Automatic Discovery of Bioluminescent Proteins from Large Protein Databases

Tao Meng, Mei-Ling Shyu
Department of Electrical and Computer Engineering
University of Miami
Coral Gables, FL 33124
Email: t.meng@umiami.edu, shyu@miami.edu

Hua Zhang
School of Computer and Information Engineering
Zhejiang Gongshang University
Hangzhou 310018, PR. China
Email: zerozhua@126.com

***Abstract*—Accurate annotation of different protein features becomes increasingly important in enriching gene ontology databases. In this work, we present a framework to predict the bioluminescence of any given protein sequence. Bioluminescent proteins are produced by living organisms and emit light naturally. Bioluminescence is deemed to have different functions in living organisms including camouflage, attraction to prey, communication, etc. In addition, bioluminescent proteins are also widely used as labels in assay development, reporters of gene expression, and imaging agents in biotechnology. Currently, bioluminescent proteins are mainly curated by researchers through experimental analysis, which is a time consuming process. However, the data mining based algorithms provide an efficient way to detect candidate bioluminescent proteins and suggest prioritization of the experimental work. While traditional alignment based algorithms (such as BLAST) show promising results in terms of sequence analysis, it suffers from the limitation that the testing sequence should show homology to the sequences in the available training data sets. In order to overcome such a limitation, our proposed framework uses a set of homology-independent features that are extracted directly from the primary sequences to represent the global physicochemical properties as well as the sequence order characteristics of proteins. In addition, a novel subspace-based data filtering algorithm is proposed to eliminate noise from the training data. One existing framework addressing the same problem was implemented and compared with our proposed framework. The experimental results indicate that our proposed framework shows promising performance. In addition, the proposed framework is generic and could easily be applied to annotations of other protein properties.**

***Keywords*-Bioluminescence, Classification, Lasso, Subspace-based filtering.**

I. Introduction

In recent years, next generation sequencing (NGS) has revolutionized biological research. For example, researchers can sequence the entire human genome and produce data in roughly one week, for a cost of less than $5000. However, the first human genome required 10 years to sequence and an additional three years to finish analysis, for a cost of nearly 3 billion USD. With the help of this technology, the amount of available protein/DNA sequences increases exponentially. This large volume of sequence data poses great challenges for data storage, search, and knowledge discovery in the bioinformatics and database research domain [1][2][3]. Since protein sequences often contain important hidden patterns and semantic information, automatic annotation of high level features of proteins based on their primary sequence has become a research challenge in the bioinformatics domain. In order to address this challenge, we propose a fully automatic framework for protein semantic information annotation and mining in this paper. Specifically, we use the annotation of the bioluminescent proteins as an application to present our framework.

Bioluminescence is common in various environments in which the light is emitted by a living organism. Light emitting mostly occurs in a remarkable variety of sea creatures, from bacteria to fish [4]. They display an extensive palette of visible fluorescence and coloring via the chemical generation of light for finding food, attracting mates, evading predators, camouflage, communication between bioluminescent bacteria (quorum sensing), illumination of prey, burglar alarm, etc. [4][5][6].

In part, the vibrant coloration, from violet through red, is due to a growing family of intrinsically fluorescent proteins [7]. Among these fluorescent proteins, the best understood so far is the green fluorescent protein (GFP) [8] that has been changed from a nearly unknown protein to a commonly used molecular imaging tool in biology, chemistry, genetics, and medicine. Consequently, the 2008 Nobel Chemistry Prize was awarded to Shimomura, Chalfie and Tsien for their pioneering discovery and development of GFP [9]. In the past decade, GFP and its numerous variants have led research in post-genomic era to direct visualization of biological functions as a powerful set of tools for living cell imaging [10]. They hold great promises for enabling the researchers to examine complex cellular context as biosensors [11][12][13][14][15], to study the protein-protein interactions using bioluminescence resonance energy transfer (BRET) [16][17], to act as live-cell markers in drug-discovery assays [18][19], and to guide the thermal treatment of cancer [20].

Regarding the potential biomedical and commercial importance, the identification of new bioluminescent proteins is desirable to be detected that may help to understand more functions in live-cell and to design new medical applications. Until now, both experimental and computational methods have been developed to investigate the bioluminescent proteins [21][22][23]. However, the experiments for the annotation of proteins are in general time-consuming and expensive with the limited application to the available huge amount of data by the advanced sequencing techniques [24]. On the other hand, the data mining and machine learning techniques could provide fast, high-throughput, and automatic solutions to many biological problems and have been widely used in the bioinformatics domain [25][26][27][28][29]. This motivates us to develop a data mining based application to detect bioluminescent proteins automatically.

There have been some existing studies that attempted to address bioluminescent prediction research issues. In [30], the Positive Specific Iterated BLAST (PSI-BLAST) algorithm is utilized to extract the evolutionary information for each protein sequence. In essence, this method is a sequence alignment based approach and relies on sequence homology. Therefore, although the reported accuracy was relatively high in their work, the performance could be adversely affected in the case that the testing protein sequences do not show homology to the training protein sequences. In [31], the researchers used the amino acid index [32] to represent each protein sequence. The amino acid index based features capture the physicochemical properties of the protein and do not rely on the sequence homology, which shows an advantage. On the other hand, the amino acid index features of the protein lose all sequence order information which sometimes determines the properties of the protein. Therefore, it would be better to develop a new framework which does not rely on homology but is able to capture the sequence order information of the protein in order to further improve the accuracy of the bioluminescent protein prediction.

Given the practical needs and the limitations of the previous work, a high-throughput bioluminescent protein prediction pipeline is proposed in this paper. In order to capture the homology-independent characteristics of the protein, a new feature representation that includes the amino acid index [32], pseudo amino acid composition [33], and other sequence-based features is proposed to represent each protein sequence. Among these features, the pseudo amino acid composition feature is one of the well-acknowledged protein features used in data mining and machine learning fields, and it not only covers the composition of each type of amino acid in a protein sequence, but also contains the sequence order information. This extra information could recover the lost information by using the amino acid index features alone. Because of the internal diversity and the fuzziness of the biological sequences, the data could contain noise, which hurts the classification model. Accordingly, we also propose a novel subspace-based data/instance filtering algorithm to clean the training data set with an attempt to train a better classification model. In this paper, the Support Vector Machine (SVM) classification model is utilized to build the classifier using the pruned training data set. In addition, other data processing modules such as feature selection are also integrated into the pipeline to improve the final predication results. The whole pipeline is fully automatic and could be easily deployed in a biological research lab for practical use. To evaluate our proposed framework, several experiments are conducted and comparison with the "BLProt" framework [31] demonstrates the promising performance of our proposed framework as well as our proposed feature representation and subspace-based instance filtering algorithm.

The paper is organized as follows. The proposed framework is introduced in Section II. The experimental results and our observations are described in details in Section III. Section IV concludes the paper and discusses some future research directions.

II. The Proposed Framework

Figure 1 and Figure 2 show an overview of the proposed framework. It consists of the training phase (Figure 1) and the testing phase (Figure 2). In the training phase, a novel feature representation including a set of sequence-based features is proposed to represent the protein sequence. These features are extracted based on the training protein sequences. Since they are of different scales, the features are normalized to prevent the features of a large scale from overshadowing the features of a small scale. Next, the feature selection module is included to select the most significant features for this classification task. The training instances are selected randomly from the database and could contain noise which affects the accuracy of the classification model. Accordingly, a novel subspace-based instance filtering algorithm is applied to eliminate the outliers in the training data set. The support vector machine (SVM) with the radial basis function (RBF) kernel is used as the classification model. The parameters of the model are selected in a ten-fold cross validation process. The normalization parameters, selected feature indices, and the SVM model with the optimized parameters are stored for the testing phase.

In the testing phase, the same set of features are extracted for the testing protein sequences. Using the normalization parameters computed in the training phase, the testing features are normalized in the same way. Next, only the features corresponding to the selected feature indices are kept. The testing data are then classified by the trained SVM model. Different evaluation criteria are adopted to evaluate the framework. Because the testing instances are independent, they are not exposed to the training process. As a result, the evaluation done on these instances simulates a real world case. The details of each component are introduced in the following subsections. For convenience, the bioluminescent protein sequences are named positive data instances or positive instances in this paper; while the non-bioluminescent protein sequences are named negative data instances or negative instances.

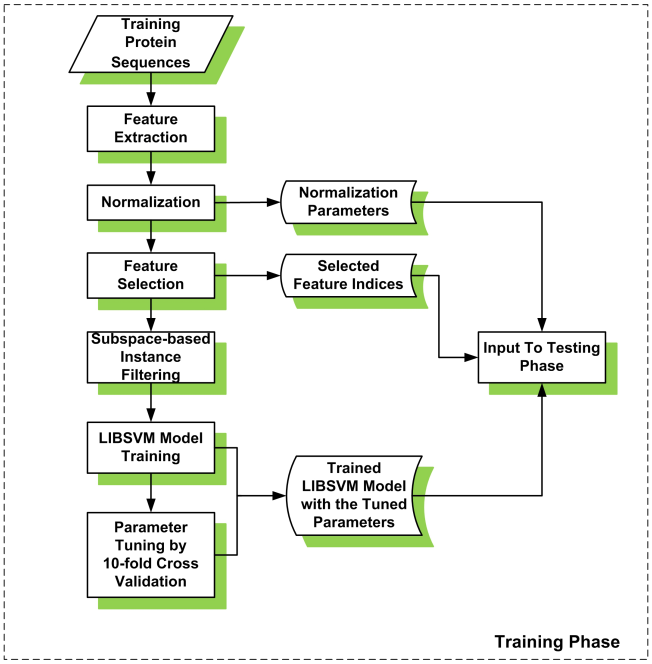

Figure 1. The Proposed Framework - Training Phase

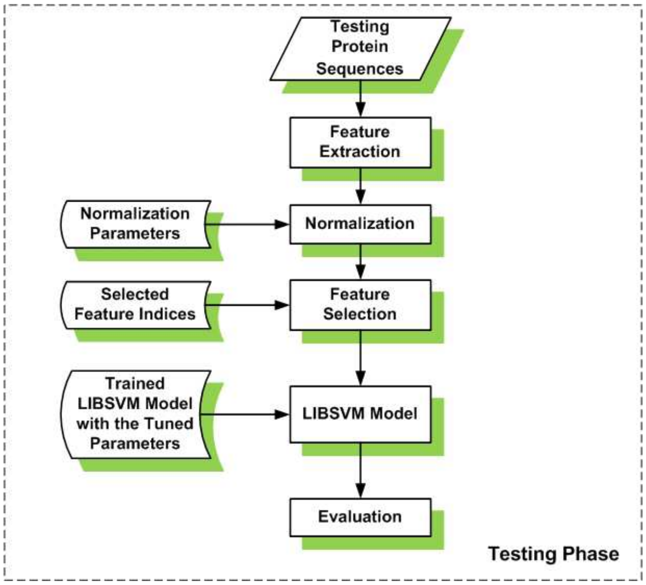

Figure 2. The Proposed Framework - Testing Phase

A. *Feature Extraction and Feature Normalization*

Before applying any classification model, each protein sequence is represented numerically as a feature vector which captures the characteristics of the original sequence. In this paper, a new feature representation is proposed to represent the protein sequence, which includes the amino acid index (AAIndex) [32], pseudo amino acid composition [33], and peptide property features provided by the BioJava framework [34].

The AAindex database stores the numerical indices representing different physicochemical and biochemical properties of all the amino acids and pairs of amino acids. The AAindex database contains AAindex1, AAindex2, and AAindex3 data sets. The AAindex1 data set which contains 544 amino acid indices is used to represent each protein sequence as a 544-D vector. Specifically, for each protein sequence S, let $S^{(q)}$ indicate the q-th amino acid in S and l is the total number of amino acids in S. Suppose for any amino acid A, $f_b(A)$ $(1 \leq b \leq 544)$ indicates the b-th index for amino acid A in the AAindex1 data set. The b-th feature F_b for protein sequence S is computed using Equation (1).

$$F_b = \frac{1}{l}\sum_{q=1}^{l} f_b(S^{(q)}). \tag{1}$$

Pseudo amino acid composition (PAAC) is one of the most popular features used in the protein sequence mining field. The advantage of this method is that it is able to capture the sequence-order information of the protein sequence. The details of the feature extraction algorithms are introduced in [33] and [35]. The open source application PseAAC-Builder [36], which implements the pseudo amino acid composition computation algorithm, is used to extract the features from all the sequences automatically. The algorithm provides flexibility so that the users can set the parameters according to specific applications. Based on the results of the empirical study, we use the Type I PAAC and set the number of features and the weight factor to 40 and 0.4, respectively.

BioJava [34] is an open source framework for DNA and protein sequence analysis built on the Java platform. The “IPeptideProperties” class provides several properties for a protein sequence including molecular weight, absorbance and extinction coefficient, etc. We capitalize on this facility and extract 41 features which supplement the original feature set.

After extracting the aforementioned three kinds of features, all the feature values are concatenated to form the final feature vector which is 625-dimensional. Since the features are of different scales, Z-score normalization is applied to make all features comparable. Assume $X_d^{(i)}$ represents the d-th feature of protein sequence i and the total number of the training instances is m, the feature value after Z-score normalization is $Z_d^{(i)}$ which is computed using Equation (2).

$$Z_d^{(i)} = \frac{X_d^{(i)} - \mu_d}{\sigma_d}, \tag{2}$$

$$\text{where } \mu_d = \frac{1}{m}\sum_{i=1}^{m} X_d^{(i)},$$

$$\text{and } \sigma_d = \frac{1}{m}\sum_{i=1}^{m} (X_d^{(i)} - \mu_d)^2.$$

μ_d and σ_d are the normalization parameters which are computed using the training data and saved for the testing phase. For a testing instance, the same equation is used to compute the normalized feature value after plugging in these parameters.

B. Feature Selection

A large set of features usually increases the computational cost and affects the accuracy of classification because of the curse of dimensionality problem. Therefore, finding the most significant features before applying the classification model helps improve the overall performance of the framework. However, the feature selection problem is NP-hard under many different cases and calls for a heuristic solution. We choose one of the shrinkage methods, which is Lasso regression [37], as the feature selection algorithm because it can select the important features and safely discard the unimportant ones by setting the corresponding weights to zero. In addition, the selected features usually reveal some biological insights. The classic Lasso regression method aims at solving the following optimization problem:

$$\hat{\beta}^{Lasso} = \underset{\beta}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{m} (y^{(i)} - \beta_0 - \sum_{d=1}^{D} x_d^{(i)} \beta_d)^2 + \lambda \sum_{d=1}^{D} |\beta_d| \right\}$$

Here, m indicates the total number of training instances, D is the dimension of the feature vector, $x_d^{(i)}$ $(1 \le i \le m, 1 \le d \le D)$ indicates the feature value for feature d of instance i; while $y^{(i)}$ is the response of instance i. β=$[\beta_0, \beta_1, ..., \beta_D]$ is the regression coefficient vector and λ is the regularization parameter. By choosing different parameters λ, the features corresponding to the non-zero coefficients of β are selected. The method was extended in [38] to be used in the classification task and is used in our framework. The Glmnet [39] software implementation of the Lasso algorithm is integrated into our framework. The selection of the regularization parameter λ and more discussions of the selected features are introduced in Section III-B.

C. Subspace-based Instance Filtering

Most supervised classification models need to learn important parameters from the training instances in order to construct the models. Therefore, the quality of the training data could affect the performance of the overall classification model directly. If the data set contains a lot of outliers or noise, the model built on these data could be biased. Accordingly, an important question is how we can detect the outliers and eliminate noise from the training data set.

Some previous work eliminates noisy data based on Mahalanobis distance [25][26]. However, such a method suffers from the limitation that if the number of features is greater than the number of data instances and the covariance matrix is singular, the Mahalanobis distance can not be computed. In order to overcome this difficulty, we propose the subspace-based algorithm based on the idea of collateral representative subspace projection modeling (CRSPM) to filter the noisy instances [40][41][42][43]. Specifically, suppose all the positive instances form a matrix $\boldsymbol{P} = \{p_{uv}\}$, u = 1, 2, ..., U, and v = 1, 2, ..., V, where U indicates the total number of positive training data instances and V is the total number of selected features. The principal component analysis is applied on matrix $\boldsymbol{P}$ and the first K principal components are retained so that 99% of the variance of the positive instances is kept. Suppose the retained principal components span the K-dimensional subspace which is represented by W, by projecting $\boldsymbol{P}$ to W, a new data matrix $\boldsymbol{E} = \{e_{uk}\}$ in which the row vector $\boldsymbol{e_u} = [e_{u1}, e_{u2}, ...e_{uk}, ..., e_{uK}]$ $(K \le V)$ is the projection of the instance $\boldsymbol{p_u} = [p_{u1}, p_{u2}, ..., p_{uV}]$ in the original matrix $\boldsymbol{P}$. For each instance u in $\boldsymbol{E}$, a distance value C for instance u is computed using Equation (3).

$$C^{(u)} = \sum_{k=1}^{K} \frac{e_{uk}^2}{\alpha_k}, \tag{3}$$

where α_k is the singular value corresponding to the k-th principal component and indicates the variance on the k-th dimension. The C values for all the U positive instances could be computed and sorted in an ascending order. If the sorted C value list is represented as L, where

$$L = \{L(1), L(2), \ldots, L(U)\},$$

and $L(1) \le L(2) \le ... \le L(U)$. The data instances whose C values are greater than a certain threshold T are eliminated from the training data set. The threshold is determined using Equation (4).

$$\begin{aligned} T &= L(G) \qquad (4) \\ \text{where } G &= \underset{g}{\operatorname{argmax}}(L(g+1) - L(g)) \\ \text{subject to } 1 &\le g \le (U-1) \end{aligned}$$

The same algorithm is also applied to the negative instances. In order to illustrate this idea clearly, a numerical example is given here. In one round of the experiments, we use 300 positive instances. For all positive instances, the C values are computed and the list L is formed. The C values in the list according to the ranks are plotted in Figure 3. The G computed in this case is 295. Correspondingly, as is shown in the figure, the gap between $L(245)$ and $L(246)$ is the largest.

D. Classification Model and Evaluation Criteria

The support vector machine (SVM) is a supervised pattern classification model [44]. It is extended from the maximal margin classifier and utilizes the kernel trick to map the data instances to a higher dimensional space where they could be classified easier than the original space. The main idea

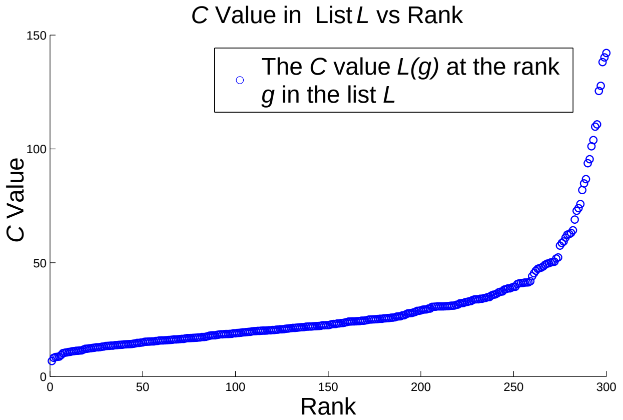

Figure 3. The Plot of Values in the List L

of SVM is to find the optimal hyperplane to maximize the distance between the hyperplane and the support vectors of the two classes. Specifically, given the training data set of the instance-label pairs $(x^{(i)}, y^{(i)})$, $i = 1, 2, \ldots, m$, where m is the total number of data instances, $x^{(i)} \in R^n$ and $y^{(i)} \in \{1, -1\}$ indicating the instance is positive or negative. The SVM model requires the solution of the following optimization problem:

$$\underset{w,b,\xi}{\text{argmin}} \frac{1}{2} w^T w + Q \sum_{i=1}^{m} \xi^{(i)}$$

$$\text{subject to } y^{(i)}(w^T \varphi(x^{(i)}) + b) \geq 1 - \xi^{(i)}$$

$$\xi^{(i)} \geq 0$$

The training vectors $x^{(i)}$ are mapped to a higher dimensional space using the function φ. In order to apply the kernel trick, the kernel function $H(x^{(i)}, x^{(j)}) \equiv \varphi(x^{(i)})^T \varphi(x^{(j)})$ is defined. The RBF kernel in Equation (5) is applied in this paper. More details of SVM could be found in [44] and are skipped here.

$$H(x^{(i)}, x^{(j)}) = exp(-\gamma \left\| x^{(i)} - x^{(j)} \right\|^2). \tag{5}$$

The implementations of the SVM classification model used in this paper is the LIBSVM [45] package which is an off-the-shelf SVM software implementation. The parameter Q and γ are optimized by the grid search approach using 10-fold cross validation.

In order to evaluate the current framework, the following evaluation criteria for binary classification are used in this paper.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Precision(P) = \frac{TP}{TP + FP}$$

$$Sensitivity(R) = \frac{TP}{TP + FN}$$

$$F1 = \frac{2PR}{P + R}$$

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

$$AUC: \text{area under the ROC curve}$$

Here, TP, FP, TN, and FN stand for the numbers of true positive instances, false positive instances, true negative instances, and false negative instances in the prediction result. The MCC stands for the Matthew's correlation coefficient. The range of MCC is from -1 to 1. $MCC = 1$ indicates the algorithm gives the best possible predictions; while $MCC = -1$ indicates the algorithm gives the worst possible predictions. Among these evaluation criteria, the accuracy and MCC evaluate the performance of the framework from the perspectives of both the positive instances and the negative instances. F1 focuses on the positive instances only.

III. Experimental Results and Discussion

Several experiments are conducted to evaluate our proposed framework as well as the feature selection (feature representation) module and the subspace-based instance pruning algorithm.

A. Data Set

In order to compare with the existing framework, the data set provided in [31] is utilized to evaluate the proposed framework. The data set contains 441 bioluminescent proteins (positive instances) and 18202 non-bioluminescent proteins (negative instances). Following the same procedure in [31], we randomly select 300 positive instances and 300 negative instances to form the training data set and leave the rest 141 positive instances and the 17902 negative instances as the independent testing data set. It could be seen that the number of negative instances is about 126 times of the number of positive instances in the testing data set. This simulates the real world case that most proteins are non-bioluminescent. It is important to point out that the negative instances in the training data set and the testing data set are not exactly the same as those in [31] because the authors of [31] did not provide the exact 300 training negative instances they used in their paper.

B. Feature Selection Result

As described in Section II-B, the LASSO algorithm is applied in this paper to select the significant features. By increasing the λ value in Equation (3), more significant features are selected. In order to decide the suitable value for parameter λ, an empirical study is carried out to select the λ value to maximize the cross validation accuracy.

As a result, a total number of 87 features are selected. Among these features, the features of the protein composition of the Tryptophan (W) and Tyrosine (Y) are selected. As is well-acknowledged, both of these two amino acids contain aromatic amino acid residues and contribute to the fluorescence of the proteins. From this point of view, the feature selection procedure could also provide some insights about the features and these insights could further the understanding of the relative domain.

C. Classification Performance

In order to train the framework, the 10-fold cross validation is applied to decide several important parameters. Since the "Accuracy" values could reflect the performance of the model on both positive and negative instances, they are used as the optimization criterion in this work.

Table I shows the experimental results for the independent testing data set that compare the performance of three frameworks, namely "BLProt", "Proposed (No Filter)", and "Proposed (Filter)". "BLProt" indicates the framework proposed in [31]. "Proposed (No Filter)" indicates the proposed framework without applying the proposed subspace-based instance filtering strategy introduced in Section II-C. "Proposed (Filter)" indicates the proposed framework with the proposed subspace-based instance filtering strategy. It is apparent that the proposed framework outperforms the "BLProt" framework in terms of all the evaluation criteria with and without utilizing the subspace-based filtering strategy. This suggests that our new feature representation and feature selection module are able to select a better set of features to represent the protein sequences. By incorporating the pseudo amino acid composition features and the peptide property features, the performance of the framework can be enhanced significantly. It could also be observed that the proposed subspace-based instance filtering strategy could further improve the performance of the framework. It is well understood that the support vector machine (SVM) classification model relies on the support vectors which are close to the decision boundary, and hence the noisy data could significantly affect the performance of the SVM model. Our proposed filtering approach helps reduce the noise and improve the performance of the SVM model.

Another observation from the results is that the precision is especially low, which leads to the low F1 value. One possible reason for this is the relative high negative to positive ratio in the independent testing data set. Such an imbalanced data set may lead to the misclassification of many negative instances as positive. This suggests that a proper testing data instances filtering module may further improve the overall performance of the framework.

IV. Conclusion and Future Work

In this paper, an automatic pipeline framework for predicting bioluminescent proteins is proposed. The protein sequences are first converted to a new feature representation - a feature vector based on three different types of features. The most significant features are then selected from the feature pool and may be used to provide insight on the application domain. By applying the proposed subspace-based instance filtering strategy to the training data set, the noisy data in the training data set are eliminated. The retained training data are then used to train the LIBSVM classification model. Different evaluation criteria are adopted to evaluate the performance of the proposed framework on a relatively large independent testing data set. The proposed framework is compared with the state-of-the-art framework using the same data set. The experimental results reveal that our proposed framework with the novel feature representation and the subspace-based instance filtering algorithm outperforms the existing framework in the comparison under all evaluation criteria.

Our proposed framework is relatively generic and could be utilized to annotate other protein properties by adjusting the relevant parameters. Therefore, we plan to extend our current framework to cover other protein properties in the future work. In addition, including the module to filter the testing instances will be investigated. Finally, more features will be explored to improve the classification performance.

Acknowledgment

The authors would like to thank Mr. Yang Liu from the Department of Electrical and Computer Engineering at the University of Miami for his help in data processing.

References

[1] Y. Li, A. Terrell, and J. M. Patel, "Wham: A high-throughput sequence alignment method," in *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*, June 2011, pp. 445–456.

[2] B. G. Richter and D. P. Sexton, "Managing and analyzing next-generation sequence data," *PLOS Computational Biology*, vol. 5, no. 6, June 2009.

Table I
EXPERIMENTAL RESULTS OF THE INDEPENDENT DATA SET

Framework	Accuracy	Precision	Sensitivity	Specificity	F1	MCC	AUC
BLProt	0.8182	0.0327	0.7801	0.8185	0.0628	0.1354	0.8727
Proposed (No Filter)	0.8543	0.0410	0.7872	0.8549	0.0779	0.1583	0.8846
Proposed (Filter)	0.8748	0.0431	0.7890	0.8758	0.0817	0.1598	0.8870

[3] J. K. Martin Kircher, "High-throughput dna sequencing-concepts and limitations," *BioEssays*, vol. 32, no. 6, pp. 524–536, June 2010.

[4] E. A. Widder, "Bioluminescence in the ocean: Origins of biological, chemical, and ecological eiversity," *Science*, vol. 328, no. 5979, pp. 704–708, May 2010.

[5] S. H. Haddock, M. A. Moline, and J. F. Case, "Bioluminescence in the sea," *Annual Review of Marine Science*, vol. 2, no. 1, pp. 443–493, January 2010.

[6] T. Wilson and J. W. Hastings, "Bioluminescence," *Annual Review of Cell and Development Biology*, vol. 14, pp. 197–230, November 1998.

[7] G. Jach and J. Winter, "Focus on fluorescent proteins," *Studies in Natural Products Chemistry*, vol. 33, pp. 3–67, May 2006.

[8] R. Y. Tsien, "The green fluorescent protein," *Annual Review of Biochemistry*, vol. 67, pp. 509–544, July 1998.

[9] A. Miyawaki, "Green fluorescent protein glows gold," *Cell*, vol. 135, no. 6, pp. 987–990, December 2008.

[10] V. Sample, R. H. Newman, and J. Zhang, "The structure and function of fluorescent proteins," *Chemical Society Reviews*, vol. 38, no. 10, pp. 2852–2864, October 2009.

[11] D. M. Chudakov, M. V. Matz, S. Lukyanov, and K. A. Lukyanov, "Fluorescent proteins and their applications in imaging living cells and tissues," *Physiological Reviews*, vol. 90, no. 3, pp. 1103–1163, 2010.

[12] O. Griesbeck, "Fluorescent proteins as sensors for cellular functions," *Current Opinion in Neurobiology*, vol. 14, no. 5, pp. 636–641, October 2004.

[13] A. E. Palmer, Y. Qin, J. G. Park, and J. E. McCombs, "Design and application of genetically encoded biosensors," *Trends in Biotechnology*, vol. 29, no. 3, pp. 144–152, January 2011.

[14] O. V. Stepanenko, V. V. Verkhusha, I. M. Kuznetsova, V. N. Uversky, and K. K. Turoverov, "Fluorescent proteins as biomarkers and biosensors: Throwing color lights on molecular and cellular processes," *Current Protein and Peptide Science*, vol. 9, no. 4, pp. 338–369, August 2008.

[15] B. Wu, K. D. Platkevich, T. Lionnet, R. H. Singer, and V. V. Verkhusha, "Modern fluorescent proteins and imaging technologies to study gene expression, nuclear localization, and dynamics," *Current Opinion in Cell Biology*, vol. 23, no. 3, pp. 310–317, June 2011.

[16] F. Ciruela, "Fluorescence-based methods in the study of protein-protein interactions in living cells," *Current Opinion in Biotechnology*, vol. 19, no. 4, pp. 338–343, 2008.

[17] A. Dragulescu-Andrasi, C. T. Chan, A. De, T. F. Massoud, and S. S. Gambhir, "Bioluminescence resonance energy transfer (bret) imaging of protein-protein interactions within deep tissues of living subjects," *Proceedings of the National Academy of Sciences*, vol. 108, no. 29, pp. 12 060–12 065, July 2011.

[18] C. Chakraborty, C.-H. Hsu, Z.-H. Wen, and C.-S. Lin, "Recent advances of fluorescent technologies for drug discovery and development," *Current Pharmaceutical Design*, vol. 15, no. 30, pp. 3552–3570, 2009.

[19] M. Wolff, J. Wiedenmann, G. U. Nienhaus, M. Valler, and F. Heilker, "Novel fluorescent proteins for high-content screening." *Drug Discovery*, vol. 11, no. 23-24, pp. 1054–1060, December 2006.

[20] J. T.Au, L. Gonzalez, C.-H. Chen, I. Serganova, and Y. Fong, "Bioluminescence imaging serves as a dynamic marker for guiding and assessing thermal treatment of cancer in a preclinical model," *Annals of Surgical Oncology*, vol. 19, March 2012.

[21] O. Shimomura, F. H. Johnson, and Y. Saiga, "Extraction, purification and properties of aequorin, a bioluminescent protein from the luminous hydromedusan, aequorea," *Journal of Cellular Physiology*, vol. 59, no. 3, pp. 223–239, June 1962.

[22] P.-A. Vidi and V. J. Watts, "Fluorescent and bioluminescent protein-fragment complementation assays in the study of g protein-coupled receptor oligomerization and signaling," *Molecular Pharmacology*, vol. 75, no. 4, pp. 733–739, April 2009.

[23] H. Yu, M. West, B. H. Keon, G. K. Bilter, S. Owens, J. Lamerdin, and J. K. Westwick, "Measuring drug action in the cellular context using protein-fragment complementation assays," *Current Opinion in Cell Biology*, vol. 1, no. 6, pp. 811–822, December 2003.

[24] M. L. Metzker, "Sequencing technologies - the next generation," *Nature Reviews Genetics*, vol. 11, no. 1, pp. 31–46, January 2010.

[25] T. Meng, L. Lin, M.-L. Shyu, and S.-C. Chen, "Histology image classification using supervised classifiation and multimodal fusion," in *IEEE International Symposium on Multimedia*, December 2010, pp. 145–152.

[26] T. Meng, M.-L. Shyu, and L. Lin, "Multimodal information integration and fusion for histology image classification," *International Journal of Multimedia Data Engineering and Management*, vol. 2, no. 2, pp. 54–70, April-June 2011.

[27] H. Zhang, T. Zhang, J. Gao, J. Ruan, S. Shen, and L. Kurgan, "Determination of protein folding kinetic types using sequence and predicted secondary structure and solvent accessibility," *Amino Acids*, vol. 42, no. 1, pp. 271–283, 2012.

[28] X. Bi, H. Huang, S. Matis-Mitchell, P. Mcgarvey, M. Torii, H. Shatkay, and C. Wu, "Building a classifier for identifying sentences pertaining to disease-drug relationships in tardive dyskinesia," in *Proceedings of the 2012 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2012, pp. 1–4.

[29] P. P. Kuksa, "Biological sequence classification with multivariate string kernels," *IEEE Transactions on Computational Biology and Bioinformatics*, no. 99, 2013.

[30] X. Zhao, J. Li, Y. Huang, Z. Ma, and M. Yin, "Prediction of bioluminescent proteins using auto covariance transformation of evolutional profiles," *International Journal of Molecular Sciences*, vol. 13, no. 3, pp. 3650–3660, March 2012.

[31] K. K. Kandaswamy, G. Pugalethi, M. K. Hazrati, K.-U. Kalies, and T. Martinetz, "Blprot:prediction of bioluminescent proteins based on support vector machine and relieff feature selection," *BMC Bioinformatics*, August 2011.

[32] S. Kawashima, P. Pokarowski, M. Pokarowska, A. Kolinski, T. Katayama, and M. Kanehisa, "Aaindex: Amino acid index database, progress report 2008," *Nucleic Acids Research*, vol. 36, pp. D202–D205, January 2008.

[33] K.-C. Chou, "Prediction of protein cellular attributes using pseudo amino acid composition," *Proteins*, vol. 43, no. 3, pp. 246–255, May 2001.

[34] R. C. G. Holland, T. A. Down, M. Pocock, A. Prlic, D. Huen, K. James, S. Foisy, A. Drager, A. Yates, M. Heuer, and M. J. Schreiber, "Biojava:an open-source framework for bioinformatics," *Bioinformatics*, vol. 24, no. 18, pp. 2096–2097, September 2008.

[35] K.-C. Chou, "Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes," *Bioinformatics*, vol. 21, no. 1, pp. 10–19, January 2005.

[36] P. Du, X. Wang, C. Xu, and Y. Gao, "Pseaac-builder: A cross-platform stand-alone program for generating various special chou's pseudo-amino acid compositions," *Analytical Biochemistry*, vol. 425, no. 2, pp. 117–119, June 2012.

[37] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society, Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[38] T. T. Wu, Y. F. Chen, T. Hastie, E. Sobel, and K. Lange, "Genome-wide association analysis by lasso penalized logistic regression," *Bioinformatics*, vol. 25, no. 6, pp. 714–721, March 2009.

[39] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *Journal of Statistical Software*, vol. 33, no. 1, pp. 1–22, August 2010.

[40] T. Quirino, Z. Xie, M.-L. Shyu, S.-C. Chen, and L. Chang, "Collateral representative subspace projection modeling for supervised classification," in *Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence*, November 2006, pp. 98–105.

[41] L. Lin, C. Chen, M.-L. Shyu, and S.-C. Chen, "Weighted subspace filtering and ranking algorithms for video concept retrieval," *IEEE Multimedia*, vol. 18, no. 3, pp. 32–43, July-September 2011.

[42] M.-L. Shyu, Z. Xie, M. Chen, and S.-C. Chen, "Video semantic event/concept detection using a subspace-based multimedia data mining framework," *IEEE Transactions on Multimedia*, vol. 10, no. 2, pp. 252–259, February 2008.

[43] M.-L. Shyu, C. Chen, and S.-C. Chen, "Multi-class classification via subspace modeling," *International Journal of Semantic Computing*, vol. 5, no. 1, pp. 55–78, 2011.

[44] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, June 1998.

[45] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machine," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 27:1–27:27, 2011.