Predicting Human and Animal Protein Subcellular Location

Sepideh Khavari[*] Xiangjia Min[†] James D. Munyon[‡] Guang-Hwa Chang[§]

Abstract

An important objective in cell biology is to determine the subcellular location of different proteins and their functions in the cell. Identifying the subcellular location of proteins can be accomplished either by using biochemical experiments or by developing computational predictors that aid in predicting the subcellular location of proteins. The main objective of this study is to use several different classifiers to predict the subcellular location of animal and human proteins and to determine which of these classifiers performs the best in predicting protein subcellular location. The data for this study was obtained from The Universal Protein Resource (UniProt) which is a database of protein sequence and annotation.A reliable benchmark dataset is obtained by following and applying criteria established in earlier studies for predicting protein subcellular locations. After applying the above criteria to the original dataset, the working benchmark dataset includes 2944 protein sequences. The method used for representing proteins in the study is the pseudo-amino acid composition (PseAA composition) adapted from earlier studies. The predictors used to predict the subcellular location of proteins in animal and human include Random Forest, Adaptive Boosting (AdaBoost), and Stagewise Additive Modeling using a Multi-class Exponential loss function (SAMME), Support Vector Machines (SVMs), and Artificial Neural Networks (ANNs). The results from this study establish that the SVMs classifier yielded the best overall accuracy for predicting the subcellular location of proteins. Most of the computational classifiers used in this study produced better prediction results for determining the subcellular location of proteins in the nucleus, the secreted, and the cell membrane. The secreted and the cell membrane locations had high specificity values with all of the classifiers used in this study. The nucleus had the best prediction results, including a high sensitivity and a high MCC value by using the Bagging method.

Key Words: Protein subcellular location prediction, Bagging, Random forest, AdaBoost, SAMME, Support vector machines, Neural networks

1. Introduction

A cell is the smallest unit of any life form. The size of a cell can vary from $1 - 5\mu$m, in prokaryotic cells, to a size of greater than 50μm, in eukaryotic cells [25]. Cells perform many diverse and important functions necessary for the survival of all living beings. There are many smaller different components and organelles inside a cell that are responsible for carrying out these diverse and important functions. The organelles inside a cell have specific functions and nearly all of them are surrounded by membranes that have embedded proteins.

The functions that are performed by the organelles are vital to the cell's survival. These functions are carried out by the proteins embedded in the organelles or other compartments in the cell. On average, a cell contains 10^9 proteins located in different parts of the cell[16].

The location of proteins in the cell is referred to as the "subcellular location" in the literature. An important objective in cell biology and proteomics is to determine the subcellular

[*] Youngstown State University, Youngstown, Ohio 44555; skhavari@ysu.edu

[†] Youngstown State University, Youngstown, Ohio 44555; xmin@ysu.edu

[‡] Bowling Green State University, Bowling Green, Ohio 43403-0001; jmunyon@bgsu.edu

[§] Youngstown State University, Youngstown, Ohio 44555; gchang@ysu.edu

location of different proteins and their functions in the cell. The knowledge gained about the location of proteins also can help in determining the specific functions they carry out for the cell's survival [16].

Identifying the subcellular location of proteins can be accomplished by using two methods. The first method is using biochemical experiments in the lab to locate proteins in the cell. However, the drawback to this approach is that it can be both very time-consuming and expensive. Given that the number of newly discovered proteins is increasing so quickly, as a result of genome sequencing project, and that the identification of their subcellular location lags behind, this approach is not a very efficient approach to solving the protein sublocalization problem.

The second method is developing computational predictors that help in predicting the subcellular location of proteins [16]. The main objective of this study is to use several different classifiers to predict the subcellular location of animal and human proteins and to determine which of these classifiers performes the best in predicting protein subcellular location.

2. Background and Literature Review

In the post-genomic era, there has been a significant increase in the number of newly discovered proteins. Given the need to predict the subcellular location of these proteins, there have been many approaches in the literature to address protein sublocalization problem.

Zheng Yuan [31] uses the Markov Chains to predict protein subcellular location. The author points out that by using the Jack-knife test the prediction accuracy is 8% greater than using the neural network method and incorporating amino acid composition [25].

Chou and Shen [16] discuss their review of various methods of prediction available to identify protein subcelluar locations. Their paper examines how the problem of predicting protein sublocation can be viewed in terms of how proteins are represented and what type of algorithm can be used to produce the most accurate prediction. The authors point out that proteins can be represented in various ways such as sequential; non-sequential (amino acid composition and pseudo-amino acid composition [PseAA]); the Functional domain (FunD) discrete model, which is a representation by functional domain since function of a protein is related to its subcellular location; the Gene Ontology (GO) discrete model, which is a representation of protein that is defined in GO database– proteins in a GO database are clustered in such a way that is reflective of their subcellular location; and the hybridization discrete model which is a combination of GO discrete model and PseAA [16].

Next, the authors discuss the algorithms used to predict protein subcellular location and the testing methods incorporated to test the accuracy of each algorithm. The covariant discriminant (CD), the K nearest neighbor (KNN), the optimized evidence-theoretic K nearest neighbor (OET-KNN), and ensemble classifiers were extensively discussed as examples of prediction algorithms. For testing the quality of prediction algorithms, the authors investigated the self-consistency examination, the cross-validation examination, and the Jack-knife examination [16]. A review of protein sequence representation and prediction algorithms for this paper, which is based upon the paper by Chou et. al. [16], will be presented in the following section.

Finally, the authors survey the available prediction methods that have been placed on web servers and are available for free to the general public. The prediction web servers have been classified in terms of the type of eukaryotic organism for which they perform the prediction of protein subcellular location [16].

Cai et al.[30] uses Support Vector Machines (SVMs) to predict the subcellular location of proteins for 12 different locations in the cell. The authors conclude that both the self-consistency and the Jackknife tests have high prediction accuracies, ranging from 75% to 94%, for these locations. Park and Kanehisa [15] also have used SVMs on 12 subcellular locations in eukaryotic cells. They report that, by using the RBF kernel with SVMs, the total accuracy of prediction is 72.4% and the location accuracy is 54.6%. The authors also compare their results to the results obtained by Cai et al. [30]. Park and Kanehisa conclude that, with the use of 5-fold cross validation, their accuracy results are better than the results of the Jackknife test obtained by Cai et al. Another more recent study using SVMs is performed by Dehzangi et al. [1] to predict the subcellular location of proteins in Gram-positive and Gram-negative bacteria. The findings of this study show that, by using 10-fold cross validation, the overall accuracies for prediction of the subcellular locations of proteins in Gram-positive and Gram-negative bacteria are 87.7% and 79.6% respectively.

Reinhardt and Hubbard [23] use neural networks to classify the subcellular location of proteins in prokaryotic and eukaryotic cells. Their results reveal that the accuracy of classification in prokaryotic cells for three subcellular locations is 81%. The accuracy of classification for eukaryotic cells for predicting protein in four subcellular locations is 66%. Singh et al. [3] also use the neural networks classification method to predict plant protein subcellular locations. In their work, they used various methods to represent proteins including amino acid composition and dipeptide composition. For predicting the protein subcellular location, they solve the classification problem by performing multiple binary classification predictions. In the final step, they combine all of the binary classifiers to arrive at a final classifier. They conclude that, with the Pseudo Amino Acid Composition, the overall accuracy of classification using their neural network model is 75%. Furthermore, the overall accuracy of prediction they obtained is more efficacious than the performance of web tools such as YLOC+ and Euk-mPloc for protein subcellular prediction.

Chou and Shen [17] have introduced a web-server called Euk-mPLoc 2.0, which is a hybrid model of gene ontology (GO) information, functional domain information, and sequential evolutionary information, using different forms of pseudo-amino acid composition. The authors point out that this predictor is able to predict the subcellular location of proteins in 22 locations inside the cell. They note that Euk-mPLoc 2.0 is a very powerful predictor in predicting eukaryotic proteins that reside in multiple locations inside the cell [16].

Other papers have also evaluated the current computational methods of prediction that have been proposed to predict protein subcellular location and are available freely for public use. Min [19] discusses the prediction accuracy of various software tools in predicting the secretomes of Eukaryotes. Prediction accuracy was reflected by using the Mathews' Correlation Coefficient in this paper. His research indicates that there is no one single software that can result in the highest accuracy in protein subcellular location prediction in all Eukaryotes. Min concludes that, among the tools tested, Phobius is best for animal protein prediction, SignalP is best for plant protein prediction, and WoLF PSORT is best for fungal protein prediction [19].

Sprenger et.al. [14] have evaluated prediction methods under the criteria that predictions tools accept large amount of protein sequence, are publicly available, and are also able to predict protein locations in at least nine subcellular locations. The prediction methods surveyed and evaluated included CELLO, MultiLoc, Proteome Analyst, pTARGET and WoLF PSORT. The sources of data used in this study are Swiss Prot and LOCATE. The authors calculate sensitivity and specificity data analyzed by each prediction method and then compare the results to the outcome represented by random chance. They concluded that the prediction methods did not show a level of sensitivity on either data set to prove to be a reliable prediction method for predicting subcellular location of new proteins. In addition, the authors pointed out that the prediction methods produced a lower accuracy for the data from LOCATE [14].

3. Methods

3.1 Protein Representation Method

In order to predict the subcellular location of proteins, it is important to employ a method for representing a protein. There are multiple approaches in the literature such as sequential and non-sequential methods, for representing the protein samples. The method used in this study to represent protein data is the pseudo-amino acid composition (PseAA composition) method, which has been adopted from [16]. PseAA composition is a non-sequential method of representation of proteins. It has an advantage over the sequential method of representation in that the sequential method doesn't perform well when a protein of interest has little homology to proteins of known location [16]. It also has an advantage over Amino Acid composition method (AA composition), because in the AA composition method, the order of sequence in the protein is lost [16].

3.2 Dataset

The data for this study was obtained from The Universal Protein Resource (UniProt) which is a database of protein sequence and annotation [9]. The proteins of interest in this paper are animal and human proteins. Therefore, by accessing UniProt Knowledgebase (UniProt KB) [9], the human and animal proteins that were manually reviewed and annotated (Swiss-Prot) were chosen for this study.

In order to obtain a reliable benchmark dataset the following criteria were further applied to the obtained data, based upon the suggestions found in the literature for benchmark data set:

- All proteins in this dataset are reviewed and annotated for subcellular location.
- All proteins in the dataset are annotated only for one subcellular location(protein sequences that have multiple subcellular locations have been excluded from the dataset).
- Protein sequences with vague and uncertain labels such as “probable” or “by similarity” have been excluded.
- Proteins that contain unknown amino acids in their sequence and are marked by **X** have been excluded
- Proteins with the label ‘fragment’ are excluded from the dataset (‘fragments’ are those protein sequences that do not start with the letter **M**- referring to the amino acid methionine).
- Proteins in this dataset have at least 100 amino acids in their sequence. Any protein with less than 100 amino acids in their sequence have been excluded from this dataset.

After applying the above criteria to the original dataset, "$50/50$" BLASTClust [11] is used on the dataset. The purpose of using "$50/50$" BLASTClust is to cluster similar proteins together and to choose one proteins at random in each cluster for the final dataset. The rule for similarity of proteins is that if two or more proteins are at least 50% similar over at least 50% of their length, then they belong to the same cluster. After applying "$50/50$" BLASTClust to the dataset and removing proteins that have 'NA' marked as their locations, the working benchmark dataset includes 2944 protein sequences in 9 different subcellular locations.

In the next step, by adopting the [22] and [20] method of partitioning the benchmark dataset, the benchmark dataset is partitioned into a 70% training (learning) dataset and a 30% independent testing dataset. The training dataset includes a total of 2060 proteins. The subcellular locations of these proteins are in the cell membrane (210 proteins), the cytoplasm (369), the endoplasmic reticulum (145 proteins), the Golgi apparatus (63 proteins), the lysosome (21 proteins), the mitochondria (236 proteins), the nucleus (702 proteins), the peroxisome (20 proteins), the secreted (294 proteins). Also, the testing dataset includes a total of 884 proteins. The subcellular locations of these proteins are in the cell membrane (76 proteins), the cytoplam (171), the endoplasmic reticulum (62 proteins), the Golgi apparatus (23 proteins), the lysosome (9 proteins), the mitochondria (92 proteins), the nucleus (299 proteins), the peroxisome (10 proteins), the secreted (142 proteins).

3.3 Classification Methods

The classifications methods used in this study include the following ensemble learning methods: Random Forest, AdaBoost, and SAMME. In addition to the ensemble methods, this study uses Support Vector Machines (SVMs) and Artificial Neural Networks (ANNs) to predict the subcellular location of proteins in human and animals. This paper will first provide a brief review of these and related classification methods, present their respective algorithms, and examine each method's advantages and disadvantages. Next, each method of classification is applied in predicting the subcellular location of proteins in this study. Finally, the results obtained from each classification method will be discussed. The R software environment and other R packages are used to apply the classification methods to predict the protein subcellular location [26].

3.3.1 Bootstrap Aggregating (Bagging)

Bootstrap Aggregating or Bagging is a machine learning method that uses an ensemble of predictors to solve classification problems. In "Bagging Predictors," Leo Breiman describes Bagging as a method of classification in which multiple versions of a predictor are generated and used to achieve an aggregate predictor. The multiple versions of a predictor are generated by making bootstrap replicates of the training set and using these replicates as new training set. The aggregate predictor averages over the multiple versions when it is predicting a numerical value and searches for a popular (majority) vote to predict a class [4]. Breiman points out that when Bagging is used on real data, it has high accuracy in solving classification problems.

3.3.2 Random Forests

Random Forest is an ensemble of decision trees each voting for a class. The most popular vote among these trees is then chosen as the final class. The 2001 Paper "Random Forest" by

Leo Breiman first introduced Random Forest as an ensemble of predictors (decision trees) where the most popular vote among the trees determines the class of an object [6]. Breiman defines Random Forest as "a classifier consisting of a collection of tree-structured classifiers $h(x, \Theta_k), k = 1, ...$ where the Θ_k are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input x" [6]. Breiman further explains the procedure of Random Forest by noting that "for the kth tree, a random vector Θ_k is generated independent of past vectors $\theta_1, \theta_2, \ldots, \theta_{k-1}$ but with the same distribution. A tree is grown using the training set and the random vector Θ_k which results in a classifier $h(x, \Theta_k)$" [6]. By following this procedure, one can construct a large number of trees. These trees will each vote for a class and the most popular class is chosen as the final class [6].

Also, according to Breiman, the Random Forest has an accuracy competitive with and sometimes better than Adaboost, its accuracy is not affected by outliers and noise in the data, the algorithms run faster than bagging or boosting, and they generate out-of-bag estimates which include the internal estimate of error, strength, and correlation [6].

3.3.3 Adaptive Boosting (Adaboost)

Yoav Freund and Robert E. Schapire introduce a boosting algorithm called "Adaboost" that improves the accuracy of classifiers immensely [28]. In other words, Adaboost combines many weak classifiers to achieve a strong and very accurate classifier as the final classifier. Adaboost works by starting with a base algorithm and the training data set. Equal weights are assigned to each data point in the data set, and then the algorithm is run which results in a new classifier. In the following step, for all of the data points that were classified incorrectly, the weight is increased and for all of the data points that were correctly classified, the weight is decreased, then the algorithm is run again giving another classifier. This process is repeated until all of the data points in the training data set are classified with 100 percent accuracy. Finally, all of the classifiers in the process are combined as a linear combination to produce a final classifier with a very high accuracy in classification. It is important to note that the Adaboost method is a great for solving two-class problems. If the problem of classification is extended to include multiple-class classification, the problem will be reduced to solving two-class problems using Adaboost.

3.3.4 Stagewise Additive Model using a Multi-class Exponential loss function (SAMME)

Adaboost is a great classification ensemble for two-class classification problems. However, when dealing with multi-class problems, Adaboost approaches solving the classification by breaking down the multi-class problem into multiple two-class classification problems. The Stagewise Additive Model using a Multi-class Exponential loss function or in short SAMME, improves on Adaboost by solving the multi-class classification problems without condensing them into multiple two-class problems. SAMME combines weak classifiers and requires the performance of each classifier to be better than random guessing [12].

3.3.5 Support Vector Machines (SVMs)

Support Vector Machines (SVMs) classify data into groups by finding the optimal hyperplane that maximizes the margin of separation between classes. Although SVMs are binary computational predictors, the binary predictors can be combined to achieve a final multi-class classifier [8].

In a classification problem, the dataset can be partitioned into training and testing datasets. Each data point in the training set has specific features, referred to as "attributes", and belongs to a specific class, referred to as “target value”. The goal of SVMs is to teach the data in the training set so that the SVM algorithm will be able to predict the correct “target value” for each data point in the testing set [7].

As far as the model selection for SVMs used to classify the protein subcellular locations in this paper, the RBF kernel was chosen as the kernel function in accordance with the reasoning provided by Hsu et. al[7]. For choosing the best parameter values for C and γ for the RBF kernel, the methods of k-fold cross validation and “grid search” are recommended by [7].

3.3.6 Artificial Neural Networks (ANNs)

A branch of artificial intelligence, artificial neural networks (ANNs) or neural networks, are inspired by the biological brain and the nervous system [24]. A neural network is comprised of a set of artificial neurons that are inter-connected in a manner similar to the neural connections in the human brain. ANNs are used for classification problems in areas such as bankruptcy detection, speech recognition, product inspection, and fault detection [24].

ANNs were first formulated by McCulloch and Pitts in 1943. In the 1960’s Rosenblatt introduced the Perceptron Convergence Theorem, while Minsky and Papert worked on showing the limitation of a simple perceptron. In the 1980’s, the interest in ANNs was renewed by Hopefield’s research and findings in this area. In addition, the back propagation learning algorithm for multilayer perceptron by Webros reinvigorated the field furthermore reinvigorated interest in ANNs [2].

As far as the architecture of an ANN, a standard neural network is comprised of three layers. An input layer, a hidden layer, and an output layer. All neurons in the network are identical in structure and include a sum and a function unit. When inputs are fed into the neural network, the network assigns weight for each input. Next, the weighted inputs are all summed up. The result of this step is used as an input for the transfer or activation function in the neuron. The output from the activation function is the output of the neuron [**?**].

4. Results

4.1 Key Terms

In analyzing the results of this study, there are several important key terms that will be presented in this section and incorporated later when discussing the results of this study. Therefore, this section will first define these terms before proceeding to present the results.

In order to analyze the performance of each prediction or classification method, the term “accuracy” is employed. The “accuracy” of a prediction or classification method describes the number or the percentage of correctly classified object in a dataset. Furthermore, there are various methods for assessing performance of a predictor or classifier such as sensitivity, specificity, balanced accuracy, and Mathew’s correlation coefficient (MCC).

The following are the formulas for sensitivity, specificity, balanced accuracy, and Mathew’s correlation coefficient [19]. Both “balanced accuracy” and the “Mathew’s correlation coefficient” are used in measuring the performance in classification in machine learning. “Balanced accuracy” is the arithmetic mean of sensitivity and specificity. While

MCC considers true and false positives and negatives and can be used with classes that have different sizes. The MCC values range between -1 and 1 with the MCC value of 1 indicating perfect prediction [18]. In the following formulas, TP represents the number of true positives, FN represents the number of false negative, FP represents the number of false positives, and TN represents the number of true negatives.

$$\text{Sensitivity \%} = [TP/(TP + FN)] * 100$$

$$\text{Specificity \%} = [TN/(TN + FP)] * 100$$

$$\text{Balanced Accuracy \%} = [(\text{Sensitivity} + \text{Specificity})/2] * 100$$

$$\text{Mathew's Correlation Coefficient \%} = \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} * 100$$

4.2 Bagging model

Table 1: Overall Statistics

Accuracy	0.44
95% CI	$(0.40, 0.47)$
No Information Rate	0.34
P-Value [Acc > NIR]	8.1×10^{-10}
Kappa	0.22

Table 2: Statistics by Class

	Sen	Spe	PPV	NPV	Pre	DR	DP	BA	MCC	NoP
Cell membrane	0.39	0.94	0.41	0.94	0.09	0.04	0.09	0.66	0.39	76
Cytoplasm	0.1	0.97	0.46	0.82	0.19	0.02	0.04	0.54	0.1	171
Endoplasmic Reticulum	0.29	0.98	0.54	0.94	0.07	0.02	0.04	0.63	0.29	62
Golgi Apparatus	0	1	NA	0.97	0.03	0	0	0.5	0	23
Lysosome	0	1	NA	0.99	0.01	0	0	0.5	0	9
Mitochondria	0.19	0.96	0.35	0.91	0.11	0.02	0.06	0.57	0.19	92
Nucleus	0.93	0.35	0.42	0.91	0.34	0.31	0.74	0.64	0.93	299
Peroxisome	0	1	NA	0.99	0.01	0	0	0.5	0	10
Secreted	0.16	0.99	0.79	0.87	0.15	0.02	0.03	0.58	0.16	142

Key for Abbreviated Measures:
Sen: Sensitivity, Spe: Specificity, PPV: Positive Predicted Value, NPV: Negative Predicted Value, Pr: Prevalence, DR: Detection Rate, DP: Detection Prevalence, BA: Balanced Accuracy, MCC: Matthews Correlation Coefficient, NoP: Number of Proteins

The two tables above summarize the results from the bagging model. The key terms for table 2 will be used as reference for analyzing results for the similar tables generated by other classifiers in this study in the following sections. Table 1 shows that the model can predict proteins to their subcellular location with a 44% accuracy. The Table 2 illustrates statistics by class and indicates the performance of the model using the testing data. The locations with a sensitivity value of 0 and a specificity value of 1 indicate that there are no proteins predicted to these subcellular locations.

The subcellular location with the best prediction in the model is the nucleus with an MCC of 0.93, a sensitivity of 93% and a specificity of 35%. The sensitivity measure for the nucleus indicates that the model correctly predicted nucleus location 93% of the time. The specificity measure conveys the fact that the model correctly predicted the absence of proteins in the nucleus 35% of the time. The balanced accuracy value for this location is 64%. Although the bagging method predicts the nucleus proteins with high sensitivity, further work is needed to improve the value of specificity. In addition, other subcellular locations have not had good predictions with this method.

4.3 Bagging with Cross Validation model

Table 3: Overall Statistics

Accuracy	0.42
95% CI	$(0.40, 0.44)$
No Information Rate	0.34
P-Value [Acc > NIR]	2.2×10^{-16}
Kappa	0.22

Table 4: Statistics by Class

	Sen	Spe	PPV	NPV	Pre	DR	DP	BA	MCC	NoP
Cell membrane	0.3	0.95	0.38	0.93	0.1	0.03	0.08	0.62	0.27	76
Cytoplasm	0.08	0.95	0.27	0.82	0.18	0.01	0.05	0.52	0.05	171
Endoplasmic Reticulum	0.3	0.97	0.45	0.95	0.07	0.02	0.05	0.64	0.33	62
Golgi Apparatus	0	1	NA	0.97	0.03	0	0	0.5	0	23
Lysosome	0	1	NA	0.99	0.01	0	0	0.5	0	9
Mitochondria	0.15	0.98	0.44	0.9	0.11	0.02	0.04	0.56	0.21	92
Nucleus	0.89	0.38	0.43	0.87	0.34	0.3	0.71	0.64	0.28	299
Peroxisome	0	1	NA	0.99	0.01	0	0	0.5	0	10
Secreted	0.25	0.95	0.46	0.88	0.15	0.04	0.08	0.6	0.26	142

Table 3 shows that the model can predict proteins to their subcellular location with a 42% accuracy. Table 4 illustrates statistics by class and indicates the performance of the model using the testing data. The locations with a sensitivity value of 0 and a specificity value of 1 indicate that there are no proteins predicted to these subcellular locations.

The ER location has the highest MCC value of 0.33, with a sensitivity of 30% and a specificity of 97%. Next, the nuclear subcellular location has an MCC value of 0.28. The sensitivity and specificity values for the nucleus are 89% and 38%. The balanced accuracy value for both of these locations are 64%. Similar to the bagging methods, other subcellular locations failed to perform as well as they had performed in the other models.

4.4 Random Forests Model

The results for Random Forest model is summarized in the following tables. The package "randomForest" in R which is based upon the Random Forest algorithm proposed by Leo Brieman was utilized. The results are based upon partitioning the dataset into a 70% training dataset and a 30% testing dataset. This "random forest" is comprised of 500 trees. (Each tree is built by using a random selection of variables).

Table 5: Overall Statistics

Accuracy	0.51
95% CI	$(0.49, 0.56)$
No Information Rate	0.34
P-Value [Acc > NIR]	2.2×10^{-6}
Kappa	0.36

Table 6: Statistics by Class

	Sen	Spe	PPV	NPV	Pre	DR	DP	BA	MCC	NoP
Cell membrane	0.49	0.95	0.49	0.95	0.09	0.04	0.09	0.72	0.44	76
Cytoplasm	0.27	0.92	0.45	0.84	0.19	0.05	0.12	0.6	0.24	171
Endoplasmic Reticulum	0.21	0.99	0.62	0.94	0.07	0.01	0.02	0.6	0.34	62
Golgi Apparatus	0	1	NA	0.97	0.03	0	0	0.5	0	23
Lysosome	0	1	NA	0.99	0.01	0	0	0.5	0	9
Mitochondria	0.33	0.95	0.42	0.92	0.1	0.03	0.08	0.64	0.3	92
Nucleus	0.89	0.58	0.52	0.91	0.34	0.3	0.58	0.73	0.45	299
Peroxisome	0	1	NA	0.99	0.01	0	0	0.5	0	10
Secreted	0.51	0.96	0.73	0.91	0.16	0.08	0.11	0.74	0.55	142

The two tables above summarize the results from the Random Forest model. Table 5 shows that the model can predict proteins to their subcellular location with a 51% accuracy. Table 6 illustrates statistics by class and indicates the performance of the model using the testing data. The locations with a sensitivity value of 0 and a specificity value of 1 indicate that there are no proteins predicted to these subcellular locations.

The secreted location has the highest MCC value of 0.55, with a sensitivity of 51% and a specificity of 96%. This location has a balanced accuracy of 74%. Next, the nuclear subcellular location has an MCC value of 0.45. The sensitivity, specificity, and balanced accuracy values for the nucleus are 89%, 58%, and 73% . Finally, the cell membrane location has an MCC value of 0.44. The sensitivity and specificity values for the nucleus are 49% and 95%. The balanced accuracy value for this location is 72%.

4.5 ADA model

Table 7: Overall Statistics

Accuracy	0.49
95% CI	$(0.45, 0.52)$
No Information Rate	0.34
P-Value [Acc > NIR]	2.2×10^{-6}
Kappa	0.31

Table 8: Statistics by Class

	Sen	Spe	PPV	NPV	Pre	DR	DP	BA	MCC	NoP
Cell membrane	0.53	0.94	0.47	0.95	0.09	0.05	0.1	0.73	0.44	76
Cytoplasm	0.25	0.9	0.38	0.83	0.19	0.05	0.13	0.58	0.18	171
Endoplasmic Reticulum	0.21	0.98	0.43	0.94	0.07	0.01	0.03	0.59	0.27	62
Golgi Apparatus	0	1	NA	0.97	0.03	0	0	0.5	0	23
Lysosome	0	1	NA	0.99	0.01	0	0	0.5	0	9
Mitochondria	0.3	0.94	0.39	0.92	0.1	0.03	0.08	0.62	0.28	92
Nucleus	0.84	0.54	0.48	0.87	0.34	0.28	0.59	0.69	0.36	299
Peroxisome	0	1	NA	0.99	0.01	0	0	0.5	0	10
Secreted	0.38	0.99	0.86	0.89	0.16	0.06	0.07	0.68	0.53	142

The above tables summarize the results from the Adaptive Boosting (ADA) model. Table 7 indicates that the ADA model predicts the subcellular location of proteins with an accuracy of 49%. Therefore the overall accuracy of the ADA model in predicting protein subcellular location is slightly lower than Random Forest model.

Table 8 presents the statistics by class and indicates the performance of the model, using testing data. The secreted location has the highest MCC value of 0.53, with a sensitivity of 38%, and a specificity of 99%. This location has a balanced accuracy of 68%. Next, the cell membrane subcellular location has an MCC value of 0.44. The sensitivity, specificity, and balanced accuracy values for the nucleus are 53%, 94%, and 73%.

4.6 ADA model with Cross Validation

Table 9: Overall Statistics

Accuracy	0.42
95% CI	$(0.40, 0.44)$
No Information Rate	0.34
P-Value [Acc > NIR]	2.2×10^{-6}
Kappa	0.19

Table 10: Statistics by Class

	Sen	Spe	PPV	NPV	Pre	DR	DP	BA	MCC	NoP
Cell membrane	0.3	0.95	0.38	0.93	0.1	0.03	0.08	0.62	0.27	286
Cytoplasm	0.07	0.96	0.28	0.82	0.18	0.01	0.04	0.51	0.05	540
Endoplasmic Reticulum	0.29	0.98	0.48	0.95	0.07	0.02	0.04	0.63	0.34	207
Golgi Apparatus	0	1	NA	0.97	0.03	0	0	0.5	0	87
Lysosome	0	1	NA	0.99	0.01	0	0	0.5	0	30
Mitochondria	0.13	0.98	0.45	0.9	0.11	0.01	0.03	0.56	0.2	329
Nucleus	0.9	0.37	0.42	0.87	0.34	0.31	0.72	0.63	0.28	1002
Peroxisome	0	1	NA	0.99	0.01	0	0	0.5	0	30
Secreted	0.26	0.95	0.46	0.88	0.15	0.04	0.08	0.6	0.26	438

The above tables summarize the results from the Adaptive Boosting (ADA) with cross validation model. Table 9 indicates that the ADA model predicts the subcellular location of proteins with an accuracy of 42%. Therefore the overall accuracy of the ADA with cross validation model in predicting protein subcellular location is lower than Random Forest model.

Table 10 presents the statistics by class and indicates the performance of the model, using testing data. The ER location has the highest MCC value of 0.34, with a sensitivity of 29%, and a specificity of 98%. This location has a balanced accuracy of 63%. Next, the nuclear subcellular location has an MCC value of 0.28. The sensitivity, specificity, and balanced accuracy values for the nucleus are 90%, 37%, and 63%. Finally, the cell membrane location has an MCC value of 0.27. The sensitivity and specificity values for the nucleus are 30% and 95%. The balanced accuracy value for this location is 62%.

The cell membrane and secreted subcellular locations failed to perform as well as they had performed in the Random Forest and ADA models. The predictions for the nuclear location are also lower than the predictions for this location using the Bagging method.

4.7 SAMME Model

Table 11: Overall Statistics

Accuracy	0.51
95% CI	$(0.48, 0.59)$
No Information Rate	0.34
P-Value [Acc > NIR]	2.2×10^{-6}
Kappa	0.37

Table 12: Statistics by Class

	Sen	Spe	PPV	NPV	Pre	DR	DP	BA	MCC	NoP
Cell membrane	0.53	0.95	0.48	0.96	0.09	0.05	0.09	0.74	0.45	76
Cytoplasm	0.41	0.84	0.39	0.86	0.19	0.08	0.2	0.63	0.25	171
Endoplasmic Reticulum	0.21	0.98	0.48	0.94	0.07	0.01	0.03	0.6	0.29	62
Golgi Apparatus	0	1	0	0.97	0.03	0	0	0.5	-0.01	23
Lysosome	0	1	NA	0.99	0.01	0	0	0.5	0	9
Mitochondria	0.36	0.93	0.38	0.93	0.1	0.04	0.1	0.65	0.3	92
Nucleus	0.77	0.71	0.57	0.86	0.34	0.26	0.45	0.74	0.45	299
Peroxisome	0	1	NA	0.99	0.01	0	0	0.5	0	10
Secreted	0.49	0.95	0.67	0.91	0.16	0.08	0.12	0.72	0.51	142

The two tables above summarize the results from the Stagewise Additive Model using a Multi-class Exponential loss function (SAMME) model. Table 11 indicates that the SAMME model predicts the subcellular location of proteins with an accuracy of 48%.

Table 12 indicates that the subcellular locations of the secreted, the cell membrane, and the nucleus have the highest MCC values. These results follow the same pattern of prediction from the previous models, where these three subcellular locations have had the highest values for MCC. The sensitivity values of these locations, in order, are 49%, 53%, 77%. The specificity value of these locations, in order, are 95%, 95%, 71%. The MCC values for these locations are 0.51, 0.45, and 0.45 respectively.

4.8 SVMs Model

Table 13: Overall Statistics

Accuracy	0.55
95% CI	$(0.53, 0.57)$
No Information Rate	0.34
P-Value [Acc > NIR]	2.2×10^{-16}
Kappa	0.42

Table 14: Statistics by Class

	Sen	Spe	PPV	NPV	Pre	DR	DP	BA	MCC	NoP
Cell membrane	0.59	0.95	0.57	0.96	0.1	0.06	0.1	0.77	0.54	286
Cytoplasm	0.34	0.87	0.38	0.85	0.18	0.06	0.17	0.61	0.22	540
Endoplasmic Reticulum	0.35	0.98	0.54	0.95	0.07	0.02	0.05	0.66	0.4	207
Golgi Apparatus	0.03	1	0.5	0.97	0.03	0	0	0.52	0.13	86
Lysosome	0	1	NA	0.99	0.01	0	0	0.5	0	30
Mitochondria	0.48	0.94	0.52	0.94	0.11	0.05	0.1	0.71	0.44	328
Nucleus	0.76	0.76	0.62	0.86	0.34	0.26	0.42	0.76	0.5	1001
Peroxisome	0	1	NA	0.99	0.01	0	0	0.5	0	30
Secreted	0.63	0.92	0.57	0.93	0.15	0.09	0.16	0.77	0.52	436

Table 13 shows that the model can predict the subcellular location of proteins with a 55% accuracy. The subcellular location with the best prediction results in the model was the cell membrane, with a sensitivity of 59% and a specificity of 95%. Other locations with good model performance, in the context of these two measurements, include the secreted, with a sensitivity of 63% and a specificity of 92% and the nucleus, with a sensitivity of 76% and a specificity of 76%.

The three subcellular locations of the cell membrane, the secreted, and the nucleus also have the three highest values of balanced accuracy and MCC. The cell membrane has a balanced accuracy of 77% and an MCC of 0.54 , the secreted has a balanced accuracy of 77% and an MCC of 0.52, and the nucleus which has a balanced accuracy of 76% and an MCC of 0.50.

4.9 ANN Model

Table 15: Overall Statistics

Accuracy	0.49
95% CI	$(0.46, 0.52)$
No Information Rate	0.34
P-Value [Acc > NIR]	2.2×10^{-16}
Kappa	0.33

Table 16: Statistics by Class

	Sen	Spe	PPV	NPV	Pre	DR	DP	BA	MCC	NoP
Cell membrane	0.39	0.95	0.43	0.94	0.09	0.03	0.08	0.67	0.36	76
Cytoplasm	0.12	0.95	0.35	0.82	0.19	0.02	0.07	0.53	0.11	171
Endoplasmic Reticulum	0.21	0.98	0.43	0.94	0.07	0.01	0.03	0.59	0.27	62
Golgi Apparatus	0	1	NA	0.97	0.03	0	0	0.5	0	23
Lysosome	0	1	NA	0.99	0.01	0	0	0.5	0	9
Mitochondria	0.5	0.89	0.35	0.94	0.1	0.05	0.15	0.7	0.34	92
Nucleus	0.85	0.65	0.55	0.89	0.34	0.29	0.52	0.75	0.47	299
Peroxisome	0	1	NA	0.99	0.01	0	0	0.5	0	10
Secreted	0.49	0.91	0.52	0.9	0.16	0.08	0.15	0.7	0.42	142

Table 15 shows that the model can predict the subcellular location of proteins with a 49% accuracy. The subcellular location with the best prediction in the model was the nucleus, with a sensitivity of 85% and a specificity of 65%. The next best result was seen with the secreted location also has a sensitivity of 49% and a specificity of 91%.

The subcellular locations of the nucleus and the secreted also have the two highest values of balanced accuracy and MCC. The nucleus has a balanced accuracy of 75% and an MCC of 0.47 while the secreted has a balanced accuracy of 70% and an MCC of 0.42.

5. Analysis

The main objective of this study was to use several different classifiers to predict the subcellular location of animal and human proteins and to determine which of these classifiers performed the best in predicting protein subcellular location. The working benchmark dataset includes 2944 protein sequences. The subcellular locations of these proteins are the nucleus

(1001 proteins), the cytoplasm (540 proteins), the secreted (436 proteins), the mitochondria (328 proteins), the cell membrane (286 proteins), the endoplasmic reticulum (207 proteins), the Golgi apparatus(86 proteins), the peroxisome (30 proteins), and the lysosome (30 proteins). Therefore, there are 9 different subcellular locations for proteins in this dataset.

The method used for representing proteins in the study is the pseudo-amino acid composition (PseAA composition), adapted from earlier studies. The computational predictors that are used to predict the subcellular location of proteins in animals and humans in this study include Bagging, Bagging with cross validation, Random Forest, the Adaptive Boosting (AdaBoost), AdaBoost with cross validation, the Stage-wise Additive Modeling using a Multi-class Exponential loss function (SAMME), Support Vector Machines (SVMs), and Artificial Neural Networks (ANNs).

The results from this study demonstrate that the SVM model has the best overall accuracy of 55% for predicting the subcellular location of proteins. However, since the dataset for this study is imbalanced (the number of proteins in each subcellular location varies greatly), the overall accuracy is not a good measure of performance of the computational predictors. Therefore, for the analysis of the prediction results, the measures of sensitivity, specificity, and MCC are chosen to gauge how well a predictor can predict the correct subcellular locations of proteins.

After analyzing the results, the predictions for the three subcellular locations of the nucleus, cell membrane, and secreted demonstrate the best outcomes out of the nine subcellular locations in this study. The Table 17 displays the performance of the various predictors in predicting the subcellular location of proteins for these three locations.

The nucleus proteins are best predicted by the Bagging method, with a sensitivity of 0.93, a specificity of 0.93, and an MCC of 0.93. The next best predictor for the nucleus proteins is the SVMs with, an MCC value of 0.50, which is the second highest MCC value. The sensitivity and specificity values are 0.76 and 0.76. respectively.

The cell membrane proteins are best predicted by the AdaBoost, with cross validation with a sensitivity of 0.30, a specificity of 0.95, and an MCC value of 0.62. The SVMs method has an outcome with a sensitivity of 0.59, a specificity of 0.95, and an MCC of 0.54. The SAMME predictor can predict cell membrane protein with a sensitivity of 0.53, a specificity of 0.95, and an MCC of 0.45. For cell membrane protein prediction, the predictors fail to produce higher sensitivity results compared to the results observed for the nucleus location. Therefore, the future work will include efforts to improve the sensitivity metric for the cell membrane protein location prediction. Finally, the MCC value for each of the mentioned predictors is low, as a result of the lower sensitivity values in each predictor.

The secreted proteins are best predicted by Random Forest, with a sensitivity of 0.51, a specificity of 0.96, and an MCC of 0.55. Next, the AdaBoost predictor shows a sensitivity of 0.38, a specificity of 0.99, and MCC of 0.53. The SVMs method has a sensitivity of 0.63, a specificity of 0.92, and an MCC of 0.52. Similar to the prediction of the cell membrane subcellular location, the sensitivity results from each predictor need to be further improved. The specificity results are optimal and high. The lower MCC level in each case mentioned above is as a result of lower sensitivity value from each predictor.

Therefore, in general, it can be concluded that of the methods used in this study to predict the location of the nucleus proteins, the Bagging method produces the best prediction results (with a sensitivity of 0.93, specificity of 0.93, and MCC of 0.93). For the prediction of cell membrane protein, the AdaBoost with cross validation produces the best results with an MCC of 0.62, a sensitivity of 0.30, and a specificity of 0.95. Finally, to predict the location of secreted proteins, Random Forest has the best results with an MCC of 0.55, a

Table 17: Protein Subcellular Location Prediction Results Using Various Predictors

Subcellular Locations	Predictors	Sensitivity	Specificity	Balanced Accuracy	MCC
Nucleus	Bagging	0.93	0.35	0.64	0.93
	Bagging with CV	0.89	0.38	0.64	0.28
	Random Forest	0.89	0.58	0.73	0.45
	AdaBoost	0.84	0.54	0.69	0.36
	AdaBoost with CV	0.90	0.37	0.63	0.28
	SAMME	0.77	0.71	0.74	0.45
	SVMs	0.76	0.76	0.76	0.50
	ANNs	0.85	0.65	0.75	0.47
Cell Membrane	Bagging	0.39	0.94	0.66	0.39
	Bagging with CV	0.30	0.95	0.62	0.27
	Random Forest	0.49	0.95	0.72	0.44
	AdaBoost	0.53	0.94	0.73	0.44
	AdaBoost with CV	0.30	0.95	0.27	0.62
	SAMME	0.53	0.95	0.74	0.45
	SVMs	0.59	0.95	0.77	0.54
	ANNs	0.39	0.95	0.67	0.36
Secreted	Bagging	0.16	0.99	0.58	0.16
	Bagging with CV	0.25	0.95	0.60	0.26
	Random Forest	0.51	0.96	0.74	0.55
	AdaBoost	0.38	0.99	0.68	0.53
	AdaBoost with CV	0.26	0.95	0.60	0.26
	SAMME	0.49	0.95	0.72	0.51
	SVMs	0.63	0.92	0.77	0.52
	ANNs	0.49	0.91	0.70	0.42

sensitivity of 0.51, and a specificity of 0.96.

6. Future Work

The future direction for this project includes exploring other available predictors in order to determine whether the overall accuracy of protein subcellular location prediction can be increased. Also, other methods can be explored to attempt to obtain a higher predictive value for other subcellular locations than the three locations discussed in this paper. Examples of these methods include the covariant discriminant algorithm and random walk on graphs.

Another issue that arose from this research was how to analyze imbalanced data. In this paper, the measures of sensitivity, specificity, and MCC were used to analyze the results. However, additional attention could be brought to finding better ways to prepare the data set (over-sampling or under-sampling), before using the computational predictors. The R software offers a package for “Synthetic Sampling", which can be used on this dataset for future work. In addition, using “penalized SVM", in which additional costs are imposed on the model for making mistakes in classification, can also be applied to this work to improve the overall analysis of the data.

References

[1] ABDOLLAH DEHZANGI, RHYS HEFFERNAN, A. S. J. L. K. P. Gram-positive and gram-negative protein subcellular localization by incorporating evolutionary-based descriptors into chou’s general pseaaceneral pseaac. *Journal of Theoretical Biology 364* (2015), 284–294.

[2] ANIL K. JAIN, JIANCHANG MAO, K. M. Artificial neural networks: A tutorial. *IEEE Computer Society 29*, 3 (1996), 31–44.

[3] ASHUTOSH KUMAR SINGH, S. S. S., AND MISHRA, A. Sub-cellular localization prediction using machine learning approach. *Nucleus 734*, 568 (2015), 63.

[4] BREIMAN, L. Bagging predictors. *Machine Learning 24* (1996a), 123–140.

[5] BREIMAN, L. Out-of-bag estimation. ftp://ftp.stat.berkeley.edu/pub/users/breiman/OOBestimation.ps.Z., 1996b.

[6] BREIMAN, L. Random forests. *Machine Learning 45* (2001), 5–32.

[7] CHIH-WEI HSU, C.-C. C., AND LIN, C.-J. A practical guide to support vector classification. 4 2010.

[8] CHIH-WEI HSU, C.-J. L. A comparison of methods for multi-class support vector machines. *IEEE Transactions on Neural Networks 13*, 2 (2002), 415–425.

[9] CONSORTIUM, T. U. Uniprot knowledgebase (uniprotkb), 11 2015.

[10] D. CHARIF, J. L. Sequinr: A contributed pacage to the r project for statistical computing devoted to biological sequence retrieval and analysis.

[11] FOR DEVELOPMENTAL BIOLOGY, M.-P. I.

[12] JI ZHU, SAHARON ROSSET, H. Z. T. H. Multi-class adaboost. http://www.researchgate.net/profile/Trevor_Hastie/publication/228947999_Multi-class_adaboost/links/0c960521b946de42a9000000.pdf., 2006.

[13] JOHN MEINKEN, GARY WALKER, C. R. C., AND MIN, X. J. Metazseckb: the human and animal secretome and subcellular proteome knowledgebase. *Database: The Journal of Biological Databases and Curation Article ID bav077* (2015).

[14] JOSEFINE SPRENGER, J LYNN FINK, R. D. T. Evaluation and comparison of mammalian subcellular localization prediction methods. *BMC Bioinfomatics 7*, (Suppl 5):S3 (2006).

[15] KEUN-JOON PARK, M. K. Prediction of protein subcellular locations by support vector machines using compositions of amino acids and amino acid pairs. *Bioinformatics 19* (2003), 1656–1663.

[16] KOU-CHEN CHOU, H.-B. S. Recent progress in protein subcellular location prediction. *Analytical Biochemistry 370* (2007), 1–6.

[17] KOU-CHEN CHOU, H.-B. S. A new method for predicting the subcellular localization of eukaryotic proteins with both single and multiple sites: Euk-mploc 2.0. *PLOSONE 5*, 4 (2010).

[18] MATTHEWS, B. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochim. Biophys. Acta 405* (1975), 442–451.

[19] MIN, X. J. Evaluation of computational methods for secreted protein prediction in different eukaryotes. *Journal of Proteomics and Bioinformatics 3*, 5 (2010).

[20] MUNYON, J. Predicting fungal protein subcellular location. May 2015.

[21] N. XIAO, Q.S. XU, D. C. protr: Generating various numerical representation scheme of protien sequence.

[22] NEIZER-ASHUN, K. A. Prediction of plant protein subcellular location. 7th International Conference on Bioinformatics and Computational Biology (BICoB-2015).

[23] REINHARDT, A., AND HUBBARD, T. Using neural networks for prediction of the subcellular location of proteins. *Nucleic Acids Research 26*, 9 (1998), 2230–2236.

[24] SARAVANAN K, S. S. Review on classification based on artificial neural networks. *International Journal of Ambient Systems and Applications (IJASA) 2*, 4 (2014), 11–18.

[25] SYLVIA S. MADER, M. W. *ESSENTIALS OF BIOLOGY*, fourth edition ed. MC GRAW HILL EDUCATION, 2015.

[26] TEAM, R. C. R: A language and environment for statistical computing, 11 2015.

[27] YOAV FREUND, R. E. S. Experiments with a new boosting algorithm. *Machine Learning: Proceedings of the Thirteenth International Conference* (1996).

[28] YOAV FREUND, R. E. S. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences 55* (1997), 119–139.

[29] YOAV FREUND, R. E. S. A short introduction to boosting a short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence, 14*, 5 (1999), 771–780.

[30] YU-DONG CAI, XIAO-JUN LIU, X.-B. X., AND CHOU, K.-C. Support vector machines for prediction of protein subcellular location by incorporating quasi-sequence-order effect. *Journal of Cellular Biochemistry 84*, 343-348 (2002).

[31] YUAN, Z. Prediction of protein subcellular location using markov chain model. *FEBS letters 451*, 23-26 (1999).