *via* general PseAAC. *Genomics,* **2017**, *110*(4), 231-239. http://dx.doi.org/10.1016/j.ygeno.2017.10.002 PMID: 28989035

[8] Hartmann, T.; Bergsdorf, C.; Sandbrink, R.; Tienari, P.J.; Multhaup, G.; Ida, N.; Bieger, S.; Dyrks, T.; Weidemann, A.; Masters, C.L. Alzheimer's disease βA4 protein release and amyloid precursor protein sorting are regulated by alternative splicing. *J. Biol. Chem.,* **1996**, *271*(22), 13208-13214. http://dx.doi.org/10.1074/jbc.271.22.13208

[9] Hadizadeh, M.; Tabatabaiepour, S.N.; Tabatabaiepour, S.Z.; Hosseini N.H.; Mohammadi, M.; Sohrabi, S.M. Genome-wide identification of potential drug target in enterobacteriaceae family: a homology-based method. *Microb. Drug Resist.,* **2018**, *24*(1), 8-17. http://dx.doi.org/10.1089/mdr.2016.0259 PMID: 28520499

[10] Camp, R.L.; Chung, G.G.; Rimm, D.L. Automated subcellular localization and quantification of protein expression in tissue microarrays. *Nat. Med.,* **2002**, *8*(11), 1323-1327. http://dx.doi.org/10.1038/nm791 PMID: 12389040

[11] Kuo-Chen, C. Artificial intelligence (AI) tools constructed *via* the 5-steps rule for predicting post-translational modifications. *Trends Artifi. Intell.,* **2019**, *3*(1), 60-74.

[12] Emanuelsson, O.; Nielsen, H.; Brunak, S.; von Heijne, G. Predicting subcellular localization of proteins based on their N-terminal amino acid sequence. *J. Mol. Biol.,* **2000**, *300*(4), 1005-1016. http://dx.doi.org/10.1006/jmbi.2000.3903 PMID: 10891285

[13] Lin, C.; Zou, Y.; Qin, J.; Liu, X.; Jiang, Y.; Ke, C.; Zou, Q. Hierarchical classification of protein folds using a novel ensemble classifier. *PLoS One,* **2013**, *8*(2), e56499. http://dx.doi.org/10.1371/journal.pone.0056499 PMID: 23437146

[14] Cao, Z.; Pan, X.; Yang, Y.; Huang, Y.; Shen, H-B. The lncLocator: a subcellular localization predictor for long non-coding RNAs based on a stacked ensemble classifier. *Bioinformatics,* **2018**, *34*(13), 2185-2194. http://dx.doi.org/10.1093/bioinformatics/bty085 PMID: 29462250

[15] Hua, S.; Sun, Z. Support vector machine approach for protein subcellular localization prediction. *Bioinformatics,* **2001**, *17*(8), 721-728. http://dx.doi.org/10.1093/bioinformatics/17.8.721 PMID: 11524373

[16] Park, K.J.; Kanehisa, M. Prediction of protein subcellular locations by support vector machines using compositions of amino acids and amino acid pairs. *Bioinformatics,* **2003**, *19*(13), 1656-1663. http://dx.doi.org/10.1093/bioinformatics/btg222 PMID: 12967962

[17] Pierleoni, A.; Martelli, P.L.; Fariselli, P.; Casadio, R. BaCelLo: a balanced subcellular localization predictor. *Bioinformatics,* **2006**, *22*(14), e408-e416. http://dx.doi.org/10.1093/bioinformatics/btl222 PMID: 16873501

[18] Hoglund, A.; Donnes, P.; Blum, T.; Adolph, H.W.; Kohlbacher, O. MultiLoc: prediction of protein subcellular localization using N-terminal targeting sequences, sequence motifs and amino acid composition. *Bioinformatics,* **2006**, *22*(10), 1158-1165. http://dx.doi.org/10.1093/bioinformatics/btl002 PMID: 16428265

[19] Yu, C.S.; Chen, Y.C.; Lu, C.H.; Hwang, J.K. Prediction of protein subcellular localization. *Proteins,* **2006**, *64*(3), 643-651. http://dx.doi.org/10.1002/prot.21018 PMID: 16752418

[20] Yu, C.S.; Lin, C.J.; Hwang, J.K. Predicting subcellular localization of proteins for Gram-negative bacteria by support vector machines based on n-peptide compositions. *Protein Sci.,* **2004**, *13*(5), 1402-1406. http://dx.doi.org/10.1110/ps.03479604 PMID: 15096640

[21] Wang, J.; Sung, W.K.; Krishnan, A.; Li, K.B. Protein subcellular localization prediction for Gram-negative bacteria using amino acid subalphabets and a combination of multiple support vector machines. *BMC Bioinformatics,* **2005**, *6*, 174. http://dx.doi.org/10.1186/1471-2105-6-174 PMID: 16011808

[22] Bhasin, M.; Garg, A.; Raghava, G.P. PSLpred: prediction of subcellular localization of bacterial proteins. *Bioinformatics,* **2005**, *21*(10), 2522-2524. http://dx.doi.org/10.1093/bioinformatics/bti309 PMID: 15699023

[23] Gardy, J.L.; Laird, M.R.; Chen, F.; Rey, S.; Walsh, C.J.; Ester, M.; Brinkman, F.S. PSORTb v.2.0: expanded prediction of bacterial protein subcellular localization and insights gained from comparative proteome analysis. *Bioinformatics,* **2005**, *21*(5), 617-623. http://dx.doi.org/10.1093/bioinformatics/bti057 PMID: 15501914

[24] Gardy, J.L.; Spencer, C.; Wang, K.; Ester, M.; Tusnady, G.E.; Simon, I.; Hua, S.; deFays, K.; Lambert, C.; Nakai, K.; Brinkman, F.S. PSORT-B: Improving protein subcellular localization prediction for Gram-negative bacteria. *Nucleic Acids Res.,* **2003**, *31*(13), 3613-3617. http://dx.doi.org/10.1093/nar/gkg602 PMID: 12824378

[25] Uddin, M.R.; Sharma, A.; Farid, D.M.; Rahman, M.M.; Dehzangi, A.; Shatabda, S. EvoStruct-Sub: an accurate Gram-positive protein subcellular localization predictor using evolutionary and structural features. *J. Theor. Biol.,* **2018**, *443*, 138-146. http://dx.doi.org/10.1016/j.jtbi.2018.02.002 PMID: 29421211

[26] Wan, S.; Mak, M.-W.; Kung, S.-Y. mPLR-Loc: an adaptive decision multi-label classifier based on penalized logistic regression for protein subcellular localization prediction. *Anal. Biochem.,* **2015**, *473*, 14-27. http://dx.doi.org/10.1016/j.ab.2014.10.014 PMID: 25449328

[27] Mott, R.; Schultz, J.; Bork, P.; Ponting, C.P. Predicting protein cellular localization using a domain projection method. *Genome Res.,* **2002**, *12*(8), 1168-1174. http://dx.doi.org/10.1101/gr.96802 PMID: 12176924

[28] Zhou, H.; Yang, Y.; Shen, H.-B. Hum-mPLoc 3.0: prediction enhancement of human protein subcellular localization through modeling the hidden correlations of gene ontology and functional domain features. *Bioinformatics,* **2017**, *33*(6), 843-853. PMID: 27993784

[29] Cozzetto, D.; Minneci, F.; Currant, H.; Jones, D.T. FFPred 3: feature-based function prediction for all Gene Ontology domains. *Sci. Rep.,* **2016**, *6*, 31865. http://dx.doi.org/10.1038/srep31865 PMID: 27561554

[30] Marcotte, E.M.; Xenarios, I.; van Der Bliek, A.M.; Eisenberg, D. Localizing proteins in the cell from their phylogenetic profiles. *Proc. Natl. Acad. Sci. USA,* **2000**, *97*(22), 12115-12120. http://dx.doi.org/10.1073/pnas.220399497 PMID: 11035803

[31] Cheng, Y.; Perocchi, F. ProtPhylo: identification of protein-phenotype and protein-protein functional associations *via* phylogenetic profiling. *Nucleic Acids Res.,* **2015**, *43*(W1), W160-8. http://dx.doi.org/10.1093/nar/gkv455 PMID: 25956654

[32] Goceri, E. *Formulas Behind Deep Learning Success*., In: International Conference on Applied Analysis and Mathematical Modeling (ICAAMM2018), Istanbul, Turkey, **2018**.

[33] Goceri, E.; Gooya, A. *On The Importance of Batch Size for Deep Learning,* **2018**.

[34] Hinton, G.; Deng, L.; Yu, D.; Dahl, G.; Mohamed, A.-R.; Jaitly, N.; Senior, A.; Vanhoucke, V.; Nguyen, P.; Kingsbury, B. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Process. Mag.,* **2012**, *29*, 1-27.

[35] Hussain, W.; Khan, Y.D.; Rasool, N.; Khan, S.A.; Chou, K.-C. SPrenylC-PseAAC: a sequence-based model developed *via* Chou's 5-steps rule and general PseAAC for identifying S-prenylation sites in proteins. *J. Theor. Biol.,* **2019**, *468*, 1-11. http://dx.doi.org/10.1016/j.jtbi.2019.02.007 PMID: 30768975

[36] Apweiler, R.; Bairoch, A.; Wu, C. H.; Barker, W. C.; Boeckmann, B.; Ferro, S.; Gasteiger, E.; Huang, H.; Lopez, R.; Magrane, M. UniProt: the universal protein knowledgebase. *Nucleic Acids Res.,* **2004**, *32*(suppl_1), D115-D119. http://dx.doi.org/10.1093/nar/gkh131

[37] Li, W. Fast program for clustering and comparing large sets of protein or nucleotide sequences. Encyclopedia of Metagenomics: Genes, Genomes and Metagenomes: Basics. *Methods, Databases and Tools*; **2015**, pp. 173-177.

[38] Xiao, N.; Cao, D.-S.; Zhu, M.-F.; Xu, Q.-S. protr/ProtrWeb: R package and web server for generating various numerical representation schemes of protein sequences. *Bioinformatics,* **2015**, *31*(11), 1857-1859. http://dx.doi.org/10.1093/bioinformatics/btv042 PMID: 25619996

[39] Team, R.C. *R: A language and environment for statistical computing*., R Foundation for Statistical Computing. Vienna, Austria, 2013. https://www.R-project.org/

[40] Bengio, Y. Learning deep architectures for AI. *Foundations and Trends® in Machine Learning,* **2009**, *2*(1), 1-127. http://dx.doi.org/10.1561/9781601982957

[41] Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167,* **2015**.

[42] Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.,* **2014**, *15*(1), 1929-1958.

[43] Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980,* **2014**.

[44] Goceri, E. *A Method for Leukocyte Segmentation Using Modified Gram-Schmidt Orthogonalization and Expectation-Maximization*., International Conference on Applied Analysis and Mathematical Modeling ICAAMM18. Istanbul, Turkey, **2018**, p. 18.

[45] Mondal, M.; Semwal, R.; Raj, U.; Aier, I.; Varadwaj, P.K. An entropy-based classification of breast cancerous genes using microarray data. *Neural Comput. Appl.,* **2018**, *1-8*, 1433-3058.

[46] Goceri, E.; Martinez, E. D. A level set method with sobolev gradient and haralick edge detection. *Int. J. Technol.,* **2014**, *5*, 2147-5369.

[47] Goceri, E. *In Effects of chosen scalar products on gradient descent algorithms,* **2015**, 115.

[48] Goceri, E. CapsNet topology to classify tumours from brain images and comparative evaluation. *IET Image Process.,* **2020,** *14*, 882-889.

[49] Goceri, E. Diagnosis of Alzheimer's disease with Sobolev gradient-based optimization and 3D convolutional neural network. *Int. J. Numer. Methods Biomed. Eng.,* **2019**, *35*(7), e3225. http://dx.doi.org/10.1002/cnm.3225 PMID: 31166647

[50] Zhang, S.; Yang, K.; Lei, Y.; Song, K. iRSpot-DTS: Predict recombination spots by incorporating the dinucleotide-based sparse-cross covariance information into Chou's pseudo components. *Genomics,* **2019**, *111*(6), 1760-1770. http://dx.doi.org/10.1016/j.ygeno.2018.11.031 PMID: 30529702

[51] Le, N.Q.; Ou, Y.Y. Prediction of FAD binding sites in electron transport proteins according to efficient radial basis function networks and significant amino acid pairs. *BMC Bioinformatics,* **2016**, *17*(1), 298. http://dx.doi.org/10.1186/s12859-016-1163-x PMID: 27475771

[52] Mohabatkar, H.; Beigi, M.M.; Abdolahi, K.; Mohsenzadeh, S. Prediction of allergenic proteins by means of the concept of Chou's pseudo amino acid composition and a machine learning approach. *Med. Chem.,* **2013**, *9*(1), 133-137. http://dx.doi.org/10.2174/157340613804488341 PMID: 22931491

[53] Le, N.Q.K.; Ho, Q.T.; Ou, Y.Y. Incorporating deep learning with convolutional neural networks and position specific scoring matrices for identifying electron transport proteins. *J. Comput. Chem.,* **2017**, *38*(23), 2000-2006. http://dx.doi.org/10.1002/jcc.24842 PMID: 28643394

[54] Semwal, R.; Aier, I.; Varadwaj, P. K. *PROcket, an Efficient Algorithm to Predict Protein Ligand Binding Site*; Springer, **2019**, pp. 453-461.

[55] Abma, B. *Evaluation of requirements management tools with support for traceability-based change impact analysis.* Master's thesis*,* University of Twente, Enschede, **2009**.

[56] Valverde-Albacete, F.J.; Carrillo-de-Albornoz, J.; Pelaez-Moreno, C. In a proposal for new evaluation metrics and result visualization technique for sentiment analysis tasks. *International Conference of the Cross-Language Evaluation Forum for European Languages,* **2013**, pp. 41-52. http://dx.doi.org/10.1007/978-3-642-40802-1_5

[57] Valverde-Albacete, F.J.; Pelaez-Moreno, C. 100% classification accuracy considered harmful: the normalized information transfer factor explains the accuracy paradox. *PLoS One,* **2014**, *9*(1), e84217. http://dx.doi.org/10.1371/journal.pone.0084217 PMID: 24427282

[58] Van Asch, V. Macro-and micro-averaged evaluation measures [basic draft]. *Belgium: CLiPS,* **2013**, *1*, 27.

[59] Semwal, R.; Aier, I.; Raj, U.; Varadwaj, P.K. Pharmadoop: a tool for pharmacophore searching using Hadoop framework. *Netw. Model. Anal. Health Inform. Bioinform.,* **2017**, *6*(1), 20. http://dx.doi.org/10.1007/s13721-017-0161-x

[60] Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.,* **2011**, *12*, 2825-2830.

[61] Fawcett, T. An introduction to ROC analysis. *Pattern Recognit. Lett.,* **2006**, *27*(8), 861-874. http://dx.doi.org/10.1016/j.patrec.2005.10.010

[62] Almagro Armenteros, J.J.; Sonderby, C.K.; Sonderby, S.K.; Nielsen, H.; Winther, O. DeepLoc: prediction of protein subcellular localization using deep learning. *Bioinformatics,* **2017**, *33*(21), 3387-3395. http://dx.doi.org/10.1093/bioinformatics/btx431 PMID: 29036616

[63] Yu, C.S.; Lin, C.J.; Hwang, J.K. Predicting subcellular localization of proteins for Gram-negative bacteria by support vector machines based on n-peptide compositions. *Protein Sci.,* **2004**, *13*(5), 1402-1406. http://dx.doi.org/10.1110/ps.03479604

[64] Rastogi, S.; Rost, B. LocDB: experimental annotations of localization for *Homo sapiens* and *Arabidopsis thaliana*. *Nucleic Acids Res.,* **2010**, *39*(1), D230-D234.