Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: 23 March 2018
Manuscript Category

OXFORD

COMPGI10 - Bioinformatics

Predicting the subcellular location of eukaryotic proteins

Vasileios Papastefanopoulos 17046496 [1,*]

[1] Department of Computer Science, University College London, London, WC1E 6BT, United Kingdom

* To whom correspondence should be addressed.

Received on 23/03/2018; revised on 23/032018; accepted on 23/03/2018

Abstract

Motivation: Knowledge of the subcellular localization of a protein is essential for understanding its structure and function as well as for determining the origin of various diseases and identifying the possible targets for intervention in the medicine discovery process. However, predicting the location of proteins for which there exist no annotated homologues can be challenging. Therefore, tools and methods that can determine the location of a protein, solely relying on its amino acid sequence are of great significance.
Results: In this work, we propose a method that uses XGBoost to predict the subcellular localization of a protein, relying on a set of features, engineered using the amino acid chain of the protein. The model was developed and evaluated on a dataset of 9222 protein sequences. Our model achieves an accuracy of 71%, an F1 score of 71% and an unweighted mean of AUC of 90% over four different classes.
Availability: The dataset, code, predictions for the blind set of proteins as well as high resolution images of the figures presented in this report are all available for download and can be found and at the following Box repository: https://goo.gl/WFBtEv. All the source code is contained in a single fully-commented Jupyter Notebook.
Contact: ucabvp1@ucl.ac.uk

1 Introduction

Prediction of the subcellular localization of eukaryotic proteins is a Bioinformatics problem that focuses on forecasting where a protein resides in a cell. The compartment or organelle where a protein is located, determines the physiological context concerning its function and as a result valuable information can be extracted for the latter by studying the former [12]. More specifically, knowledge regarding the subcellular locations of proteins is critical for comprehensive cell biology research and of high importance for drug development, system biology and proteomics [5]. Furthermore, irregularities regarding subcellular localization of proteins responsible for the structural, metabolic or signaling cell properties may lead to disorders associated with biogenesis, protein aggregation, cell metabolism or signaling such as metabolic, cardiovascular and neurodegenerative diseases, as well as cancer [12]. For some proteins, subcellular localization can be predicted via homology searching. Sequence similarity search can recognize homologous proteins by spotting statistically significant similarity that reflects common ancestry. It is typically performed with BLAST, which is widely used and considered to be the most reliable tool for characterizing newly determined sequences [14]. However, in many cases, no annotated homologous proteins exist and as a result predictive tools that determine the protein location using its sequence are highly desirable. Such tools typically accept as input information regarding a protein, such as its amino acid sequence and produce a predicted location within the cell as output, such as the nucleus, Endoplasmic reticulum, Golgi apparatus, extracellular space, or other organelles.

2 Related work

Much research has been conducted in this area and a plethora of machine learning techniques have been applied in this task with great success. Prior work includes classical machine learning methods that are based on feature engineering and prior knowledge, as well as, recently developed, deep or more accurately representation learning methods that process the sequences in a more natural way and exploit their spatial structure. The body of literature is mostly composed of classical machine learning methods, with Support Vector Machines (SVMs) being the driving force behind most approaches. In [11], SVMs were employed to create SubLoc, a system able to predict the subcellular location of proteins from their amino acid chain. Trained using five-fold cross validation on the Reinhardt and

Hubbard dataset [15] (also known as the RH2427 dataset), which allowed for no more than 90% pair-wise sequence homology, SubLoc achieved a total prediction accuracy of 91.4% on the test set over three subcellular locations for prokaryotic cells and 79.4% over four locations for eukaryotic ones.

A hybrid model (ESLpred) that combined SVMs and PSI-BLAST for sequence querying and utilized protein features such as dipeptide compositions and various properties was developed in [2]. The model was trained, tuned and evaluated using a five-fold cross validation on the Reinhardt and Hubbard dataset [15] and accuracies of 95.3%, 85.2%, 68.2% and 88.9% were respectively reported over the four classes: nuclear, cytoplasmic, mitochondrial and extracellular.

Traditional amino acid and dipeptide composition were also utilized by SVMs in [8], where overall accuracy scores of 76.6 and 77.8% over four classes were reported. In the same study, a SVM model (hybrid3) relying on amino acid composition, dipeptide compositions of standard and higher order and PSI-BLAST outputs was elaborated and reached an accuracy of 84.4%. Results were reported using five-fold cross-validation on a dataset where no two sequences 90% or more sequence identity.

Another combinatorial approach (LOCSVMPSI) was developed in [17], where a combination of position-specific scoring matrices (produced from PSI-BLAST profiles) and SVMs achieved an accuracy of 90.2% using five-fold cross validation, which is greater than the corresponding accuracies reported by SubLoc [3] and ESLpred [2] on the same dataset (RH2427).

In [10], two new methods for protein subcellular localization were introduced, namely TargetLoc and MultiLoc. The former is a multi-layer SVM based method that was developed and tested using five-fold cross validation on the TargetP dataset [7], which allowed for up to 80% pair-wise homology. MultiLoc has three different versions for animals, fungal and plants. Its architecture is similar to TargetLoc but additionally incorporates a method for identifying signal anchors, while also including several additional features in order to enable predictions for the increased number of localizations. Training and evaluation of Multiloc were performed using five-fold cross validation on a dataset that contained no protein sequences with pair-wise similarity of 80% or greater. In terms of performance, TargetLoc achieved an overall accuracy of 89.7% over four plant categories and 92.5% over three non-plant categories, whereas the overall accuracy of the three MultiLoc versions: plant, fungal and animal was approximately 75%.

In a more recent study [13], the relationship between protein subcellular localization and residue exposure was explored and a two-stage method that uses multiple SVMs and Neural Networks for the two stages respectively was introduced. A balanced training set was generated by selecting proteins annotated to reside in the four following locations: nucleocytoplasmic, extracellular, cytoplasmic and nuclear. For both hyperparameter tuning and accuracy evaluation, ten-fold cross validation was used. The method reached an accuracy of 62% without taking into account 3D-information and 68% when 3D-produced values of amino acid exposure were passed as input . Interestingly, the proposed method was able to correctly assign proteins to different locations in spite of them sharing high levels of identity.

A very recent approach (MKLoc) [9] tried to address the single kernel inefficiencies of regular SVMs by introducing a multiple kernel SVM system for multi-label protein subcellular localization prediction, which outperformed regular single kernel SVM methods. MKLoc was evaluated using a dataset consisting of 5447 proteins that have a single location (part of the Hoglund dataset [10] and 3056 proteins occuring in multiple locations (part of the DBMLoc dataset [18]). Five-fold cross validation was implemented for developing and evaluating the classifier and an overall accuracy of 69.4% over 9 possible locations was reported.

Although classical machine learning methods achieve a good overall performance in this task they do not have a way of exploiting the spatial structure that is exhibited in a protein sequence, something that Deep Learning techniques fare quite impressively in.

Before the advent of Deep Learning, there were some attempts to predict the subcellular location of proteins using Neural Networks, most notably [15] and [7].

The most recent methods involve Deep Learning models, which are based on learning hierarchical data representations, as opposed to feature engineering. In [16], a deep learning based predictor (DeepPSL) that uses Stacked Auto-Encoder networks (SAE) is proposed. Experimental results show that DeepPSL outperforms traditional machine learning based methods. The model was train and tested using three-fold cross validation on a dataset that was generated using Human protein sequences, collected from a UniProtKB. Homolog bias was avoided as any two of proteins in the dataset had less than 70% sequence similarity, which is less compared to the Hoglund dataset which allowed for up to 80% homology identity. The overall accuracy was 37.4% over 10 classes but with high deviation between different class scores, possibly due to the imbalanced dataset.

In a very recent approach [1], four different Deep Learning architectures were examined: a Feedforward Multilayer Perceptron, a Bidirectional LSTM, a Bidirectional LSTM with attention mechanisms and a Convolutional Bidirectional LSTM with attention mechanisms as well. Experiments identified Convolutional Bidirectional LSTM with attention as the highest performing deep architecture. All architectures were trained and evaluated with two separate datasets: the Hoglund dataset, which as already mentioned allowed up to 80% sequence similarity and the DeepLoc dataset, which was developed specifically for these experiments and for which a more strict policy was enforced regarding homology. The final model (DeepLoc) consisted of an ensemble of 16 models. Five folds generated in total, each of which contained proteins equally distributed from all classes. Four of them were used for developing and tuning the model and one for its final evaluation. A final accuracy of 78% over 10 categories was achieved, which was an improvement over the state-of-the-art methods, even the ones exploiting knowledge of homology to their benefit.

3 Approach

3.1 Datasets

Two different datasets were obtained:

- a labeled dataset in fasta format to be used for training and evaluation.
- an unlabeled dataset in fasta format to be used for independent evaluation of the method.

Regarding the first dataset, 9222 amino acid sequences were distributed among four different classes according to their subcellular location: Cytosolic, Secreted, Mitochondrial and Nuclear Proteins. Every sequence was assumed to be non-homologous The dataset suffers from a slight imbalance between the four categories with Nuclear proteins being the most over-represented category (approximately 36% of the dataset) and Mitochondrial being the most under-represented (14%). A more comprehensive analysis of the distribution of the dataset along with the length statistics for each class is reported in table 1.

The second dataset contained 20 amino acid sequences for which the corresponding labels were completely unknown. Predictions for the proteins of this dataset will be used for independent external evaluation.

3.2 Preprocessing

It was noticed that some sequences contained few non-standard amino acids, namely B, U and X. Since the number of such sequences was

insignificant (less than 1%), not much research was conducted regarding how to replace them and as a result U and X were simply eliminated from those sequences, while N was substituted for B. Another issue which was encountered, concerned the distribution of the data over the four categories. As mentioned before, there was a slight imbalance between the four classes. However, the extent of the problem was not deemed significant enough to resort to oversampling or other imbalance-countering strategies.

	Avg. Length	**Std**	**Min**	**Max**	**No. Seq**	**Percent**
Cyto	665	549	19	7393	3004	33%
Mito	376	248	19	2628	1299	14%
Nuclear	625	501	35	5596	3314	36%
Secreted	305	518	11	13100	1605	17%
Overall	547	514	11	13100	9222	100%

Table 1. Dataset Statistics: Data distribution and sequence length information

3.3 Data splitting

After the sequence preprocessing step, the labeled dataset was divided in two parts using stratified sampling, meaning that in each part the class distribution of the original dataset is preserved in all subsets. The first part containing 90% of the initial dataset was used for training and hyperparameter tuning, while the second part, which contained the rest of the sequences (10%) was used as a test set to evaluate the final performance of the model. In order to split the data, sklearn's method .train_test_split was used with the parameter Random_state being set to 13, so that any results can be easily reproduced.

3.4 Feature engineering

Traditional machine learning techniques require a list of features to be fed to a classifier in order to be able to learn from existing data and make predictions for new unseen data. In this case, a number of features were derived from the raw amino acid sequences, that describe and try to capture the properties exhibited by the different proteins in terms of their classes (locations). In total, 497 features were created, many of which were computed using the BioPython module [6]. The created features are listed below:

1. Protein sequence length
2. Percentages of all possible dipeptides present in whole sequence
3. Percentages of all 20 amino acids present in whole sequence, first and last 50 elements of the sequence
4. Percentages of hydrophobic amino acids present in whole sequence, first and last 50 elements of the sequence
5. Percentages of negatively and positively charged amino acids present in whole sequence, first and last 50 elements of the sequence
6. Percentages of acidic and basic amino acids present in whole sequence, first and last 50 elements of the sequence
7. Percentages of polar amino acids present in whole sequence, first and last 50 elements of the sequence
8. Secondary structure fraction of amino acids (helix, turn and sheet)
9. Information regarding the Isoelectric point of the sequence
10. Molecular weight
11. Molar extinction coefficients
12. Protein GRAVY (GRand AVerage of hydropathY) value
13. Aromaticity
14. Instability Index

3.5 Training Procedure

3.5.1 The classifier

As discussed in a previous section, SVMs were the most popular classifier among researchers for this task before the advent of Deep Learning. While hierarchical (deep) learning approach would probably yield better results, in this work an alternative approach is followed as a classifier based on tree boosting is developed. Tree methods are well known for their simplicity and interpretability, while also maintaining a strong predictive power. Furthermore, they can be easily extended to include more features and can be trained and tuned at a much lower computational cost compared to Deep Learning architectures. More specifically, an XGBoost classifier was trained and used for predictions on the unlabeled test. XGBoost is scalable end-to-end tree boosting system that has been extensively used by data scientists and has reported state-of-the-art results on numerous machine learning problems [4]. In general, similarly to other boosting methods, XGBoost combines weak models (decision trees in this case) into a single strong model (ensemble) using an iterative process.

3.5.2 Hyperparameter tuning

XGBoost includes an extended list of trainable hyperparameters, which makes grid search over them a quite computationally heavy task. Due to limited computational resources, grid-search optimizing was reduced to just three important hyperparameters of the classifier: n_estimators (number of voting trees), max_depth (the maximum depth of the created trees) and gamma, which is a regularizing parameter (the higher its value, the higher the regularization). For each combination of these 3 hyperparameters (125 models in total) a ten-fold cross validation was performed to evaluate their average performance over the 10 folds. Each one with these folds was generated using Stratified sampling meaning that the class distribution was preserved in each of the 10 folds. The best performing model had n_estimators = 1000, max _depth = 10 and gamma = 0, reporting an mean accuracy of 69.47% over the 10 folds. The accuracy scores over the ten folds reported in greater detail in Table 2.

1st fold	**2nd fold**	**3rd fold**	**4th fold**	**5th fold**
0.6887	0.6923	0.6931	0.706	0.7072
6th fold	**7th fold**	**8th fold**	**9th fold**	**10th fold**
0.6791	0.7032	0.6863	0.6996	0.6908
Mean	**Std. Dev.**			
0.6946	0.0086			

Table 2. Accuracy scores over the ten folds

3.5.3 Model analysis

Tree-based classifiers rely heavily on feature quality in order to create efficient decision trees. Inherently, they use some measurement of feature quality, such as information gain to identify the features that are most helpful in determining the category of the training examples. After training the model on all the training data using the optimal hyperparameters, the relative importance of each feature was calculated in an attempt to gain a deeper understanding of the dataset and to observe which are the most and less common characteristics among proteins that share the same living environment. In figures 1 and 2 the 30 most significant and 30 least important features along with their relative scores are presented. These plots indicate that the percentages of Cysteine and basic amino acids over the whole sequence, the dipeptide "Lysine - Arginine", the fractions of a protein's secondary structures of helix, turn and sheet, its isoelectric point, the percetange of acidic amino acids present in the first fifty elements, the instability index and a protein's tolerance to water all are good indicators of its location. On the other hand, the percentages of positively and negatively

charged amino acids either over the whole sequence or within the first and last fifty elements of the sequence do not provide the classifier with any useful information as to where a protein resides.

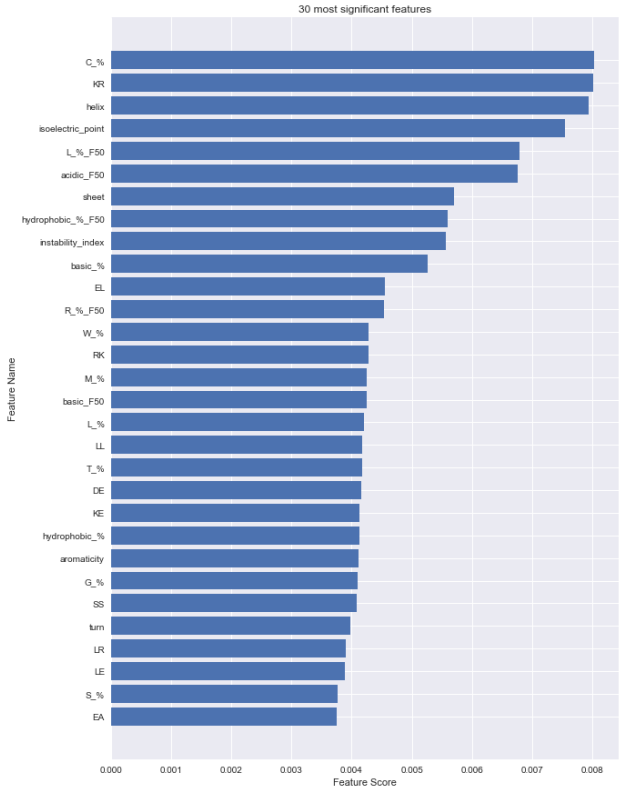

Figure 1: 30 most significant features

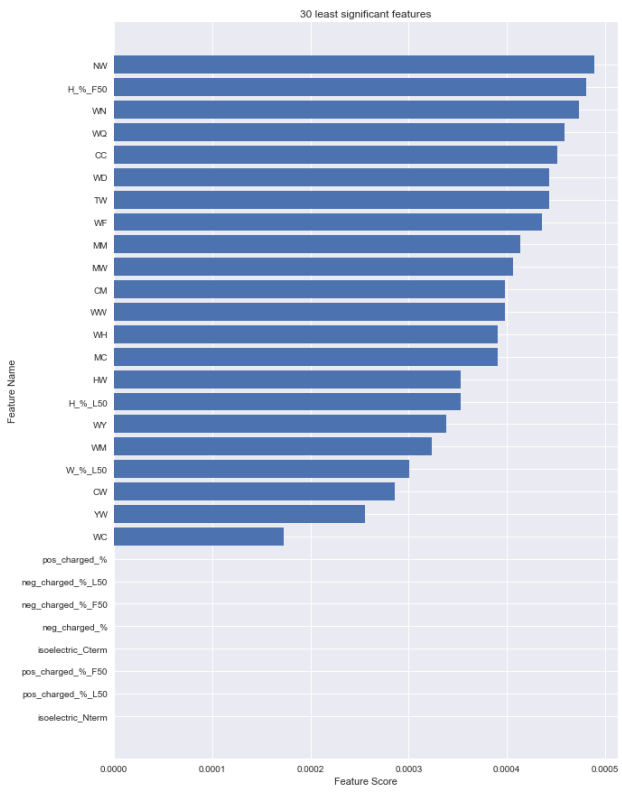

Figure 2: 30 least significant features

4 Results

As already stated, the Random_state parameter was set to 13 when splitting between training and test set and the results can be fully reproduced.

4.1 Holdout test set

To get a measure of the models performance, the holdout test set was used, consisting of 923 proteins with the same distribution over the four classes as the training set. The method achieved 71% accuracy, 71% weighted F1 score and an unweighted mean of AUC of 90% over the four classes on the previously unseen dataset. Specific F1 and AUC scores for each class separately are presented in table 3.

	Cytosolic	**Mitochondrial**	**Nuclear**	**Secreted**
F1 score	0.6531	0.7148	0.7053	0.8410
AUC score	0.8161	0.9387	0.8705	0.9683

Table 3. Class-wise F1 and AUC scores

The confusion matrix, calculated on the test examples and its normalized form are illustrated in figures 3 and 4 respectively. It is evident that the most confused protein classes are Nuclear and Cytosolic. To further elaborate, approximately 27% of Nuclear proteins were wrongly predicted as Cytosolic and about 22% of Cytosolic were wrongly predicted as Nuclear. This indicates that proteins residing in these two locations share similar feature values and probably more features need to be engineered in order to distinguish between the two categories. Another observation is that in most cases, if a protein is misclassifed, then it is misclassified as Cytosolic than anything else despite Cytosolic having less share than Nuclear in terms of class distribution. Furthermore, AUC score was calculated for each class individuall (Table 3) and the respective ROC curves were plotted. In addition, ROC curves for the micro and macro averages of the classes were generated in the same graph, presented in Figure 5. AUC scores confirm that the model performs poorest on the Cytosolic class.

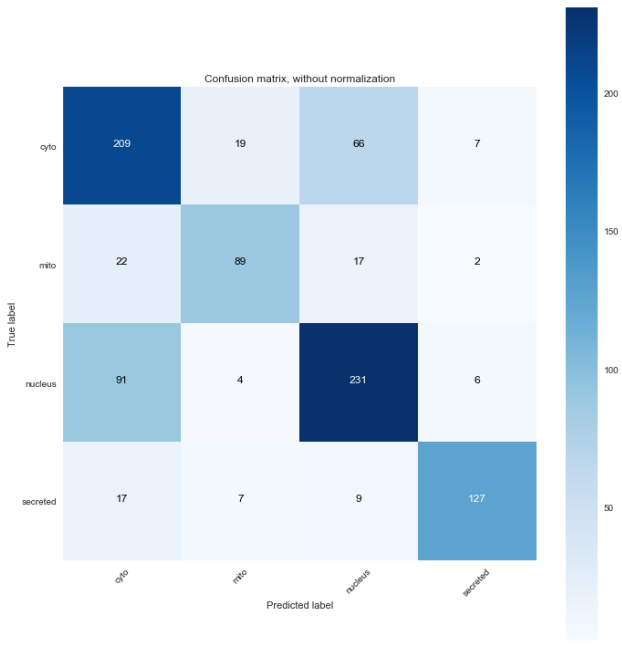

Figure 3: Confusion matrix

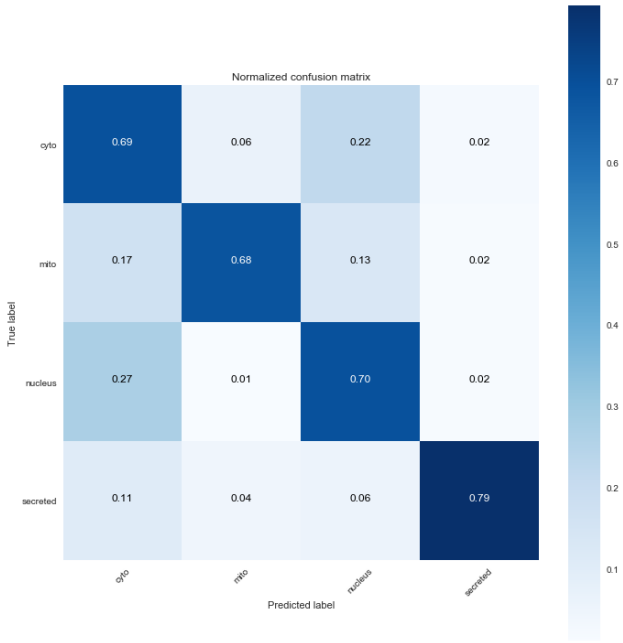

Figure 4: Normalized Confusion matrix

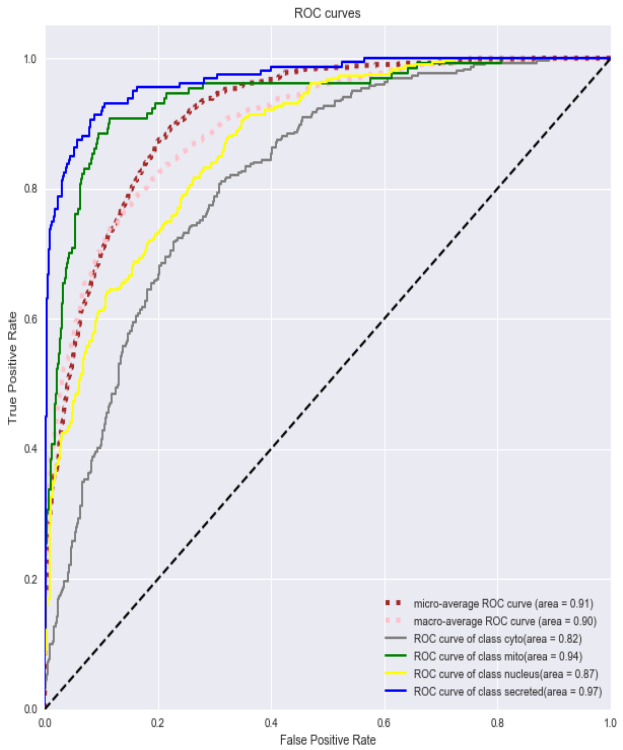

Figure 5: ROC curves

4.2 Unlabeled test set

After evaluating the model on the labeled test set, then the sequences contained there were incorporated to the training dataset in order to to build a final classifier, trained on all 9222 amino acid sequences. This final model was used to predict the location of 20 proteins for which the label was unknown. The predicted classes along with their corresponding confidence scores for each of the 20 examples are presented in Table 4. Confidence scores were calculated by evaluating all decision trees in the ensemble belonging to a specific class and then summing the resulting scores. Put simply, confidence scores represent how many of the voting trees in the ensemble agree on the same prediction: if all trees agree on the same prediction for a given datapoint then the confidence score for this datapoint would be 100%.

Sequence ID	**Predicted Class**	**Confidence**
SEQ677	Cytosolic	66.64%
SEQ231	Secreted	99.95%
SEQ871	Mitochondrial	71.82%
SEQ388	Nuclear	99.73%
SEQ122	Nuclear	99.34%
SEQ758	Nuclear	99.80%
SEQ333	Cytosolic	57.06%
SEQ937	Cytosolic	94.86%
SEQ351	Cytosolic	97.14%
SEQ202	Mitochondrial	99.81%
SEQ608	Mitochondrial	99.87%
SEQ402	Mitochondrial	99.77%
SEQ433	Secreted	92.90%
SEQ821	Secreted	99.97%
SEQ322	Nuclear	99.98%
SEQ982	Nuclear	99.93%
SEQ951	Cytosolic	99.39%
SEQ173	Cytosolic	97.58%
SEQ862	Mitochondrial	99.43%
SEQ224	Cytosolic	85.22%

Table 4. Blind set predictions and confidence scores

5 Conclusions and future work

In this work, we examined the problem of predicting the subcellular location of eukaryotic proteins. To counter this problem a model based on tree boosting was trained and developed, achieving an accuracy and F1 score of approximately 71% in an unseen test set. Although the model performed reasonably well, it was evidently indicated that there were some clear-cut issues that need to be addressed, the most noticeable one being the confusion between Cytosolic and Nuclear proteins. Regarding the aforementioned problem, a possible solution would be to include some features related to Nuclear localization signals. A nuclear localization signal or sequence (NLS) is a specific amino acid pattern contained within the amino acid sequence of proteins that are to be imported into the cell nucleus. In most cases, such a signal is composed of one or more short chain of positively charged Arginines or Lysines. We argue that including nuclear localization signals as features, would most likely have an positive impact on the predictive power of the model, as they would provide the classifier with more clear information regarding whether a protein belongs to the Nuclear class or not, resulting in less confusion between the Nuclear and Cytosolic classes. In the future, computational resources permitting, it would be interesting to experiment with Deep Learning architectures, such as CNNs or LSTMs or a combination of the two and explore their predictive capabilities on this task.

References

[1]J. J. Almagro Armenteros, C. K. Sønderby, S. K. Sønderby, H. Nielsen, and O. Winther. Deeploc: prediction of protein subcellular localization using deep learning. *Bioinformatics*, 33(21):3387–3395, 2017.

[2]M. Bhasin and G. Raghava. Eslpred: Svm-based method for subcellular localization of eukaryotic proteins using dipeptide composition and psi-blast. *Nucleic acids research*, 32(suppl_2):W414–W419, 2004.

[3]H. Chen, N. Huang, and Z. Sun. Subloc: a server/client suite for protein subcellular location based on soap. *Bioinformatics*, 22(3):376–377, 2006.

[4]T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. ACM, 2016.

[5]K.-C. Chou and H.-B. Shen. A new method for predicting the subcellular localization of eukaryotic proteins with both single and multiple sites: Euk-mploc 2.0. *PLoS One*, 5(4):e9931, 2010.

[6]P. J. Cock, T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski, et al. Biopython: freely available python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 25(11):1422–1423, 2009.

[7]O. Emanuelsson, H. Nielsen, S. Brunak, and G. Von Heijne. Predicting subcellular localization of proteins based on their n-terminal amino acid sequence. *Journal of molecular biology*, 300(4):1005–1016, 2000.

[8]A. Garg, M. Bhasin, and G. P. Raghava. Support vector machine-based method for subcellular localization of human proteins using amino acid compositions, their order, and similarity search. *Journal of biological Chemistry*, 280(15): 14427–14432, 2005.

[9]M. A. M. Hasan, S. Ahmad, and M. K. I. Molla. Protein subcellular localization prediction using multiple kernel learning based support vector machine. *Molecular BioSystems*, 13(4):785–795, 2017.

[10]A. Höglund, P. Dönnes, T. Blum, H.-W. Adolph, and O. Kohlbacher. Multiloc: prediction of protein subcellular localization using n-terminal targeting sequences, sequence motifs and amino acid composition. *Bioinformatics*, 22 (10):1158–1165, 2006.

[11]S. Hua and Z. Sun. Support vector machine approach for protein subcellular localization prediction. *Bioinformatics*, 17(8):721–728, 2001.

[12]M.-C. Hung and W. Link. Protein localization in disease and therapy. *J Cell Sci*, 124(20):3381–3392, 2011.

[13]A. S. Mer and M. A. Andrade-Navarro. A novel approach for protein subcellular location prediction using amino acid exposure. *BMC bioinformatics*, 14(1):342, 2013.

[14]W. R. Pearson. An introduction to sequence similarity ("homology") searching. *Current protocols in bioinformatics*, pages 3–1, 2013.

[15]A. Reinhardt and T. Hubbard. Using neural networks for prediction of the subcellular location of proteins. *Nucleic acids research*, 26(9):2230–2236, 1998.

[16]L. Wei, Y. Ding, R. Su, J. Tang, and Q. Zou. Prediction of human protein subcellular localization using deep learning. *Journal of Parallel and Distributed Computing*, 2017.

[17]D. Xie, A. Li, M. Wang, Z. Fan, and H. Feng. Locsvmpsi: a web server for subcellular localization of eukaryotic proteins using svm and profile of psi-blast. *Nucleic acids research*, 33(suppl_2):W105–W110, 2005.

[18]S. Zhang, X. Xia, J. Shen, Y. Zhou, and Z. Sun. Dbmloc: a database of proteins with multiple subcellular localizations. *BMC bioinformatics*, 9(1):127, 2008.