B.Sc. in Computer Science and Engineering Thesis

Reimplementation of sub-Golgi protein classifiers using large dataset with novel protein sequences

Submitted by

Asif Hossain Rafeen
201305099

Adnan Farooque
201305044

Supervised by

Dr. Mohammad Saifur Rahman

Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology

Dhaka, Bangladesh

February 2020

CANDIDATES’ DECLARATION

This is to certify that the work presented in this thesis, titled, “Reimplementation of sub-Golgi protein classifiers using large dataset with novel protein sequences”, is the outcome of the investigation and research carried out by us under the supervision of Dr. Mohammad Saifur Rahman.

It is also declared that neither this thesis nor any part thereof has been submitted anywhere else for the award of any degree, diploma or other qualifications.

Asif Hossain Rafeen
201305099

Adnan Farooque
201305044

CERTIFICATION

This thesis titled, **"Reimplementation of sub-Golgi protein classifiers using large dataset with novel protein sequences"**, submitted by the group as mentioned below has been accepted as satisfactory in partial fulfillment of the requirements for the degree B.Sc. in Computer Science and Engineering in February 2020.

Group Members:

Asif Hossain Rafeen

Adnan Farooque

Supervisor:

Dr. Mohammad Saifur Rahman
Assistant Professor
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to our thesis supervisor Dr. Mohammad Saifur Rahman for his constant guidance and support. This thesis would not have been possible without his direction and care. His thoughtful counsel in every step of the thesis and his belief in us helped us to complete this thesis.

We would also like to thank Department of Computer Science and Engineering, BUET for providing all the support that we required.

Dhaka
February 2020

Asif Hossain Rafeen

Adnan Farooque

Contents

ABSTRACT

The Golgi Apparatus (GA) is a major collection and dispatch station for numerous proteins destined for secretion, plasma membranes and lysosomes. The dysfunction of GA proteins can result in neurodegenerative diseases. Therefore, accurate identification of protein subGolgi localizations may assist in drug development and understanding the mechanisms of the GA involved in various cellular processes. Some study has been published on identification of subGolgi.Those studies was done on very small amount of dataset.In this paper we focused on collecting new dataset of protein sequence and analyze the results of existing method on larger dataset.we have collected protein sequence from UniprotKB with some specific condition.Then extracted evolutionary features based on existing methods.After that,we implemented accordingly with the paper to produce results.

Chapter 1

Introduction

The Golgi apparatus is an important eukaryotic organelle, which is consisted of a stack of membrane-bounded cisternae located between the endoplasmic reticulum and the cell surface. Many different enzymes and other proteins are retained in Golgi apparatus to perform their various synthetic activities [1] . The cis-Golgi and trans-Golgi apparatus are thought to be specialized cisternae leading proteins in and out of the Golgi apparatus [2] . The function of cis-Golgi proteins is to receive and process the biosynthetic output from endoplasmic reticulum. Then the proteins modified by cis-Golgi proteins are packaged and sent to the required destination by trans-Golgi proteins. Many studies have demonstrated that neurodegenerative diseases, such as amyotrophic lateral sclerosis (ALS) [3], Parkinson's disease [4], and Alzheimer's disease (AD) [5]. The accumulation and aggregation of b-amyloid (Ab) protein is one of the characteristic hallmarks of AD [6] [7]. The Group 9 complexes presented in [8] have great potential as inhibitors of Ab1-40 peptide aggregation that is linked to neurodegeneration in AD patients. Protein S-nitrosylation might represent a potentially viable therapeutic target for a wide range of neurodegenerative diseases [9]. As neuroprotective and anti-inflammatory therapies have largely proved unsatisfactory, considerable effort will be needed to make progress towards effective therapies for neurodegenerative diseases [10]. As demonstrated in [11], dysfunction of Golgi apparatus and its cisternae can give rise to muscular dystrophy, diabetes, cancers and other inheritable diseases. In addition, the GA is considered as an early target of the neurodegenerative diseases [3]. The GA is a major cargo sorting and glycosylation station [12]. Glycans have also been proved to be associated with a number of epidemic diseases such as some inherited diseases, cancers and diabetes. However, the corresponding molecular clues are only just being elucidated [11]. Accurate identification of protein subGolgi localizations could provide useful clues to clarify the contribution of GA dysfunction to diseases, which will significantly impact our ability to develop more effective therapies for diseases and spur further research into the links between glycosylation and disease pathology

Over the last few years, predicting protein subcellular locations by computational methods has become an overwhelming research field in bioinformatics. The subcellular location of a protein is important in understanding its molecular function as well as its role in biological processes. A number of algorithms have been developed in predicting protein subcellular locations in different resolutions and contexts, as reviewed in [13]. The arts of these works reside not only in novel algorithms, but also in sophisticated designed online servers

Many studies, such as protein subcellular location prediction [14–18], tight turn prediction [19], protein quaternary attribute prediction [20,21], enzyme family class prediction [22,23], predicting the cofactors of oxidoreductases [24], identification of membrane proteins and their types [25,26], outer membrane protein prediction [27], identification of GPCR and their types [28, 29], identification of proteases and their types [30, 31], protein cleavage site prediction [32,33], signal peptide prediction [34,35], protein 3D structure prediction based on sequence alignment [36], and sub-organelle location of proteins [37–39] have indicated that sequence-based prediction approaches can timely provide very useful information and insights for both basic research and drug design.

Chapter 2

Related Works

In this chapter we show others work of prediction of Golgi-resident protein.

2.1 Prediction using Feature Selection Technique by Ding et al.

In this work, a support vector machine method was developed to identify the types of Golgi-resident proteins by using only amino acid sequence information [40].

2.1.1 Dataset

After strictly following the some procedures, they obtained 671 proteins including 162 cis-Golgi proteins and 509 trans-Golgi proteins.Then a program named PISCES was used to remove highly similar data to prevent the model giving biased result.As a result, a total of 42 cis-Golgi and 95 trans-Golgi proteins were obtained

2.1.2 Features

In this study Psedo Amino Acid Composition(PseAAC),Amino Acid Composition and n-gap Dipeptide Composition features were used to train the model.

2.1.3 Feature Selection

The original feature set generally contains redundant information or noise which will reduce the predictive accuracy. The ANOVA method can rank the features by measuring the ratio between their variances between groups and within groups.So feature selection was dont by ANOVA program in this study

2.1.4 Prediction Algorithm

A strictly and objective dataset including 137 proteins with the sequence identity <25% was used for training and testing the support vector machine. The analysis of variance was proposed to find out the optimized feature set.The software LibSVM software were utilized and RBF network was used to predict.

2.1.5 Performance

In the leave-one-out cross-validation, the maximum overall accuracy of 85.4% was achieved with the area under the receiver operating characteristic curves of 0.878. The results demonstrate that the proposed method can be used to discriminate the types of Golgi-resident proteins.

2.2 Prediction using Pseudo Amino Acid Compositions by Jiao et al.

In this work, a novel computational method was developed to predict Golgi-resident protein types using positional specific physicochemical properties and analysis of variance based feature selection methods [41].

2.2.1 Dataset

In this study a benchmarking dataset containing 42 cis-Golgi proteins and 95 trans-Golgi proteins was obtained. For the purpose of independent dataset test, 13 cis-Golgi proteins and 51 trans-Golgi proteins were randomly picked up from the UniProt database. There is no overlap between the independent test dataset and the benchmarking dataset.

2.2.2 Features

This study proposed the Positional Specific Physico-Chemical Properties (PSPCP), which integrates the PSSM information within artificially created physicochemical property values.

2.2.3 Feature Selection

This study used ANOVA to remove redundancy.Analysis of variance (ANOVA) is a collection of statistical models and their associated estimation procedures (such as the ”variation” among and between groups) used to analyze the differences among group means in a sample

2.2.4 Prediction Algorithm

In this study,Support Vector Machine with LIBSVM package and an radial basis function (RBF) kernel in this was used as prediction algorithm

2.2.5 Performance

This method achieved 86.9% prediction accuracy in leave-one-out cross-validations with only 59 features. This method has the potential to be applied in predicting a wide range of protein attributes.

2.3 Prediction using Oversampling and Fisher Feature Selection Methods ahmad et al.

In this study,oversampling method named SMOTE is used for the classification of sub-golgi protein with some variety of features [42].

2.3.1 Dataset

The dataset used in this study contains 304 protein sequences, in which 87 sequences are of cis-Golgi class while 217 sequences are of trans-Golgi class, which is constructed by Yang et al., from Universal Protein Knowledgebase (UniprotKB) [43].

2.3.2 Features

In this study, two discrete methods and an evolutionary technique (PSSM) normalized through bigram algorithm are used.

Dipeptide composition

This feature relies on calculation of amino acid residues' frequencies. AAC is very beneficial as it transforms the variable length protein sequences into fixed length feature vector.In order to exploit the occurrence of amino acid residues along with structural information the concept of dipeptide composition (DPC) is adopted. In DPC, the pairing information among the amino acid residues is computed.Particularly 3-gap Dipeptide Composition is used in this study.

Split pseudo amino acid composition (Split-PseAAC)

PseAAC is computed on divided sequences instead of AAC(Amino Acid Composition) since AAC calculates the occurrences of individual amino acids only which lead to loss of important information that may be concealed in the structure of protein sequence i.e. in the order of amino acid residues. PseAAC, on other hand, calculates the frequency of amino acids that constitute a protein sequence as well as it considers the long-range correlation of the physicochemical properties between two residues of the sequence

Bigram position specific scoring matrix

Position Specific Scoring Matrix (PSSM), also known as Position Specific Weight Matrix (PSWM) is commonly used for representation of amino acids structure in a biological sequence which provides evolutionary information for the given protein sequence

2.3.3 Feature Seletion

In this study, **Fisher feature selection** method is used, which is a supervised and filter based method. In this method, selection of individual feature is based on some performance criterion i.e. Fisher score. Fisher score provides subset of features in such a way that in all the selected features, the features which belong to different classes are at distance from each other while features which belong to same class are near to each other

2.3.4 Prediction Algorithm

In this study **k-nearest neighbor (k-NN) classifier** is used as prediction algorithm with different values of k.

2.3.5 Performance

Three distinct cross validation tests are used to examine the stability and efficiency of the proposed model. This model obtained 94.9% accuracy on jackknife test, 94.8% on independent dataset test and 94.9% on 10-fold cross validation test.

2.4 Prediction using A Novel Feature Extraction Method with Feature Selection by Yang et al.

In this work, a new computational method was proposed for identifying cis-Golgi proteins from trans-Golgi proteins. Based on the concept of Common Spatial Patterns (CSP), a novel feature extraction technique was developed to extract evolutionary information from protein sequences. To deal with the imbalanced benchmark dataset, the Synthetic Minority Over-sampling Technique (SMOTE) was adopted [43].

2.4.1 Dataset

To update the training datasets introduced by Ding et al [40], the Golgi-resident proteins applied in this study are collected from the latest Universal Protein KnowledgeBase (UniProtKB), which provides the scientific community with a comprehensive, high quality and freely accessible resource of protein sequences. To avoid homology bias, we remove the redundant sequences using CD-HIT with a 40% identity cutoff [44] . As a result, the training dataset consists of 87 cis-Golgi proteins and 217 trans-Golgi proteins

2.4.2 Features

In this study, dipeptide composition and evolutionary information are combined to transform the protein sequences into feature vectors. Three traditional feature extraction methods namely, PSSM-DC, Bi-gram PSSM, and ED-PSSM, are adopted to extract evolutionary information from the PSSM. Based on the concept of CSP, a novel feature extraction technique is proposed to extract features from PSSM-DC, Bi-gram PSSM, and ED-PSSM, respectively.

n-gap Dipeptide Composition

The g-gap Dipeptide Composition proposed in [40] is employed in this study to search for the important correlation between two residues.

PSSM-Dipeptide Composition

PSSM-DC transforms L X 20 PSSM into 20 X 20 PSSM which is converted as feature vetor.

Bi-gram PSSM

Bi-gram features directly extracted from PSSM have been adopted in recent studies to address the shortcoming that the computed bi-gram feature vector from the original protein sequence is very sparse.

Evolutionary Difference -PSSM

Evolutionary Difference-PSSM is proposed to represent mutation difference between adjacent residues.

Common Spatial Patterns Based Feature Extraction from Evolutionary Information

The method of common spatial patterns (CSP) has been applied successfully to extract discriminatory information from two populations of single-trial electroencephalograph . In this study, we apply the concept of CSP to extract features from PSSM-DC, Bi-gram PSSM, and ED-PSSM, respectively

2.4.3 Feature Selection

In this study, a feature selection method called RF-RFE (Random Forest-Recursive Feature Elimination) is employed to pick out high discriminative features. The RF-RFE algorithm starts with all input features and removes one feature with the lowest influence on the performance of the RF model from the feature set at each iteration.As there are 460 features in the original feature set,after using this algorithm feature set was reduced to 55 by iterating over all.

2.4.4 Prediction Algorithm

The random forest (RF) algorithm is used as a classifier in this Study.

2.4.5 Performance

Through the jackknife cross-validation, the proposed method achieves a promising performance with a sensitivity of 0.889, a specificity of 0.880, an accuracy of 0.885, and a Matthew's Correlation Coefficient (MCC) of 0.765, which remarkably outperformed previous methods. Moreover, when tested on a common independent dataset, this method also achieves a significantly improved performance.

2.5 Prediction using SVM and Random Forest Based Feature Selection by Saifur Rahman et al.

In this work, The focus was on building a new computational model that not only introduced easy ways to extract features from protein sequences but also optimized classification of trans-Golgi and cis-Golgi proteins [45].

2.5.1 Dataset

This study have collected the training and testing benchmark datasets from Yang et al [43]. The training dataset contains 304 sub-Golgi protein sequences among which there are 87 sequences of cis-Golgi type and 217 sequences of trans-Golgi type.

2.5.2 Features

Amino Acid Composition

Amino Acid Composition (AAC) of a protein sequence means the normalized frequencies of the 20 native amino acids. This can thus contribute 20 features in a feature vector. After counting the frequency of each amino acid residue, normalization is done by dividing the count by the length of the sequence.This study calculated the average AAC in the training dataset for cis-Golgi and trans-Golgi protein.

Dipeptide

The Dipeptides (Dip) feature type computes the normalized frequency of adjacent amino acids within the sequence. Another commonly used name for this feature type is Dipeptide Composition (DPC). This feature type provides into the feature vector some sequence-order information and has been successfully used in several protein related studies.Dip thus contributes 400 features to our feature vector.

Tripeptide

This study has also applied the notion of Tripeptides to extract another 8000 features. Observe that all these feature types derive from the generalized form of n-grams feature type where frequencies of n-length peptides are used as feature vectors. In this study, a total of 8420 n-grams features, for n = 1, 2 and 3 has ben extracted.

n-Gapped-Dipeptide

In the n-Gapped-Dipeptides (nGDip) feature type,the frequency of amino acid dipeptides such that the amino acids are separated by n positions. The frequency is normalized, dividing by the total number of nGDip (i.e., L n 1 for a sequence of length L). This can contribute 400 features to the feature vector.

Position Specific Features

The position specific features category generates features which identify whether specific n-grams occur in specific positions in the protein sequence. The value of each such feature in any sequence will therefore be either 0 or 1 (on or off). If the maximum sequence length is L, the feature space size would be L 20^n. For small sample sizes, however, many of the features will not have discriminating scores. Such features that are on in all samples (or vice verse) can be excluded from the final feature vector. Thus the actual size may be considerably smaller than the theoretical maximum. For a specific position, 20, 400 and 8000 PSF features can be generated for n = 1, 2, and 3 respectively. Since there are only 304 training samples, no more than 304 features can be generated for each of n = 2 and 3, such that at least one sample has the respective feature on. Thus the number of features with discriminating scores cannot be larger than (10 20 + 9 304 + 8 304) = 5368. Depending on the actual samples, this number may be less than this higher bound – some features will be on in many samples, whereas some will be off in all samples. The actual number of PSF features for training set was 4492.

PseAAC

Any feature space that tries to capture the sequence order information of a protein can be formulated in the general form of Chou's Pseudo Amino Acid Composition (PseAAC).

2.5.3 Feature Selection

Filtering Phase

In the filtering phase, this study has applied Random Forest (RF) on the entire feature set to generate a model. Through this model creation, the RF algorithm was able to set an importance score (MeanDecreaseAccuracy) to each of the input features. This importance score indicates the global importance over all out-of-bag cross validated predictions and is very robust as it is averaged over all predictions for a given feature variable. The importance score is used to rank the features and subsequently filter irrelevant features.

Wrapper Phase

2.5.4 Prediction Algorithm

This experiment was done in R language. To construct the isGPT model,this study has used Random Forest (RF) and SVM machine learning algorithms. These are available respectively from the R packages randomForest and e1071. In the RF algorithm, we have used the default parameters setting. In particular, the number of trees (ntree) was restricted to 500, while the number of variables tried at each split (mtry) was set to square root of the total number of features. RF model has been used for feature selection, while SVM is used to learn the model. Since training set is relatively small, this study has used linear kernel function in SVM to avoid overfitting.

2.5.5 Performance

This method is known , namely, identification of sub-Golgi Protein Types (isGPT), achieved accuracy values of 95.4%, 95.9% and 95.3% for 10-fold cross-validation test, jackknife test and independent test respectively. According to different performance metrics, isGPT performs better than state-of-the-art techniques.

Chapter 3

Materials and Methods

3.1 Dataset

To update the training datasets introduced by Ding et al [40] the Golgi-resident proteins applied in this study are collected from the latest Universal Protein KnowledgeBase (UniProtKB), which provides the scientific community with a comprehensive, high quality and freely accessible resource of protein sequences [46].To search the cis-Golgi proteins or trans-Golgi proteins, respectively, we use the keyword of subcellular locations ("cis-Golgi" or "trans-Golgi") and add restrictions, that is "fragment: not", "containing nonstandard letters: yes", and "reviewed: yes". These restrictions are applied to reduce the redundant, incomplete, and incorrect information. To avoid homology bias, we remove the redundant sequences using CD-HIT with a 40% identity cutoff. As a result, the dataset consists of 160 cis-Golgi proteins and 944 trans-Golgi proteins. Training dataset consists of 75% of the main dataset and the rest are the testing dataset.In order to facilitate comparison with previous studies, a dataset composed of 13 cis-Golgi proteins and 51 trans-Golgi proteins, introduced by Ding et al [40], is employed to construct the independent testing dataset. Predicting Golgi-resident protein types is formulated as a two class classification problem, where cis-Golgi proteins belong to the positive class and trans-Golgi proteins to the negative class.

3.2 Features

3.2.1 n-gap Dipeptide Composition

The diversity and specificity of protein structures and functions are largely attributed to amino acid compositions [47]. Adjoining dipeptide composition represents the occurrence frequency of each two adjacent amino acid residues. Compared to the amino acid composition, the adjoining dipeptide composition encapsulates both the fraction information of amino acids and the local order information of protein sequences, which has been used for protein attribute predictions [48,49]. Without considering the intrinsic properties deposited in the correlations between spatially close amino acid residues [50], the adjoining dipeptide composition can only depict the correlation between two adjoining amino acids.n order to exploit the occurrence of amino acid residues along with structural information the concept of dipeptide composition (DPC) is adopted. In DPC, the pairing information among the amino acid residues is computed [51]. Thus, the g-gap Dipeptide Composition proposed in [40] is employed in this study to search for the important correlation between two residues.

For a protein sequence P with L residues, the g-gap dipeptide composition can be expressed as follows.

$$F^g = \{f_1^g, f_2^g, f_i^g, ..., f_{400}^g\}^T \tag{3.1}$$

where the symbol T denotes the transpose of the vector. g is the number of intervening residue. fig denotes the frequency of the ith g-gap dipeptide and is defined as

$$f_i^g = \frac{n_i^g}{L - g - 1} \tag{3.2}$$

where n_i^g denotes the number of the ith g-gap dipeptide i denotes the number of the ith g-gap dipeptide.

3.2.2 Split pseudo amino acid composition (Split-PseAAC)

PseAAC is computed on divided sequences instead of AAC since calculates the occurences of individual amino acids which lead to loss of important information that may be concealed in the structure of amino acids residues. PseAAC on the other hand calculates the frequency of amino acids that constitute a protein sequence as well as it considers the long range correlation of the physico chemical properties between two residues of the sequence

Traditional Feature Extraction Methods from Evolutionary Information

As one of the most important aspects in biological sequence analysis, evolutionary conservation, reflects important biological functions [52]. Conserved sequences are similar or identical sequences that still share many common features during the evolution process [53]. A functionally important region is always conservative in the evolutionary process [54]. Exploiting the detailed conservation pattern of residues will largely facilitate the prediction of protein functions [55]. PSSM has been widely used to transform the variable lengths of protein sequences into fixed-length feature vectors while keeping considerable evolutionary information [56]. The PSI-BLAST (Position-Specific Iterative Basic Local Alignment Search Tool) [57] is used to generate PSSM by searching homogenous sequences for each query protein through three iterations with 0.001 as the E-value cutoff. The search is performed against the Swiss-Prot database. PSSM profile for each query protein can be expressed as

$$P_{PSSM} = \begin{bmatrix} E_{1\to1} & E_{1\to2} & ... & E_{1\to j} & ... & E_{1\to j} \\ E_{2\to1} & E_{2\to2} & ... & E_{2\to j} & ... & E_{2\to j} \\ \vdots & \vdots & \ldots & \vdots & \ldots & \vdots \\ E_{i\to1} & E_{i\to2} & ... & E_{i\to j} & ... & E_{i\to j} \\ \vdots & \vdots & \ldots & \vdots & \ldots & \vdots \\ E_{L\to1} & E_{L\to2} & ... & E_{L\to j} & ... & E_{L\to j} \end{bmatrix} \tag{3.3}$$

where L is the length of the query sequence and the values of j = 1, 2,..., 20 represent the 20 native amino acids according to their alphabetical order. $E_{i\to j}$ can be interpreted as the relative probability of jth amino acid at the ith location of the query sequence during the evolution process. Large positive scores often indicate critical functional residues. In this study, three traditional feature extraction methods namely, PSSM-DC, Bi-gram PSSM, and ED-PSSM, are adopted to extract evolutionary information from the PSSM.

3.2.3 Bigram position specific scoring matrix

Bi-gram features directly extracted from PSSM have been adopted in recent studies [58] to address the shortcoming that the computed bi-gram feature vector from the original protein sequence s very sparse. Bi-gram PSSM computes the frequency of occurrence of transition from mth amino acid to nth amino acid as follows:

$$B_{m,n} = \sum_{i=1}^{L-1} E_{i\to m} E_{i+1\to n}, m, n = 1, 2, ..., 20 \tag{3.4}$$

The values of (m, n = 1, 2, . . . , 20) denote the 20 native amino acids according to their alphabetical order. Equation (3.5) gives 400 frequencies of occurrences, which can be formulated as

$$B = [B_{1,1}, B_{1,2}, ..., B_{1,20}; B_{2,1}, B_{2,2}, ..., B_{2,20}; B_{20,1}, B_{20,2}, ..., B_{20,20}]^T \tag{3.5}$$

where T denotes the transpose of the vector.

3.2.4 PSSM-Dipeptide Composition

Previous works have exhibited the ability of PSSM-dipeptide composition (PSSM-DC) in the protein function predictions [59]. PSSM-DC transforms L ˆ 20 PSSM into 20 ˆ 20 PSSM as formulated by

$$PSSM - DC = \begin{bmatrix} \sum E_{A \to A} & \sum E_{A \to R} & \cdots & \sum E_{A \to V} \\ \sum E_{R \to A} & \sum E_{R \to R} & \cdots & \sum E_{R \to V} \\ \vdots & \vdots & \vdots & \vdots \\ \sum E_{V \to A} & \sum E_{V \to R} & \cdots & \sum E_{V \to V} \end{bmatrix} \tag{3.6}$$

where $\sum_{E_{i \to j}}$ denotes the sum of amino acid type i being changed to amino acid type j in Equation (3.3), followed by division of each element by the length of the sequence.

3.2.5 Evolutionary Difference-PSSM

Evolutionary Difference-PSSM is proposed to represent mutation difference between adjacent residues. A given protein can be expressed as a 20 ˆ 20 matrix ED-PSSM denoted by

$$ED - PSSM = (e_1, e_2, ..., e_m, ..., e_{20}), m = 1, 2, ...20 \tag{3.7}$$

where

$$e_m = (e_{1,m}, e_{2,m}, ..., e_{n,m}, ..., e_{n,20},)^T, n = 1, 2, ...20 \tag{3.8}$$

$$e_{m,n} = \sum_{i=2}^{L-1} \frac{(E_{i-1 \to m} - E_{i+1 \to n})^2}{L - 2} \tag{3.9}$$

3.2.6 Common Spatial Patterns Based Feature Extraction from Evolutionary Information

The method of common spatial patterns (CSP) has been applied successfully to extract discriminatory information from two populations of single-trial electroencephalograph [59]. In this study, we apply the concept of CSP to extract features from PSSM-DC, Bi-gram PSSM, and ED-PSSM, respectively. Through PSSM-DC, Bi-gram PSSM, or ED-PSSM, the protein sequence is represented as a 20 ˆ 20 matrix E. The normalized spatial covariance of the protein sequence can be obtained from

$$R = \frac{EE\prime}{trace(EE\prime)} \tag{3.10}$$

where ’ denotes the transpose operator and tracepxq is the sum of the diagonal elements of x. The composite spatial covariance is given as

$$R_c = \bar{R}_1 + \bar{R}_2 \tag{3.11}$$

where the spatial covariance $\bar{R}_1$ is calculated by averaging over the cis-Golgi protein sequences and the spatial covariance $\bar{R}_2$ is calculated by averaging over the trans-Golgi protein sequences. R_c can be factored as $R_c = U_c\lambda U_c'$ where U_c is the matrix of eigenvectors and λ_c is the diagonal matrix of eigenvalues.

The whitening transformation P = $\sqrt{\lambda_c^-1}U_c'$ equalizes the variances in the space spanned by U_c', i.e., all eigenvalues of PR_cP' are equal to one. If $\bar{R}_1$ and $\bar{R}_2$ are transformed as

$$S_1 = P\bar{R}_1P\prime, S_2 = P\bar{R}_2P\prime \tag{3.12}$$

then S_1 and S_2 share common eigenvectors, i.e., if $S_1 = B\lambda_1B'$, then $S_1 = B\lambda_2B'$ and $\lambda_1 + \lambda_1 = I$, where I is the identity matrix. This property indicates that for a same eigenvector, the corresponding eigenvalue for S_1 is the largest (smallest) while the corresponding eigenvalue for S_2 is the smallest (largest). Therefore, the eigenvectors is suitable to extract features for classification. With the projection matrix W =(B’P)’, the mapping of a protein sequence is given as

$$Z = WE \tag{3.13}$$

The feature vector $F = \{f_1, f_2, ..., f_{20}\}$ used for classification is obtained by

$$f_i = \log\left(\frac{var(Z_j)}{\sum_{i=1}^{20} var(Z_i)}, j = 1, 2, ...20 \tag{3.14}$$

where the subscript of Z denotes the column number of matrix Z. Based on the method of CSP, the features extracted from PSSM-DC, Bi-gram PSSM, and ED-PSSM are denoted as CSP-PSSM-DC, CSP-Bi-gram PSSM, and CSP-ED-PSSM, respectively.

3.3 Synthetic Minority Over-Sampling Technique

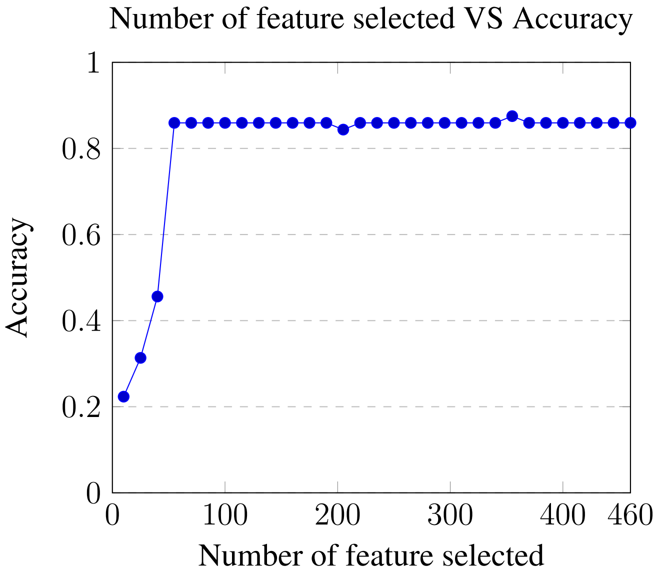
As described in the “Datasets” section, the number of cis-Golgi proteins is much smaller than that of trans-Golgi proteins. This leads to the imbalanced data classification problem. In order to deal with this imbalanced data problem, we consider the SMOTE (Synthetic Minority Over-sampling Technique) to achieve balance. To over-sampling the minority class, SMOTE selects a minority class sample and creates novel synthetic samples along the line segment joining some or all k nearest neighbors belonging to that class [60]. In this paper, to make the number of cis-Golgi samples be equal to the number of trans-Golgi samples, new cis-Golgi samples in the feature spaces are generated via SMOTE algorithm. Subsequently, this balanced dataset, having an equal number of cis-Golgi and trans-Golgi samples, is used for training the predictor. Though SMOTE is pretty useful in order to balance data. But it has drawbacks. Some drawbacks art:

- While generating synthetic examples, SMOTE does not take into consideration neighboring examples can be from other classes. This can increase the overlapping of classes and can introduce additional noise.
- SMOTE is not very practical for high dimensional data.

3.4 Feature Selection

3.4.1 Recursive Feature Elimination

The generated features by the above-mentioned feature extraction methods may be irrelevant to the prediction of golgi-resident protein types, which can result in over-fitting, information redundancy and dimension disaster [61]. To select high discriminative features and reduce computational complexity, the feature selection procedure is always indispensable in protein function predictions based on machine learning methods [62]. In this study, a feature selection method called RF-RFE (Random Forest-Recursive Feature Elimination) is employed to pick out high discriminative features. The RF-RFE algorithm starts with all input features and removes one feature with the lowest influence on the performance of the RF model from the feature set at each iteration. As there are 460 features in the original feature set, 460 iterations are carried out to extract the optimal features. The parameter "Accuracy" is used to evaluate the influence on the performance of the RF model. The first removed feature is the most unimportant feature; the second removed feature is the second most unimportant feature;...; the last removed feature is the most important feature. We run the RF-RFE algorithm to get a rank list according to the feature importance. A new feature set is constructed when another feature has been removed. The feature set that yields the highest cross-validation accuracy among all iterations is selected as the optimal feature set.The graph below shows the iterative result by removing less important feature one by one plotted against accuracy as parameter to select mimimum feature number to



gain optimum output.

3.4.2 Fisher Feature Selection

Dimensionality in data is a curse . Highly dimensional data space often yields poor classification results and increased time and space complexity. Moreover, unwanted and redundant data badly effects on the prediction of model.In additon, the predictive model becomes overfit. Feature selection methods are generaly categaroized into three clusters; filter based, wrapper based and embedded. Filter based methods rank the features, based on individual feature properties and their significance, as a step prior to learning classifier and selects a subset of top ranked feature from input features. Wrapper based methods rank features by using the learning classifier score to select feature subset. Embedded methods incorporate the selection process with the learning classifier to get optimal feature subset [63]. n this study, Fisher feature selection method is used, which is a supervised and filter based method. In this method, selection of individual feature is based on some performance criterion i.e. Fisher score. Fisher score provides subset of features in such a way that in all the selected features, the features which belong to different classes are at distance from each other while features which belong to same class are near to each other. Subset of ‘m’ features is selected from a total of ‘d’ features in such a way that each selected feature must have maximize the Fisher score of the overall subset. The selection of subset Cd m is a combinatorial optimization problem and optimal solution for such an algorithm is NP-hard. One solution is the heuristic approach to compute a score of an individual feature and then selecting top ranked ‘m’ features based on score. Fisher score of the i’th feature Si will be calculated as:

$$S_i = \frac{\sum_{k=1}^{k} n_j (\mu_{ij} - \mu_i)^2}{\sum_{k=1}^{k} n_j \rho_{ij}^2} \tag{3.15}$$

where μ_{ij} and ρ_{ij} are the mean and the squared deviation from the mean of the ith feature in the jth class, respectively. Whereas n_j is the total number of sequences in the jth class, and μ_i is the overall mean of the ith feature.

3.4.3 Wrapper Phase

In case of isGPT wrapper methd alongside filtering methods have been used. In order to this instead selecting the top most features, further experimentation have been done with the top most 3500,3000,2500,2000 and 1500 features by training SVM regression models on the existing dataset as well as on the balanced dataset. From this experiment we get the ROC and PR curves. Maximum area under this curves gives the best results.

Table 3.1: Area under ROC and PR curves for different number of top-ranked features selected.(Updated results are in parenthesis)

	Without SMOTE		With SMOTE	
Number of features	AUCROC	AUCPR	AUCROC	AUCPR
3500	0.55 (0.61)	0.33 (0.4)	0.73 (0.74)	0.68 (0.69)
3000	0.59 (0.63)	0.37 (0.38)	0.74 (0.74)	0.68 (0.68)
2500	0.75 (0.71)	0.53 (0.5)	0.95 (0.96)	0.95 (0.95)
2000	0.78 (0.78)	0.6 (0.58)	0.95 (0.96)	0.95 (0.96)
1500	0.75 (0.76)	0.57 (0.57)	0.95 (0.96)	0.95 (0.95)

Here we give the results of new dataset.

Table 3.2: Results of new dataset.

Number of features	Without SMOTE		With SMOTE	
	AUCROC	AUCPR	AUCROC	AUCPR
3500	0.61	0.25	0.8	0.8
3000	0.59	0.23	0.79	0.79
2500	0.58	0.22	0.77	0.77
2000	0.57	0.22	0.75	0.75
1500	0.54	0.21	0.64	0.65

3.5 Prediction Algorithm

3.5.1 Random Forest Classifier

To implement Yang's method we need to Random Forest classifier algorithm.The random forest (RF) algorithm, developed by Breiman [64], has been successfully applied in the field of protein function predictions [65]. The ensemble of decision trees generated by RF gives a good tolerance for the noisy data [64]. The decision trees are trained on different bootstrap samples from the training data. Each tree is fully grown without pruning. At each node, m features are selected randomly out of all features and the most optimized split on these m features is employed to split the node. For a new object, each decision tree gives a classification result. Based on the classification results of decision trees, RF assigns the new object a class label through majority voting. The Random Forest algorithm is implemented in python by scikit package where default parameters are used.

3.5.2 k-Nearest Neighbour Classifier

To Implement Ahmads Method we need to use k-NN algorithm.k-NN is a very popular and simple instance based, nonparametric classifier, which trains model upon locally available information only and the computation is delayed till classification [66]. In other words, k-NN does not generalize the model until a query is made so it is also called lazy learner. K-NN classifier depends upon the input of the neighbors where close neighbors contribute more as compared to the distant ones [67]. The algorithm is trained by considering the ‘k’ neighbors where ‘k’ is selected heuristically. The algorithm suggests the class membership by simple majority voting among the k-neighbors so if ‘k=1’ the object will be assigned to the same class as of that neighbor, if k=2, assigned class will be of the nearest neighbor and so on. The value ‘k’ is usually a small positive integer as large values make class boundaries less distant although larger values for ‘k’ reduce the impact of noise on the classification results [68]. An associated limitation of k-NN is the larger space requirements as no generalization is made in learning so all the dataset needs to be stored till classification.

3.6 Evaluation Metrics

3.6.1 Accuracy

It is a very fundamental parameter to test the efficiency of the model and is calculated as

$$Acc = \frac{TP + TN}{TP + FP + TN + FN} \tag{3.16}$$

In normal circumstances, accuracy alone provides sufficient information to test the precision of the system. However, as [69] suggests, accuracy alone may be misleading in certain situations. For instance, in case of highly imbalance data where more than 90% of the data belong to a single class; a classifier may be overfit and predict all the test samples in the majority class. The accuracy in this case would be more than 90% because of high ‘True Positive’ rate but the system is erroneous and is unacceptable.

3.6.2 Sensitivity

Sensitivity is a very powerful measure to test the effectiveness of a system. It returns the true positive rate i.e. how many cases which belong to a class are predicted accurately. For example, suppose diagnostic tests are being carried out for a specific disease. In that case, sensitivity will denote that how many of the subjects, that carry the disease, are detected accurately. Higher the sensitivity lead to higher accuracy count and vice versa [70]. The sensitivity is represented mathematically as

$$S_n = \frac{TP}{TP + FN} \tag{3.17}$$

3.6.3 Specificity

Specificity, as opposite to sensitivity, measures the true negative rate and has equal effect in the measurement of the performance of the system. Thus both, sensitivity and specificity are calculated side by side for a model. As supposed in the sensitivity example, specificity will denote that how many of the cases where patient does not carry that specific disease, are detected accurately. High values of sensitivity and specificity results in overall high accuracy count whereas if specificity is low, the accuracy is biased towards the value of sensitivity [70]. Mathematical formula for specificity calculation is given below

$$S_p = \frac{TN}{TN + FP} \tag{3.18}$$

3.6.4 Matthew's correlation coefficient (MCC)

MCC assess the overall quality of the classification and produce results in 1 and +1 range where 1 denotes that the classifier is consistently predicting the wrong classes and +1 means that the classifier has perfectly predicted whole sample accurately. A value near 0 dictates that the prediction system is producing average random predictions and the resulting precision, therefore, becomes unreliable. MCC is a very effective measure specially in case of biased dataset since high accuracy count with 0 MCC expose the true quality of the prediction system [71]. The MCC is calculated as

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}} \tag{3.19}$$

3.6.5 Leave-one-out (Jackknife)

Among these tests, Jackknife test is widely used in bioinformatics along with many other fields, in addition, it always produces unique outcome [72]. In this type of assessment, the training dataset is given to the classifier to get trained on it despite one sample that is initially removed while the removed instance from training dataset is considered as testing instance. For a better result, the test is conducted 'N' times repeatedly with different test sample extracted every time from the training dataset where ‘N’ is the number of samples in the dataset. The jackknife is a very effective evaluation test as it produces unbiased results with minimum variance [73]. However, this test has relatively slow execution time which depends upon the number of iterations/samples in the dataset.

3.6.6 Independent testing

It is also known as Hold out test, the training dataset is completely different from the samples that are to be tested once the classifier gets trained. So, in this case, we have two separate datasets, one for training and other for testing. It is very easy and fast method; however, it suffers from small test sample as well as the results are dependent upon the selection of test set so a large variance (uncertainty) is expected. It is very important that the testing dataset should be large and diverse enough to cover all/most of the training dataset otherwise, the results of the independent testing would be misleading [74]. In some cases, where two separate training and testing datasets are not available, the training dataset could be divided into two portions, one of which is used for training and other for testing. In such case, the training dataset is normally twice in size of the testing portion.

3.6.7 Cross validation

As discussed earlier, jackknife is a very effective test to measure the performance of classifiers; however, it takes considerable time and resources. Another similar test is cross validation which divides the whole dataset into ‘m’ groups of equal sizes. The classifier is trained on ‘m 1’ group and then tested on the hold out group [75]. For better average, the classifier is tested ‘m’ times and in the end the results are averaged. In normal practice, 10-fold cross validation test is mostly used which divides the dataset into 10-folds of equal size. The test results are fairly unbiased and results possess minimum variance. Apart from the type of test to be performed, it is important to discuss the parameters or the statistical measures which would be used for the assessment. In this study, four standard performance indicators are used i.e. Accuracy, Sensitivity, Specificity and Matthew’s Correlation Coefficient [76]. These calculations are carried out on confusion matrix which tabulates the error rate of an algorithm in terms of actual and predicted classifications.

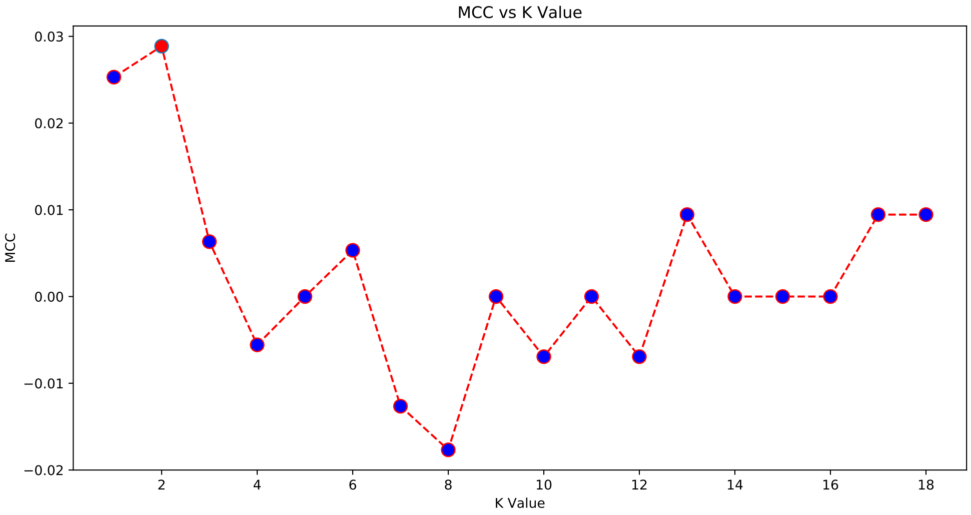
Chapter 4

Results

4.1 Ahmad’s Method

4.1.1 Without SMOTE Results

Following is the prediction results of Ahmad’s method without SMOTE using existing dataset. We have also run our implementation of Ahmad’s method without SMOTE using existing dataset. The results are diffenrent from the Ahmad’s results. May be it is beacuse of using different implementation and different machines. Following is the graph of k vs mcc of combined features.



Performance of various feature extraction techniques without SMOTE (our results are in parenthesis).

Table 4.1: Results of jackknife cross validation on existing dataset without SMOTE.

		Jackknife cross validation			
Feature Space	Dimension	Acc	Sn	Sp	MCC
Di-Peptide Composition	304×400	91.8 (68.09)	87.1 (13.79)	93.2 (89.86)	0.81 (0.05)
Bigram-PSSM	304×400	74.4 (71.94)	69.4 (43.02)	76.3 (83.41)	0.45 (0.27)
SAAC	304×66	95.1 (63.81)	93 (76.49)	95.9 (32.18)	0.89 (0.08)
DPC+Bigram+SAAC	304×866	89.1 (63.81)	83.1 (64.51)	91.6 (62.06)	0.76 (0.24)

Table 4.2: Results of independent testing on existing dataset without SMOTE.

		Independent Testing			
Feature Space	Dimension	Acc	Sn	Sp	MCC
Di-Peptide Composition	304×400	85.9 (70.31)	58.4 (53.84)	93 (74.5)	0.62 (0.24)
Bigram-PSSM	304×400	95.3 (80.95)	90 (0)	96.7 (100)	0.88 (0)
SAAC	304×66	81.3 (28.12)	63.7 (9.8)	85.7 (100)	0.51 (0.14)
DPC+Bigram+SAAC	304×866	90.6 (82.81)	79.9 (96.07)	93.4 (30.76)	0.75 (0.37)

Table 4.3: Results of 10-fold cross validation on existing dataset without SMOTE.

		10-fold cross validation			
Feature Space	Dimension	Acc	Sn	Sp	MCC
Di-Peptide Composition	304×400	89.5 (68.7)	83.8 (18.13)	91.7 (88.89)	0.76 (0.08)
Bigram-PSSM	304×400	63.8 (72.58)	42.3 (38.62)	72.5 (83.86)	0.15 (0.23)
SAAC	304×66	95.1 (62.56)	90.1 (75.99)	97.1 (28.48)	0.89 (0.04)
DPC+Bigram+SAAC	304×866	88.8 (62.84)	77.7 (63.55)	93.3 (62.9)	0.75 (0.23)

Below is the result of our implementation using new dataset without SMOTE.

Table 4.4: Results of jackknife cross validation on new dataset without SMOTE.

		Jackknife Cross Validation			
Feature Space	Dimension	Acc	Sn	Sp	MCC
Di-Peptide Composition	944×400	82.29	2.5	98.59	0.03
Bigram-PSSM	944×400	83.03	36.87	92.46	0.33
SAAC	944×66	82.5	13.12	96.67	0.16
DPC+Bigram+SAAC	944×866	83.45	17.5	96.93	0.23

Table 4.5: Results of idependent testing on new dataset without SMOTE.

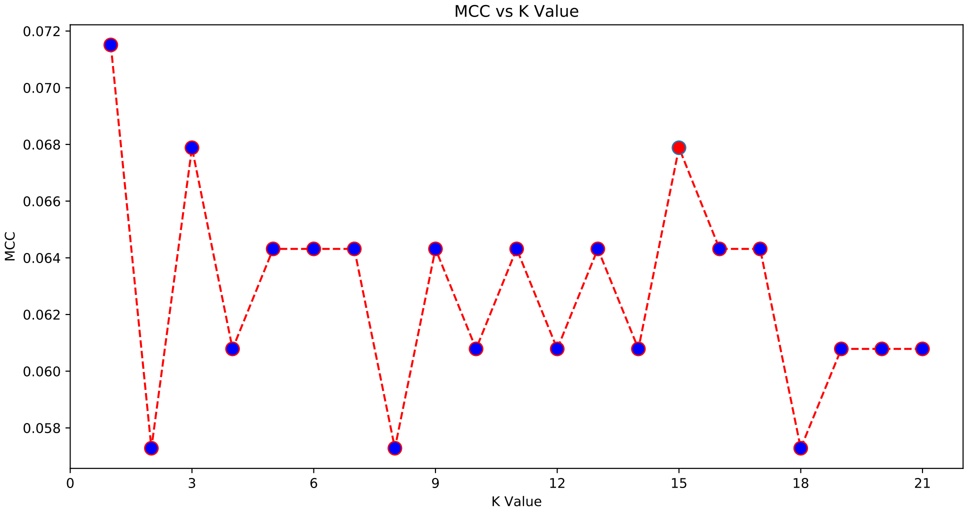
		Independent Testing			
Feature Space	Dimension	ACC	Sn	Sp	MCC
Di-Peptide Composition	944×400	83.05	0	100	0
Bigram-PSSM	944×400	79.66	17.5	92.34	0.12
SAAC	944×66	83.05	2.5	99.48	0.08
DPC+Bigram+SAAC	944×866	77.54	10	91.32	0.01

Table 4.6: Results of 10-fold cross validation on new dataset without SMOTE.

		10-fold Cross Validation			
Feature Space	Dimension	ACC	Sn	Sp	MCC
Di-Peptide Composition	944×400	82.18	2.5	98.47	0.04
Bigram-PSSM	944×400	84.61	28.11	96.22	0.33
SAAC	944×66	82.93	13.79	97.23	0.2
DPC+Bigram+SAAC	944×866	83.03	16.82	96.59	0.22

4.1.2 With SMOTE Results

Ahmad have also computed results using SMOTE. We have also computed results using our implementation using SMOTE. Again results are different. May be because of different implementation and different machines. Following is the graph of k vs mcc of combined features of balanced dataset.



Performance of various feature extraction techniques and their combination with increased samples (via SMOTE) (our results are in parenthesis).

Table 4.7: Results of jaccknife test on existing dataset with SMOTE.

		Jackknife cross validation			
Feature Space	Dimension	Acc	Sn	Sp	MCC
Di-Peptide Composition	434×400	95.4 (67.51)	97.5 (88.01)	93.3 (47)	0.91 (0.38)
Bigram-PSSM	434×400	85.9 (65.89)	90.2 (96.77)	81.7 (35.02)	0.72 (0.4)
SAAC	434×66	97.3 (73.5)	98.4 (48.38)	96.1 (98.61)	0.95 (0.54)
DPC+Bigram	434×800	90.1 (68.89)	94.8 (98.15)	85.4 (39.63)	0.81 (0.46)
Bigram+SAAC	434×466	91.9 (67.97)	95.5 (36.4)	88.4 (99.53)	0.84 (0.46)
DPC+SAAC	434×466	95.6 (64.51)	97.8 (35.94)	93.4 (93.08)	0.91 (0.35)
DPC+Bigram+SAAC	434×866	93.8 (63.82)	96.9 (29.03)	90.7 (98.61)	0.88 (0.38)

Table 4.8: Results of independent testing on existing dataset with SMOTE.

		Independent Testing			
Feature Space	Dimension	Acc	Sn	Sp	MCC
Di-Peptide Composition	434×400	85.9 (54.68)	62.2 (69.23)	92 (50.98)	0.61 (0.16)
Bigram-PSSM	434×400	96.9 (76.56)	94.6 (23.07)	97.5 (90.19)	0.92 (0.16)
SAAC	434×66	75 (28.12)	68.3 (9.8)	76.7 (100)	0.43 (0.14)
DPC+Bigram	434×800	92.2 (81.25)	80.7 (15.38)	95.1 (98.03)	0.79 (0.25)
Bigram+SAAC	434×466	93.8 (76.56)	89.2 (96.07)	94.9 (0)	0.84 (-0.09)
DPC+SAAC	434×466	89.1 (78.12)	79.2 (80.39)	91.6 (69.23)	0.72 (0.43)
DPC+Bigram+SAAC	434×866	92.2 (79.68)	80.7 (96.07)	95.1 (15.38)	0.79 (0.19)

Table 4.9: Results of 10-fold cross validation on existing dataset with SMOTE.

		10-fold cross validation			
Feature Space	Dimension	Acc	Sn	Sp	MCC
Di-Peptide Composition	434×400	93.8 (66.8)	96.7 (87.58)	90.9 (47.02)	0.88 (0.37)
Bigram-PSSM	434×400	75.3 (66.13)	84.7 (96.77)	66 (35.47)	0.53 (0.4)
SAAC	434×66	96.8 (72.81)	97 (47.36)	95.6 (98.07)	0.94 (0.52)
DPC+Bigram	434×800	88.5 (68.43)	93.8 (97.48)	83.2 (39.81)	0.78 (0.45)
Bigram+SAAC	434×466	88.3 (67.94)	93 (36.29)	83.5 (99.58)	0.77 (0.45)
DPC+SAAC	434×466	94.5 (64.33)	97.2 (36.62)	91.7 (92.95)	0.89 (0.35)
DPC+Bigram+SAAC	434×866	91.9 (64.56)	96 (31.13)	87.9 (98.26)	0.84 (0.39)

Following is the result of our implementation using new dataset which was balanced using SMOTE.

Table 4.10: Results of jackknife cross validation on new dataset with SMOTE.

		Jackknife Cross Validation			
Feature Space	Dimension	Acc	Sn	Sp	MCC
Di-Peptide Composition	1566×400	80.58	98.59	62.57	0.65
Bigram-PSSM	1566×400	88.25	98.46	78.03	0.78
SAAC	1566×66	91.31	98.59	84.03	0.83
DPC+Bigram	1566×800	79.69	98.85	60.53	0.64
Bigram+SAAC	1566×466	87.99	98.33	77.65	0.77
DPC+SAAC	1566×466	83.52	98.46	68.58	0.7
DPC+Bigram+SAAC	1566×866	79.69	98.85	60.53	0.64

Table 4.11: Results of independent testing on new dataset with SMOTE.

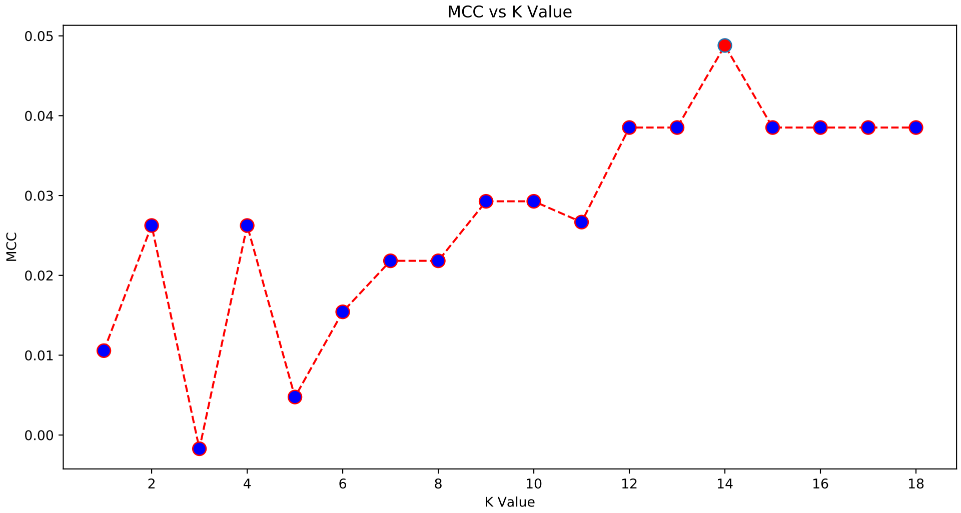
		Independent Testing			
Feature Space	Dimension	Acc	Sn	Sp	MCC
Di-Peptide Composition	1566×400	59.74	42.5	63.26	0.04
Bigram-PSSM	1566×400	61.86	65	61.22	0.19
SAAC	1566×66	80.5	32.5	90.3	0.24
DPC+Bigram	1566×800	47.45	77.5	41.32	0.14
Bigram+SAAC	1566×466	52.54	30	57.14	-0.09
DPC+SAAC	1566×466	49.15	20	55.1	-0.19
DPC+Bigram+SAAC	1566×866	21.18	95	6.12	0.01

Table 4.12: Results of 10-fold cross validation on new dataset with SMOTE.

		10-fold Cross Validation			
Feature Space	Dimension	ACC	Sn	Sp	MCC
Di-Peptide Composition	1566×400	80.07	98.13	62.05	0.64
Bigram-PSSM	1566×400	87.73	98.23	77.45	0.77
SAAC	1566×66	90.29	97.59	83.09	0.81
DPC+Bigram	1566×800	79.69	98.58	60.8	0.64
Bigram+SAAC	1566×466	88.06	97.93	78.11	0.77
DPC+SAAC	1566×466	82.75	98.1	67.43	0.68
DPC+Bigram+SAAC	1566×866	79.43	98.88	60.11	0.63

4.1.3 Using Feature Selection

Ahmad have also used feature selection to rank dataset. He used fisher selection method. After using it he took the first 83 features. Here we have also used the fisher selection method described by Ahmad in out implementaecon. Below are the result with and without fisher selection method. Following is the graph of k vs mcc of selected features of balanced dataset.



Performance comparison with and without feature selection. (our results in parenthesis).

Table 4.13: Results of jackknife cross validation on existing dataset with and without feature selection.

		Jackknife cross validation			
Feature Space	Dimension	Acc	Sn	Sp	MCC
DPC+Bigram+SAAC	866	93.8 (63.82)	96.9 (29.03)	90.7 (98.61)	0.88 (0.38)
DPC+Bigram+SAAC (Fisher)	83	94.9 (76.31)	97.2 (91.7)	92.6 (37.93)	0.9 (0.35)

Table 4.14: Results of independent testing existing dataset with or without feature selection.

		Independent Testing			
Feature Space	Dimension	Acc	Sn	Sp	MCC
DPC+Bigram+SAAC	866	92.2 (79.68)	80.7 (96.07)	95.1 (15.38)	0.79 (0.19)
DPC+Bigram+SAAC (Fisher)	83	94.8 (81.25)	94 (96.07)	93.9 (23.07)	0.86 (0.28)

Table 4.15: Results of 10-fold cross validation existing dataset with or without feature selection.

		10-fold cross validation			
Feature Space	Dimension	Acc	Sn	Sp	MCC
DPC+Bigram+SAAC	866	91.9 (64.56)	96 (31.13)	87.9 (98.26)	0.84 (0.39)
DPC+Bigram+SAAC (Fisher)	83	94.9 (75.96)	97.2 (91.7)	92.6 (38.85)	0.9 (0.34)

Below is the result of our implementation of fisher selection method using new dataset.

Table 4.16: Resutls of jackknife cross validation on new dataset with or without feature selection.

		Jackknife Cross Validation			
Feature Space	Dimension	Acc	Sn	Sp	MCC
DPC+Bigram+SAAC	866	83.45	17.5	96.93	0.23
DPC+Bigram+SAAC (Fisher Selection)	83	83.77	12.5	98.33	0.22

Table 4.17: Results of independent testing on new dataset with or without feature selection.

		Independent Testing			
Feature Space	Dimension	Acc	Sn	Sp	MCC
DPC+Bigram+SAAC	866	77.54	10	91.32	0.01
DPC+Bigram+SAAC (Fisher Selection)	83	77.96	17.5	90.3	0.09

Table 4.18: Results of 10-fold cross validation on new dataset with or without feature selection.

		10-fold Cross Validation			
Feature Space	Dimension	Acc	Sn	Sp	MCC
DPC+Bigram+SAAC	866	83.03	16.82	96.59	0.22
DPC+Bigram+SAAC (Fisher Selection)	83	82.07	14.59	95.83	0.16

4.2 Yang's Method

4.2.1 g-gapped Dipeptide Prediction

Here is our prediction result comparison on g-gap dipeptide with different value of g ranged from 0 to 8.Our result is not accurate with them which might have happend because of different machine and different tool.as we did it on python they implemented it on WEKA.

Table 4.19: g-gap dipeptide composition with different g. (Our results are in parenthesis.)

g	Sensitivity	Specifcity	Accuracy	MCC
0	0.714 (0.711)	0.908 (0.893)	0.811 (0.802)	0.634 (0.615)
1	0.724 (0.705)	0.894 (0.893)	0.809 (0.799)	0.627 (0.609)
2	0.724 (0.697)	0.899 (0.880)	0.811 (0.789)	0.632 (0.587)
3	0.733 (0.704)	0.926 (.890)	0.829 (0.797)	0.672 (0.605)
4	0.705 (0.7)	0.899 (0..885)	0.802 (0.79)	0.615 (0.592)
5	0.7 (0.69)	0.903 (0.89)	0.802 (0.791)	0.616 (0.593)
6	0.71 (0.706)	0.922 (0.903)	0.816 (0.804)	0.646 (0.62)
7	0.7 (0.705)	0.889 (0.889)	0.795 (0.797)	0.601 (0.605)
8	0.705 (67)	0.935 (0.898)	0.82 (0.786)	0.658 (0.586)

Then we run this model on our newly equipeed dataset.The result is acordingly as follows

Table 4.20: Our results on new dataset (without SMOTE).

g	Sensitivity	Specifcity	Accuracy	MCC
0	0.939	0.855	0.897	0.797
1	0.946	0.842	0.894	0.792
2	0.951	0.846	0.898	0.801
3	0.941	0.85	0.896	0.794
4	0.949	0.847	0.897	0.80
5	0.944	0.85	0.897	0.798
6	0.948	0.848	0.898	0.78
7	0.943	0.844	0.894	0.791
8	0.948	0.848	0.898	0.80

4.2.2 Performance Comparison between the CSP Based Feature Extraction Method and Traditional Feature Extraction Methods from Evolutionary Information

Now we compare the prediction result on existing model with evolutionary feature and CSP based feature

Table 4.21: Prediction results of the CSP based feature extraction method and traditional feature extraction methods from evolutionary information with SMOTE (out results are in parenthesis).

Method	Sensitivity	Specifcity	Accuracy	MCC	Feature Number
PSSM-DC	0.843 (0.85)	0.774 (0.818)	0.809 (0.838)	0.619 (0.678)	400
CSP-PSSM-DC	0.705 (0.914)	0.899 (0.835)	0.802 (0.875)	0.615 (0.752)	20
Bi-gram PSSM	0.71 (0.866)	0.922 (0.831)	0.816 (0.845)	0.646 (0.697)	400
CSP-Bi-gram PSSM	0.843 (0.9)	0.77 (0.836)	0.806 (0.869)	0.615 (0.738)	20
ED-PSSM	0.876 (0.895)	0.82 (0.759)	0.848 (0.83)	0.697 (0.66)	400
CSP-ED-PSSM	0.848 (0.9)	0.829 (0.82)	0.839 (0.861)	0.678 (0.724)	20

The result on new dataset follows as:

Table 4.22: Our results on new dataset.

Method	Sensitivity	Specifcity	Accuracy	MCC	Feature Number
PSSM-DC	0.934	0.927	0.93	0.861	400
CSP-PSSM-DC	0.893	0.936	0.914	0.829	20
Bi-gram PSSM	0.898	0.95	0.924	0.849	400
CSP-Bi-gram PSSM	0.871	0.959	0.915	0.834	20
ED-PSSM	0.875	0.952	0.913	0.829	400
CSP-ED-PSSM	0.869	0.958	0.914	0.831	20

4.2.3 Predictive Capability of Combined Features

In this section, we present the performance analysis of hybrid feature sets constructed by the combination of the CSP based feature extraction method and 3-gap DC. The hybrid features are developed by simple concatenation of individual feature sets.The comparison is as follows:

Table 4.23: The performance of models trained with combined features (Our results are in parenthesis).

Training Feature	Sensitivity	Specifcity	Accuracy	MCC
3-gapDC+CSP-PSSM-DC	0.853 (.939)	0.816 (0.924)	0.834 (0.932)	0.669 (0.864)
3-gapDC+CSP-Bi-gram PSSM	0.853 (.911)	0.839 (0.926)	0.846 (0.918)	0.691 (0.837)
3-gapDC+CSP-ED-PSSM	0.876 (0.91)	0.843 (0.932)	0.859 (0.921)	0.719 (0.842)
3-gapDC+CSP-PSSM-DC+CSP-Bi-gram PSSM	0.857 (.924)	0.793 (0.927)	0.825 (0.925)	0.651 (0.851)
3-gapDC+CSP-PSSM-DC+CSP-ED-PSSM	0.862 (.931)	0.843 (0.924)	0.853 (0.928)	0.705 (0.857)
3-gapDC+CSP-Bi-gram PSSM+CSP-ED-PSSM	0.843 (.913)	0.839 (0.928)	0.841 (0.92)	0.682 (0.841)
3-gapDC+CSP-PSSM-DC+CSP-Bi-gram PSSM+CSP-ED-PSSM	0.876 (922)	0.853 (0.933)	0.864 (0.927)	0.728 (0.855)

Table 4.24: Our results on new dataset.

Training Feature	Sensitivity	Specifcity	Accuracy	MCC
3-gap DC+CSP-PSSM-DC	0.931	0.947	0.939	0.879
3-gap DC+CSP-Bi-gram PSSM	0.957	0.959	0.958	0.916
3-gap DC+CSP-ED-PSSM	0.94	0.956	0.948	0.896
3-gap DC+CSP-PSSM-DC+CSP-Bi-gram PSSM	0.955	0.955	0.958	0.915
3-gap DC+CSP-PSSM-DC+CSP-ED-PSSM	0.938	0.954	0.946	0.892
3-gap DC+CSP-Bi-gram PSSM+CSP-ED-PSSM	0.96	0.959	0.96	0.92
3-gap DC+CSP-PSSM-DC+CSP-Bi-gram PSSM+CSP-ED-PSSM	0.956	0.96	0.958	0.915

4.2.4 Performance of the Current Method with or without SMOTE

In order to investigate the effectiveness of SMOTE in solving the imbalanced dataset problem, the models trained with or without SMOTE are constructed, respectively. Prediction results of the models with or without SMOTE are shown in following table

Table 4.25: Prediction results with and without SMOTE.(Our results are in parenthesis)

Method	Sensitivity	Specifcity	Accuracy	MCC
Without SMOTE	0.184 (.097)	0.949 (0.959)	0.73 (0.778)	0.048 (0.102)
With SMOTE	0.876 (.92)	0.853 (0.933)	0.864 (0.927)	0.728 (0.855)

Then we applied our method on newly equipped dataset to produce evaluation metrics as follows:

Table 4.26: Prediction results with and without SMOTE on our dataset

Method	Sensitivity	Specifcity	Accuracy	MCC
Without SMOTE	0.02	0.998	0.833	0.102
With SMOTE	0.95	0.958	0.954	0.908

4.2.5 Feature Selection Results

We selected exactly 55 feature based on yang's paper to compare the result.

Table 4.27: Prediction results for Golgi-resident protein types using 3-gap DC+CSP-PSSMDC+CSP-Bi-gram PSSM+CSP-ED-PSSM with and without feature selection. (Our resutls are in parenthesis).

Method	Sensitivity	Specifcity	Accuracy	MCC	Feature Number
Without Feature Selection	0.876 (.922)	0.853 (0.933)	0.864 (0.927)	0.728 (0.855)	460
With Feature Selection	0.908 (.904)	0.894 (0.892)	0.901 (0.898)	0.802 (0.796)	55

Now we run the model on newly equipped dataset and the performance is as follows:

Table 4.28: Prediction results for Golgi-resident protein types using 3-gap DC+CSP-PSSMDC+CSP-Bi-gram PSSM+CSP-ED-PSSM with and without feature selection. (Our resutls are in parenthesis).

Method	Sensitivity	Specifcity	Accuracy	MCC	Feature Number
Without Feature Selection	0.95	0.958	0.954	0.91	460
With Feature Selection	0.944	0.91	0.93	.851	55

4.3 isGPT

In order to get the optimal result 10-fold cross validation is used to determine the parameters of classification and regression models. Here we present the updated results and new dataset results.

Table 4.29: Optimal parameters for classification and regression models of isGPT, based on 10- fold results. In the Model Type column, ‘C’ and ‘R’ are used to represent classification and regression respectively. The ‘w/ S’ prefix is added if the model was computed on the dataset balanced with SMOTE. (Updated results are in parenthesis).

Model Type	Number of features	C	Threshold
C w/S	2800 (2700)	10 (10)	N/A (N/A)
C	2050 (1900)	1 (1)	N/A (N/A)
R w/S	2800 (2800)	1 (10)	0.58 (0.59)
R	2250 (2250)	10 (100)	0.44 (0.45)

Table 4.30: Results of new dataset.

Model Type	Number of features	C	Threshold
C w/S	2250	10	N/A
C	2300	10	N/A
R w/S	2000	10	0.94
R	1650	1	0.99

Here we have given the results of existing dataset and updated result. Below this table we have given the results of new dataset.

Table 4.31: Comparison of classification and regression models of isGPT. In the Type column, ‘C’ and ‘R’ are used to represent classification and regression respectively. The ‘w/ S’ prefix is added if the model was computed on the dataset balanced with SMOTE. Acc: accuracy, Sn: sensitivity, Sp: specificity, MCC: Matthew’s Correlation Coefficient. In Jackkinfe column we have used ”-” to represent none value, as we are not able to generate the results because of time constraints. (Updated results are in parenthesis).

Table 4.32: Result of 10-fold cross validation on existing dataset.

Type	10-fold cross-validation			
	Acc	Sn	Sp	MCC
C w/S	94.7 (94.47)	95.9 (92.62)	93.6 (96.31)	0.89 (0.89)
C	80.3 (79.27)	46 (43.67)	94 (93.54)	0.48 (0.44)
R w/S	95.4 (95.16)	95.4 (94.93)	95.4 (95.39)	0.91 (0.9)
R	80.9 (81.9)	48.2 (56.32)	94 (92.16)	0.5 (0.53)

Table 4.33: Results of independent testing on existing dataset.

Type	Independent test			
	Acc	Sn	Sp	MCC
C w/S	93.8 (93.75)	69.2 (76.92)	100 (98.03)	0.8 (0.79)
C	95.3 (93.75)	76.9 (76.92)	100 (98.03)	0.85 (0.79)
R w/S	95.3 (95.31)	84.6 (84.61)	98 (98.03)	0.85 (0.85)
R	92.2 (95.31)	76.9 (84.61)	96.1 (98.03)	0.75 (0.85)

Following are the results of isGPT for different testing methods on new dataset.

Table 4.34: Results of 10-fold cross validation on new dataset.

Type	10-fold Cross Validation			
	Acc	Sn	Sp	MCC
C w/S	96.33	93.01	99.65	0.92
C	81.75	94.03	21.66	0.2
R w/S	99.31	99.31	99.31	0.98
R	83.59	99.31	6.66	0.17

Table 4.35: Result of independent testing on new dataset.

Type	Independent Test			
	Acc	Sn	Sp	MCC
C w/S	81.77	94.89	17.5	0.17
C	82.62	95.4	20	0.22
R w/S	83.47	100	2.5	0.14
R	83.89	99.48	7.5	0.2

4.3.1 Comparison between different Methods

Table 4.36: Performance comparison of different methods.

Method	Jackknife Cross Validation				Independent Testing				10-fold Cross Validation			
	Acc	Sn	Sp	MCC	Acc	Sn	Sp	MCC	Acc	Sn	Sp	MCC
Ahmad	83.77	12.5	98.33	0.22	77.96	17.5	**90.3**	0.09	82.07	14.59	95.83	0.16
Yang	-	-	-	-	-	-	-	-	94.4	91	93	.851
isGPT	-	-	-	-	**81.77**	**94.89**	17.5	**0.17**	**96.33**	**93.01**	**99.65**	**0.92**

Chapter 5

Discussion

From the results we can see that both Yang's method and isGPT method on existing dataset has given the same results as before. But Ahmad's method has not given the same result as before. May it is because of the k-nearest neighbour. May be Ahmad has implemented differently than ours. If we see Yang's method in case of 3-gapped dipeptide composition it has given result as before. But in case of Ahmad it has not given the same results as before. The difference between the procedure of Yang's and Ahmad's method is that they have used different feature selection method and different classifier. So, we can say that because of different implementation of k-nearest neighbour the results have been different for ours.

In case of new dataset isGPT has outperfomed both Yang and Ahmad. Yang's method has given satisfactory results also. But the result of Ahmad is not still satisfactory.

Chapter 6

Proposed Methodology

Following improvements will be done.

6.1 Position-Specific Scoring Matrix Based Features

6.2 Sliding Window Concept

6.3 Structural Features

6.4 Split Amino Acid Composition Based

6.5 Pseudo-Amino Acid Composition

Chapter 7

Conclusion

In this study we have reimplemented sub-golgi identification related works and performed the same steps with larger datasets to analyze the performance of existing methods. As we can see from the comparison of our results,our prediction result was not totally accurate according to the published papers.Although it was close. When we implemented our newly equipped data we could see we are having higher accuracy and sensitivity but lower specificity.For instance, in case of highly imbalance data where more than 90% of the data belong to a single class; a classifier may be overfit and predict all the test samples in the majority class. The accuracy in this case would be more than 90% because of high 'True Positive' rate but the system is erroneous and is unacceptable. MCC assess the overall quality of the classification and produce results in 1 and +1 range where 1 denotes that the classifier is consistently predicting the wrong classes and +1 means that the classifier has perfectly predicted whole sample accurately. A value near 0 dictates that the prediction system is producing average random predictions and the resulting precision, therefore, becomes unreliable. MCC is a very effective measure specially in case of biased dataset since high accuracy count with 0 MCC expose the true quality of the prediction system.As per our result we have to improve our specificity and MCC with better feature extraction.The proposed methodology might improve the performance yer we have to test them on both of existing and newly equipped datasets.And that would be our new objective.

References

[1] S. W. Rhee, T. Starr, K. Forsten-Williams, and B. Storrie, "The steady-state distribution of glycosyltransferases between the golgi apparatus and the endoplasmic reticulum is approximately 90: 10," *Traffic*, vol. 6, no. 11, pp. 978–990, 2005.

[2] S. R. Pfeffer, "Constructing a golgi complex," *J Cell Biol*, vol. 155, no. 6, pp. 873–876, 2001.

[3] Y. Fujita and K. Okamoto, "Golgi apparatus of the motor neurons in patients with amyotrophic lateral sclerosis and in mice models of amyotrophic lateral sclerosis," *Neuropathology*, vol. 25, no. 4, pp. 388–394, 2005.

[4] Y. Fujita, E. Ohama, M. Takatama, S. Al-Sarraj, and K. Okamoto, "Fragmentation of golgi apparatus of nigral neurons with α-synuclein-positive inclusions in patients with parkinson's disease," *Acta neuropathologica*, vol. 112, no. 3, pp. 261–265, 2006.

[5] N. Gonatas, J. O. Gonatas, and A. Stieber, "The involvement of the golgi apparatus in the pathogenesis of amyotrophic lateral sclerosis, alzheimer's disease, and ricin intoxication," *Histochemistry and cell biology*, vol. 109, no. 5-6, pp. 591–600, 1998.

[6] C.-H. Leung, H.-J. Zhong, D. S.-H. Chan, and D.-L. Ma, "Bioactive iridium and rhodium complexes as therapeutic agents," *Coordination Chemistry Reviews*, vol. 257, no. 11-12, pp. 1764–1776, 2013.

[7] D.-L. Ma, H.-Z. He, K.-H. Leung, D. S.-H. Chan, and C.-H. Leung, "Bioactive luminescent transition-metal complexes for biomedical applications," *Angewandte Chemie International Edition*, vol. 52, no. 30, pp. 7666–7682, 2013.

[8] B. Y.-W. Man, H.-M. Chan, C.-H. Leung, D. S.-H. Chan, L.-P. Bai, Z.-H. Jiang, H.-W. Li, and D.-L. Ma, "Group 9 metal-based inhibitors of β-amyloid (1–40) fibrillation as potential therapeutic agents for alzheimer's disease," *Chemical Science*, vol. 2, no. 5, pp. 917–921, 2011.

[9] T. Nakamura and S. A. Lipton, “Protein s-nitrosylation as a therapeutic target for neurodegenerative diseases,” *Trends in pharmacological sciences*, vol. 37, no. 1, pp. 73–84, 2016.

[10] J. Brettschneider, K. Del Tredici, V. M.-Y. Lee, and J. Q. Trojanowski, “Spreading of pathology in neurodegenerative diseases: a focus on human studies,” *Nature Reviews Neuroscience*, vol. 16, no. 2, p. 109, 2015.

[11] D. Ungar, “Golgi linked protein glycosylation and associated diseases,” in *Seminars in cell & developmental biology*, vol. 20, pp. 762–769, Elsevier, 2009.

[12] A. Nakano and A. Luini, “Passage through the golgi,” *Current opinion in cell biology*, vol. 22, no. 4, pp. 471–478, 2010.

[13] F. Ali and M. Hayat, “Classification of membrane protein types using voting feature interval in combination with chou s pseudo amino acid composition,” *Journal of theoretical biology*, vol. 384, pp. 78–83, 2015.

[14] K.-C. Chou and H.-B. Shen, “Recent progress in protein subcellular location prediction.,” *Analytical biochemistry*, vol. 370, no. 1, p. 1, 2007.

[15] G.-P. Zhou and K. Doctor, “Subcellular location prediction of apoptosis proteins,” *Proteins: Structure, Function, and Bioinformatics*, vol. 50, no. 1, pp. 44–48, 2003.

[16] K.-C. Chou and H.-B. Shen, “Cell-ploc: a package of web servers for predicting subcellular localization of proteins in various organisms,” *Nature protocols*, vol. 3, no. 2, p. 153, 2008.

[17] H. Lin, H. Wang, H. Ding, Y.-L. Chen, and Q.-Z. Li, “Prediction of subcellular localization of apoptosis protein using chou’s pseudo amino acid composition,” *Acta biotheoretica*, vol. 57, no. 3, pp. 321–330, 2009.

[18] H. Lin, H. Ding, F.-B. Guo, and J. Huang, “Prediction of subcellular location of mycobacterial protein using feature selection techniques,” *Molecular diversity*, vol. 14, no. 4, pp. 667–671, 2010.

[19] K.-C. Chou, “Prediction and classification of α-turn types,” *Biopolymers: Original Research on Biomolecules*, vol. 42, no. 7, pp. 837–853, 1997.

[20] X. Xiao, P. Wang, and K.-C. Chou, “Predicting the quaternary structure attribute of a protein by hybridizing functional domain composition and pseudo amino acid composition,” *Journal of Applied Crystallography*, vol. 42, no. 2, pp. 169–173, 2009.

[21] X. Xiao, P. Wang, and K.-C. Chou, “Quat-2l: a web-server for predicting protein quaternary structural attributes,” *Molecular Diversity*, vol. 15, no. 1, pp. 149–155, 2011.

[22] X.-B. Zhou, C. Chen, Z.-C. Li, and X.-Y. Zou, "Using chou's amphiphilic pseudo-amino acid composition and support vector machine for prediction of enzyme subfamily classes," *Journal of theoretical biology*, vol. 248, no. 3, pp. 546–551, 2007.

[23] J.-D. Qiu, J.-H. Huang, S.-P. Shi, and R.-P. Liang, "Using the concept of chou's pseudo amino acid composition to predict enzyme family classes: an approach with support vector machine based on discrete wavelet transform," *Protein and peptide letters*, vol. 17, no. 6, pp. 715–722, 2010.

[24] G.-Y. Zhang and B.-S. Fang, "Predicting the cofactors of oxidoreductases based on amino acid composition distribution and chou's amphiphilic pseudo-amino acid composition," *Journal of Theoretical Biology*, vol. 253, no. 2, pp. 310–315, 2008.

[25] Y.-D. Cai, G.-P. Zhou, and K.-C. Chou, "Support vector machines for predicting membrane protein types by using functional domain composition," *Biophysical journal*, vol. 84, no. 5, pp. 3257–3263, 2003.

[26] K.-C. Chou and H.-B. Shen, "Memtype-2l: a web server for predicting membrane proteins and their types by incorporating evolution information through pse-pssm," *Biochemical and biophysical research communications*, vol. 360, no. 2, pp. 339–345, 2007.

[27] H. Lin, "The modified mahalanobis discriminant for predicting outer membrane proteins by using chou's pseudo amino acid composition," *Journal of theoretical biology*, vol. 252, no. 2, pp. 350–356, 2008.

[28] X. Xiao, P. Wang, and K.-C. Chou, "Gpcr-ca: A cellular automaton image approach for predicting g-protein–coupled receptor functional classes," *Journal of computational chemistry*, vol. 30, no. 9, pp. 1414–1423, 2009.

[29] W.-Z. Lin, X. Xiao, and K.-C. Chou, "Gpcr-gia: a web-server for identifying g-protein coupled receptors and their families with grey incidence analysis," *Protein Engineering, Design & Selection*, vol. 22, no. 11, pp. 699–705, 2009.

[30] K.-C. Chou and H.-B. Shen, "Protident: A web server for identifying proteases and their types by fusing functional domain and sequential evolution information," *Biochemical and Biophysical Research Communications*, vol. 376, no. 2, pp. 321–325, 2008.

[31] G.-P. Zhou and Y.-D. Cai, "Predicting protease types by hybridizing gene ontology and pseudo amino acid composition," *PROTEINS: Structure, Function, and Bioinformatics*, vol. 63, no. 3, pp. 681–684, 2006.

[32] K.-C. Chou, "A vectorized sequence-coupling model for predicting hiv protease cleavage sites in proteins.," *Journal of Biological Chemistry*, vol. 268, no. 23, pp. 16938–16948, 1993.

[33] K.-C. Chou, "Prediction of human immunodeficiency virus protease cleavage sites in proteins," *Analytical biochemistry*, vol. 233, no. 1, pp. 1–14, 1996.

[34] K.-C. Chou and H.-B. Shen, "Signal-cf: a subsite-coupled and window-fusing approach for predicting signal peptides," *Biochemical and biophysical research communications*, vol. 357, no. 3, pp. 633–640, 2007.

[35] J. A. Hiss and G. Schneider, "Architecture, function and prediction of long signal peptides," *Briefings in bioinformatics*, vol. 10, no. 5, pp. 569–578, 2009.

[36] K.-C. Chou, "Structural bioinformatics and its impact to biomedical science," *Current medicinal chemistry*, vol. 11, no. 16, pp. 2105–2134, 2004.

[37] H.-B. Shen and K.-C. Chou, "Nuc-ploc: a new web-server for predicting protein subnuclear localization by fusing pseaa composition and psepssm," *Protein Engineering, Design & Selection*, vol. 20, no. 11, pp. 561–567, 2007.

[38] H.-B. Shen and K.-C. Chou, "Predicting protein subnuclear location with optimized evidence-theoretic k-nearest classifier and pseudo amino acid composition," *Biochemical and Biophysical Research Communications*, vol. 337, no. 3, pp. 752–756, 2005.

[39] Z. Lei and Y. Dai, "Assessing protein similarity with gene ontology and its use in subnuclear localization prediction," *BMC bioinformatics*, vol. 7, no. 1, p. 491, 2006.

[40] H. Ding, S.-H. Guo, E.-Z. Deng, L.-F. Yuan, F.-B. Guo, J. Huang, N. Rao, W. Chen, and H. Lin, "Prediction of golgi-resident protein types by using feature selection technique," *Chemometrics and Intelligent Laboratory Systems*, vol. 124, pp. 9–13, 2013.

[41] Y.-S. Jiao and P.-F. Du, "Predicting golgi-resident protein types using pseudo amino acid compositions: Approaches with positional specific physicochemical properties," *Journal of theoretical biology*, vol. 391, pp. 35–42, 2016.

[42] J. Ahmad, F. Javed, and M. Hayat, "Intelligent computational model for classification of sub-golgi protein using oversampling and fisher feature selection methods," *Artificial intelligence in medicine*, vol. 78, pp. 14–22, 2017.

[43] R. Yang, C. Zhang, R. Gao, and L. Zhang, "A novel feature extraction method with feature selection to identify golgi-resident protein types from imbalanced data," *International journal of molecular sciences*, vol. 17, no. 2, p. 218, 2016.

[44] Y. Huang, B. Niu, Y. Gao, L. Fu, and W. Li, "Cd-hit suite: a web server for clustering and comparing biological sequences," *Bioinformatics*, vol. 26, no. 5, pp. 680–682, 2010.

[45] M. S. Rahman, M. K. Rahman, M. Kaykobad, and M. S. Rahman, "isgpt: An optimized model to identify sub-golgi protein types using svm and random forest based feature selection," *Artificial intelligence in medicine*, vol. 84, pp. 90–100, 2018.

[46] L. Michaelis and M. L. Menten, *Die kinetik der invertinwirkung*. Universitätsbibliothek Johann Christian Senckenberg, 2007.

[47] C.-T. Zhang and K.-C. Chou, "An optimization approach to predicting protein structural class from amino acid composition," *Protein Science*, vol. 1, no. 3, pp. 401–408, 1992.

[48] R. Kaundal, R. Saini, and P. X. Zhao, "Combining machine learning and homology-based approaches to accurately predict subcellular localization in arabidopsis," *Plant physiology*, vol. 154, no. 1, pp. 36–54, 2010.

[49] H. Lin and H. Ding, "Predicting ion channels and their types by the dipeptide mode of pseudo amino acid composition," *Journal of theoretical biology*, vol. 269, no. 1, pp. 64–69, 2011.

[50] K.-C. Chou, "Prediction of protein cellular attributes using pseudo-amino acid composition," *Proteins: Structure, Function, and Bioinformatics*, vol. 43, no. 3, pp. 246–255, 2001.

[51] J. X. Guo and N. N. Rao, "The influence of dipeptide composition on protein folding rates," in *Advanced materials research*, vol. 378, pp. 157–160, Trans Tech Publ, 2012.

[52] X. Zhao, X. Li, Z. Ma, and M. Yin, "Prediction of lysine ubiquitylation with ensemble classifier and feature selection," *International Journal of Molecular Sciences*, vol. 12, no. 12, pp. 8347–8361, 2011.

[53] B. Liu, J. Xu, Q. Zou, R. Xu, X. Wang, and Q. Chen, "Using distances between top-n-gram and residue pairs for protein remote homology detection," in *BMC bioinformatics*, vol. 15, p. S3, Springer, 2014.

[54] C. N. Magnan, A. Randall, and P. Baldi, "Solpro: accurate sequence-based prediction of protein solubility," *Bioinformatics*, vol. 25, no. 17, pp. 2200–2207, 2009.

[55] J. A. Capra and M. Singh, "Predicting functionally important residues from sequence conservation," *Bioinformatics*, vol. 23, no. 15, pp. 1875–1882, 2007.

[56] A. A. Schäffer, L. Aravind, T. L. Madden, S. Shavirin, J. L. Spouge, Y. I. Wolf, E. V. Koonin, and S. F. Altschul, "Improving the accuracy of psi-blast protein database searches with composition-based statistics and other refinements," *Nucleic acids research*, vol. 29, no. 14, pp. 2994–3005, 2001.

[57] S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman, "Gapped blast and psi-blast: a new generation of protein database search programs," *Nucleic acids research*, vol. 25, no. 17, pp. 3389–3402, 1997.

[58] M. Hayat, M. Tahir, and S. A. Khan, "Prediction of protein structure classes using hybrid space of multi-profile bayes and bi-gram probability feature spaces," *Journal of theoretical biology*, vol. 346, pp. 8–15, 2014.

[59] Y.-C. Zuo, Y. Peng, L. Liu, W. Chen, L. Yang, and G.-L. Fan, "Predicting peroxidase subcellular location by hybridizing different descriptors of chou'pseudo amino acid patterns," *Analytical biochemistry*, vol. 458, pp. 14–19, 2014.

[60] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[61] C. Ding, L.-F. Yuan, S.-H. Guo, H. Lin, and W. Chen, "Identification of mycobacterial membrane proteins and their types using over-represented tripeptide compositions," *Journal of proteomics*, vol. 77, pp. 321–328, 2012.

[62] T. Ebina, R. Suzuki, R. Tsuji, and Y. Kuroda, "H-drop: an svm based helical domain linker predictor trained with features optimized by combining random forest and stepwise selection," *Journal of computer-aided molecular design*, vol. 28, no. 8, pp. 831–839, 2014.

[63] Q. Gu, Z. Li, and J. Han, "Generalized fisher score for feature selection," *arXiv preprint arXiv:1202.3725*, 2012.

[64] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[65] K. K. Kandaswamy, G. Pugalenthi, E. Hartmann, K.-U. Kalies, S. Möller, P. Suganthan, and T. Martinetz, "Spred: A machine learning approach for the identification of classical and non-classical secretory proteins in mammalian genomes," *Biochemical and biophysical research communications*, vol. 391, no. 3, pp. 1306–1311, 2010.

[66] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[67] M. Waris, K. Ahmad, M. Kabir, and M. Hayat, "Identification of dna binding proteins using evolutionary profiles position specific scoring matrix," *Neurocomputing*, vol. 199, pp. 154–162, 2016.

[68] P. Hall, B. U. Park, R. J. Samworth, *et al.*, "Choice of neighbor order in nearest-neighbor classification," *The Annals of Statistics*, vol. 36, no. 5, pp. 2135–2152, 2008.

[69] Y. Jiao and P. Du, "Performance measures in evaluating machine learning based bioinformatics predictors for classifications," *Quantitative Biology*, vol. 4, no. 4, pp. 320–330, 2016.

[70] D. G. Altman and J. M. Bland, "Diagnostic tests. 1: Sensitivity and specificity.," *BMJ: British Medical Journal*, vol. 308, no. 6943, p. 1552, 1994.

[71] G. Atkinson and A. M. Nevill, "Statistical methods for assessing measurement error (reliability) in variables relevant to sports medicine," *Sports medicine*, vol. 26, no. 4, pp. 217–238, 1998.

[72] K.-C. Chou and C.-T. Zhang, "Prediction of protein structural classes," *Critical reviews in biochemistry and molecular biology*, vol. 30, no. 4, pp. 275–349, 1995.

[73] C.-J. Zhang, H. Tang, W.-C. Li, H. Lin, W. Chen, and K.-C. Chou, "iori-human: identify human origin of replication by incorporating dinucleotide physicochemical properties into pseudo nucleotide composition," *Oncotarget*, vol. 7, no. 43, p. 69783, 2016.

[74] R. Kohavi, D. Sommerfield, and J. Dougherty, "Data mining using/spl mscr//spl lscr//spl cscr/++ a machine learning library in c++," in *Proceedings Eighth IEEE International Conference on Tools with Artificial Intelligence*, pp. 234–245, IEEE, 1996.

[75] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Ijcai*, vol. 14, pp. 1137–1145, Montreal, Canada, 1995.

[76] D. M. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," 2011.

Generated using Undegraduate Thesis LaTeX Template, Version 1.4. Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh.

This thesis was generated on Friday 7th February, 2020 at 5:45am.