Open Access

Estimation of Subcellular Proteomes in Bacterial Species

Brian R. King[1,2], Lance Latham[1,2] and Chittibabu Guda[*,2,3]

[1]Department of Computer Science, [2]GenNYsis Center for Excellence in Cancer Genomics, [3]Department of Epidemiology and Biostatistics, State University of New York at Albany, One Discovery Drive, Rensselaer, NY 12144-3456, USA

Abstract: Computational methods for predicting the subcellular localization of bacterial proteins play a crucial role in the ongoing efforts to annotate the function of these proteins and to suggest potential drug targets. These methods, used in combination with other experimental and computational methods, can play an important role in biomedical research by annotating the proteomes of a wide variety of bacterial species. We use the ngLOC method, a Bayesian classifier that predicts the subcellular localization of a protein based on the distribution of *n*-grams in a curated dataset of experimentally-determined proteins. Subcellular localization was predicted with an overall accuracy of 89.7% and 89.3% for Gram-negative and Gram-positive bacteria protein sequences, respectively. Through the use of a confidence score threshold, we improve the precision to 96.6% while covering 84.4% of Gram-negative bacterial data, and 96.0% while covering 87.9% of Gram-positive data. We use this method to estimate the subcellular proteomes of ten Gram-negative species and five Gram-positive species, covering an average of 74.7% and 80.6% of the proteome for Gram-negative and Gram-positive sequences, respectively. The current method is useful for large-scale analysis and annotation of the subcellular proteomes of bacterial species. We demonstrate that our method has excellent predictive performance while achieving superior proteome coverage compared to other popular methods such as PSORTb and PLoc.

INTRODUCTION

High-throughput experimental methods continue to generate large repositories of genomic and proteomic data that must be analyzed and annotated. Unfortunately, experimental methods are prohibitively expensive, thus limiting their use in large-scale functional annotation of these proteins. Computational methods have become a crucial part of the ongoing efforts to annotate these growing sequence data repositories. Computational prediction of protein subcellular localization has been an active research area because knowledge of the subcellular localization of a protein can aid in inferring its function. Accurate prediction of subcellular proteomes can improve our understanding of defined cellular processes at various subcellular levels. Such methods can aid in suggesting plausible candidates for drug targeting in bacterial pathogen research, which is important because immunogenic proteins targeted to the cell surface make ideal drug targets.

Gram-negative bacterial cells are primarily composed of the cytoplasm, an inner membrane surrounding the cytoplasm, and a cell envelope consisting of an outer membrane and an area between the membranes known as the periplasm. Gram-positive bacteria have a slightly different structure, in that they have a peptidoglycan cell wall surrounding the inner/cytoplasmic membrane instead of a defined outer membrane and periplasm. Bacterial proteins are synthesized in the cytoplasm of the cell. These proteins remain in the cytoplasm, or are targeted to one or more possible locations in the cell through various transport systems and signal peptides in the sequence. Proteins that are transported to the extracellular space are of particular interest since they are ideal candidates for drug targeting. Although many of these transport systems are well understood, the total range of possible targeting mechanisms has yet to be characterized [1, 2]. Regardless, the signaling and transport mechanisms for the protein, along with most of the structural and functional attributes for the protein, lie in the primary structure of the protein – the sequence of amino acids. Computational methods that predict subcellular localization from sequence information alone capitalize on this knowledge.

Numerous methods have been developed for prediction of subcellular localization of bacterial proteins [2-11]. Some methods explicitly use prior knowledge of known transport and signaling mechanisms; some rely solely on machine-learned classifiers to implicitly represent this knowledge, while others use a combination of both to generate the prediction for a protein. The PSORT-I method [3], one of the first methods available for subcellular localization prediction, was a rule-based system that used amino-acid composition, known motifs, targeting signals and other structural information to make predictions for four Gram-negative and three Gram-positive subcellular locations. This work has been improved with the release of the PSORTb method [4, 5], which expanded the number of subcellular locations predicted as well as the coverage of predicted proteins. The PSORTb approach is similar to the original PSORT-I method, in that both apply prior knowledge of various features known to correlate to distinct localizations. The predictive performance of the PSORTb method was improved by the use of modern classification algorithms, such as support vector machines (SVM) and through integration of each individual component output *via* a Bayesian network to

*Address correspondence to this author at the GenNYSis Center for Excellence in Cancer Genomics, State University of New York at Albany, One Discovery Drive, Room 208, Rensselaer, NY 12144, USA; Tel: (518) 591-7155; Fax: (518) 591-7151; E-mail: cguda@albany.edu

1874-1363/09 2009 Bentham Open.

produce a final score. SVM-based methods are common in this field of research, with methods that focus on bacterial protein localization differing only in the feature space considered as the input to the classifier [5-9]. A few methods that extract keywords from the annotations associated with each sequence have appeared in recent years. The Proteome Analyst is one such method that can be used on bacterial proteins. The method works by BLASTing the query sequence against the Swiss-Prot database, selecting a set of the best homologous sequences, and extracting text associated with the homologous sequences [10, 12, 13]. For further information on bacterial protein subcellular localization prediction, we recommend that the reader refer to an excellent review by Gardy and Brinkman [2]. For a general review of many of the current methods and challenges in protein localization prediction, we recommend the reviews by Dönnes and Höglund [11] and Sprenger *et al.* [14].

Though there are a wide range of methods available for subcellular localization, most of these are not suitable for proteome-wide prediction. They often require prior structural or functional information for the protein sequences used for training purposes. This significantly limits the size and scope of datasets used, which results in datasets that are not sufficiently robust to represent entire proteomes. In addition, many methods can predict only a few prominent subcellular classes or usually perform poorly on highly unbalanced data, which is characteristic of datasets used in subcellular localization prediction.

In this study, we address the task of predicting the subcellular localization of bacterial proteins using information that can be derived from the sequence of the protein alone. Our work is based on previous work completed on the ngLOC method [15], which was developed to annotate the subcellular localization of eukaryotic proteins. The ngLOC method is an *n*-gram based Bayesian method based on sequence homology, and operates by observation of fixed length subsequences of length *n* (i.e. *n*-grams) over different subcellular localization classes. The method was designed to address many of the known limitations of existing methods that make them prohibitive for proteome-wide predictions.

MATERIALS AND METHODOLOGY

Training Dataset

The dataset used for this task is a set of protein sequences taken from the Swiss-Prot database, release 54 [16], which contains experimentally determined annotations for subcellular localization. We applied the following filters to obtain high-quality data for training and testing purposes: only bacterial (not archael) sequences were considered; sequences with predicted and ambiguous localizations were removed, including those annotated with the terms 'probable', 'potential', 'likely', or 'by similarity'; sequences shorter than 10 residues in length were removed. There are very few bacterial proteins in the Swiss-Prot database that have been experimentally determined to be multi-localized in the cell, and therefore we removed such sequences. We separated proteins from Gram-negative (i.e. have an outer membrane encasing an inner membrane, with a very thin cell wall) and Gram-positive (i.e. an inner membrane encased in a thick cell wall) species, resulting in 7,229 and 2,814 sequences, respectively. Finally, we used the cd-hit sequence clustering software [17] to reduce the maximum sequence identity to 98%, resulting in a final dataset of 6,558 Gram-negative sequences and 2,447 Gram-positive sequences. A further reduction in similarity resulted in loss of discriminatory *n*-grams (data not shown). Table **1** shows the location-wise distribution of the resulting datasets.

Proteome Datasets

The proteomes and corresponding PSORTb v2.0 predictions were downloaded from the PSORTdb online database of subcellular localization predictions for bacteria [5, 18]. We randomly chose ten Gram-negative organisms and five Gram-positive organisms for subcellular proteome estimation purposes. The Gram-negative examples include *Anaeromyxobacter dehalogenans*, *Campylobacter jejuni*, *Caulobacter crescentus*, *Escherichia coli*, *Geobacter sulfurreducens*, *Haemophilus influenzae*, *Helicobacter pylori*, *Neisseria gonorrhoeae*, *Rickettsia conorii*, and *Thiobacillus denitrificans*. The Gram-positive examples include *Bacillus anthracis*, *Clostridium acetobutylicum*, *Lactobacillus acidophilus*, *Mycobacterium tuberculosis*, and *Streptococcus pneumoniae.*

Prediction Algorithm

The ngLOC method is a Bayesian classification method for predicting protein subcellular localization as described in King and Guda [15]. Our method uses *n*-gram peptides derived solely from the primary structure of a protein to explore the search space of proteins. It is suitable for proteome-wide predictions, and is also capable of inferring multi-localized proteins, namely those localized to more than one

Table 1. **Gram-Negative and Gram-Positive Training Data**

Localization	Code	Gram-Negative Sequences	Gram-Negative % of Data	Gram-Positive Sequences	Gram-Positive % of Data
Cytoplasm	CYT	4,139	63.0%	1,776	72.6%
Extracellular	EXC	263	4.0%	292	11.9%
Inner Membrane	IN	1,397	21.3%	347	14.2%
Outer Membrane	OUT	344	5.2%	NA	NA
Periplasm	PER	415	6.3%	NA	NA
Cell Wall	WAL	NA	NA	32	1.3%
Total		**6,558**		**2,447**	

This table shows the distribution of proteins in the training data for both Gram-negative and Gram-positive bacteria over each subcellular location. NA – Not applicable.

subcellular location. We modified the original method to predict distinct subcellular localization classes over Gram-negative and Gram-positive bacterial proteins, and to emphasize a balance between high precision across all subcellular localization classes, versus high coverage across prokaryotic proteomes. The essential parts of this method are described here.

The core of the ngLOC method is based on a vast amount of work performed in document classification, where the popular naïve Bayes classifier has been used to effectively classify documents based on the frequency of occurrence of all possible words observed over different document classes [19]. In protein classification, we consider the frequency of occurrence of all possible *n*-grams in a dataset of protein sequences, where an *n*-gram is defined as a protein subsequence having a fixed length of *n*.

Given a protein sequence d_i, a probabilistic approach to subcellular localization prediction is to develop a model to estimate the probability that d_i is localized into each localization class $c_j \in \mathcal{C}$, where $\mathcal{C}$ represents the set of all possible such classes. The classifier *h* predicts the localization of d_i to the class that has the highest posterior probability. Equation 1 shows this in probabilistic terms, and shows how the well-known Bayes rule is used to derive an estimate for this probability.

$$h\left(d_i\right) = \arg\max_{c_j \in \mathcal{C}} P\left(c_j \mid d_i\right) = \arg\max_{c_j \in \mathcal{C}} \frac{P\left(d_i \mid c_j\right) P\left(c_j\right)}{P\left(d_i\right)} \tag{1}$$

An accurate Bayesian classifier is dependent on accurate estimates for the probabilities on the right-hand side of equation 1. The denominator $P(d_i)$ is dropped because it is constant. The prior probability of each subcellular localization c_j, denoted $P(c_j)$, is estimated from the training data by counting the number of sequences assigned to class c_j, divided by the total number of sequences in the training data. The posterior probability of protein sequence d_i, given location c_j, denoted as $P(d_i \mid c_j)$, is the difficult parameter to estimate. We assume that each sequence d_i is viewed as a collection of unordered *n*-grams generated by a random process that follows a multinomial distribution. Letting w_t denote the t^{th} *n*-gram over all possible *n*-grams that may occur in the data and N_{it} be a count of the number of occurrences of *n*-gram w_t in sequence d_i, then under our assumptions, the posterior probability of protein sequence d_i, given location c_j, is estimated as follows:

$$P\left(d_i \mid c_j\right) = \prod_t P\left(w_t \mid c_j\right)^{N_{it}} \tag{2}$$

To estimate $P(w_t \mid c_j)$, the posterior probability of *n*-gram w_t, given class c_j, we use the LaPlace correction to prevent probabilities of zero from being calculated and count the number of occurrences of w_t in all sequences belonging to that class. Letting *S* be the set of all possible *n*-grams that occur in the entire set of training data, we compute the estimate as:

$$P\left(w_t \mid c_j\right) = \frac{1 + Count\left(w_t\right) in\, class\, c_j}{\left|S\right| + \left(Total\ n - grams\ in\, class\, c_j\right)} \tag{3}$$

Probabilistic Confidence Score

We developed a probabilistic confidence score (CS) for sequence d_i being localized into each class c_j, and normalized it to sum to 100 over all classes. Our CS can be interpreted as an estimate of the conditional probability of class c_j, given sequence d_i and the *n*-gram model used.

For a given sequence d_i, we define d_{null} to be a sequence of null symbols of a length that is equal to the length of d_i. (A null symbol is essentially any symbol that does not represent an amino acid.) Thus, each instance of d_{null} is guaranteed to never occur in the model. To calculate the probability that each class model generated d_{null}, we can use this fact to simply Equations 2 and 3, giving us:

$$P\left(d_{null} \mid c_j\right) = \left(\frac{1}{\left|S\right| + \left(Total\ n - grams\ in\, class\, c_j\right)}\right)^{(k-n+1)} \tag{4}$$

We then set *minNullProb* to be the minimum joint probability of d_{null} and class c_j observed across all classes:

$$minNullProb = \min_{c_j \in C}\left(P\left(c_j\right) P\left(d_{null} \mid c_j\right)\right) \tag{5}$$

A log-odds ratio that sequence d_i is targeted for location c_j against *minNullProb* is calculated and then normalized by dividing by the sum over all log-odds scores, to create a separate score for each subcellular location c_j for a given sequence d_i as follows:

$$CS\left(c_j \mid d_i\right) = \frac{\log\left(P\left(c_j\right) P\left(d_i \mid c_j\right)\right) - \log\left(minNullProb\right)}{\sum_k \log\left(P\left(c_k\right) P\left(d_i \mid c_k\right)\right) - \log\left(minNullProb\right)} * 100 \tag{6}$$

The range for each score will always be between 0-100, with the sum of the scores over all classes totaling 100.

Evaluation Methods

To evaluate the performance of the method, we apply a standard validation technique known as 'leave-one-out' validation (also known as 'jack-knife'), in which one sequence is removed from the training data, the predictive model is trained using the remainder of the data, and the model is then tested on the single sequence that was removed. This process is repeated for the entire dataset, one sequence at a time. We report standard performance measures over each subcellular location, including sensitivity (recall), precision, specificity, and Matthews correlation coefficient (MCC). The latter provides a measure of performance for a single predicted class; the MCC value is 1 for perfect predictions on that class, 0 for random assignments, and less than 0 if predictions are worse than random [20]. (See supplementary material for more details on performance measurements.)

For evaluation of the overall classifier performance, we report overall accuracy as the fraction of the test datasets that were correctly classified. However, due to the highly unbalanced nature of the data over each localization class, overall accuracy is not a useful measure upon which to rely exclusively. For such unbalanced datasets, a good solution that has been used in information retrieval is to report *macro-averaged* class measures, which average the individual class measures over all classes [21]. We also report a macro-averaged F_1 measure, where the F_1 measure for a given class combines sensitivity and precision with equal weighting to produce a balanced measure [22].

Finally, we report a receiver operating characteristic (ROC) curve for each class as a graphical means of observing the discriminatory ability of the classifier. ROC curves have been increasingly adopted in the machine learning and data mining community as a more rigorous means of comparing classifiers, in part because researchers are realizing that simple accuracy measures are often a poor metric for measuring classifier performance [23, 24]. Moreover, it is known that resulting ROC curves plotting the classifier performance are independent of class distributions, which is an ideal consideration for our classification problem due to our highly unbalanced dataset. ROC curves typically plot the true positive rate (TPR, which is equivalent to sensitivity) against the false positive rate (FPR, which is equivalent to 1.0 - specificity). Individual points on the graph are plotted on the basis of selected scoring thresholds output by the classifier. The curve begins at the bottom-left corner of the plot area and rise to the top-right corner. A classifier that performs no better than random will result in a plot that is close to the diagonal. The closer a curve comes to the top-left corner, the better the classifier will be able to discriminate between positive and negative instances. We generate a single ROC curve for a classifier by computing the macro-average of the fraction of true positives and false positives over each class observed at distinct CS thresholds, denoted as a Mac-ROC curve. These curves help us determine the ability for the classifier to discriminate between true predictions and false predictions over the entire range of confidence scores for each class. We compute the area under the Mac-ROC curve as another single measure to evaluate the overall predictive performance of the classifier. The range of the AUC measure will be between 0.0 and 1.0, where 1.0 is a perfect classifier, and a 0.5 is a classifier that is performing no better than random, and less than 0.5 is worse than random guessing. This last measure is particularly useful to determine how well the confidence score generated by the ngLOC method discriminates between correct and incorrect predictions.

ngLOC-X – Proteome-Wide Predictions for a Single Species

We also developed an extension of the ngLOC core method called ngLOC-X, which is used to generate predictions for the proteome of a single species. For more information pertaining to the derivations behind ngLOC-X, for the sake of brevity, we ask the reader to refer to our previous work [15].

RESULTS

We use a naïve Bayesian approach to model the density distributions of fixed-length subsequences (i.e. *n*-grams) over five different subcellular locations in Gram-negative bacteria, and four different subcellular locations in Gram-positive bacteria. These distributions are determined from protein sequence data that contain experimentally determined annotations of subcellular localizations.

Determination of Optimal *n*-Gram Size

Our first test consisted of determining an appropriate *n*-gram length for constructing the feature space. In the context of proteins, an *n*-gram is defined as a subsequence of the primary structure of a protein with a fixed length of *n*. The optimal value of *n* chosen will depend on a variety of factors. The size of the training data and the measure of similarity of the data have a clear influence on *n*, because as the percentage of sequence identity observed in the training data increases, the value of *n* required to discriminate between sequences belonging to different classes also increases [15]. Values of *n* that are too large, however, will lack an ability to generalize beyond the training data because *n*-grams observed in the test sequence are unlikely to be observed in the training data. Likewise, a value of *n* that is too short will be unable to effectively learn discriminatory features in the training data. These problems are more pronounced when learning model parameters from highly unbalanced classes. We perform a separate test for each dataset to ensure that we select an optimal value of *n* for each classifier. Table **2** shows the performance results for both datasets.

Table 2. Performance of Different *n*-Gram Models

n-Gram Size	Gram-Negative			Gram-Positive		
	Overall Accuracy	Mac-F1	AUC	Overall Accuracy	Mac-F1	AUC
1	82.5%	0.658	0.673	87.4%	0.715	0.770
2	84.5%	0.715	0.746	88.7%	0.738	0.772
3	88.5%	0.788	0.829	91.1%	0.769	0.807
4	90.3%	0.819	0.882	92.1%	0.694	0.780
5	89.7%	0.795	0.904	90.8%	0.751	0.756
6	89.8%	0.808	0.898	89.3%	0.732	0.895
7	87.3%	0.799	0.862	87.7%	0.706	0.898
8	84.5%	0.771	0.847	86.7%	0.682	0.901

This table shows overall accuracy, macro-averaged F_1, and the area under the ROC curve (AUC) performance metrics over varying *n*-gram lengths on both the Gram-negative and Gram-positive datasets.

Notice that on both datasets, the 4-gram model has the highest overall accuracy. Based on the results for the Gram-negative data, one may consider using either the 4-gram, 5-gram or 6-gram models, because these models have relatively close performance measures. The ROC curve in Fig. (**1**) shows that the 5-gram and 6-gram models offer the most discrimination between true and false predictions. To illustrate, we use the confidence score (CS) output by the ngLOC method to select only those predictions of high confidence. This has the effect of increasing the precision by lowering false positive predictions. If we select a CS threshold (*CSthresh*) that allows only 70% coverage (where 'coverage' is defined as the percentage of the data that is meeting or exceeding the specified *CSthresh*), we obtain excellent macro-averaged precisions of 98.6%, 99.0% and 99.5% for a 4-gram, 5-gram and 6-gram model, respectively. (See Table **S1** in supplementary material.) However, we lose predictions for 30% of the data. A more liberal threshold that allows coverage of 95% of the data results in macro-averaged precisions of 89.9%, 94.9% and 95.5%, respectively, coinciding with the resulting ROC curves. While the curve does suggest that certain *CSthresh* values could be chosen to allow a slightly higher precision on a 5-gram model, the *mac-*F_1 met-

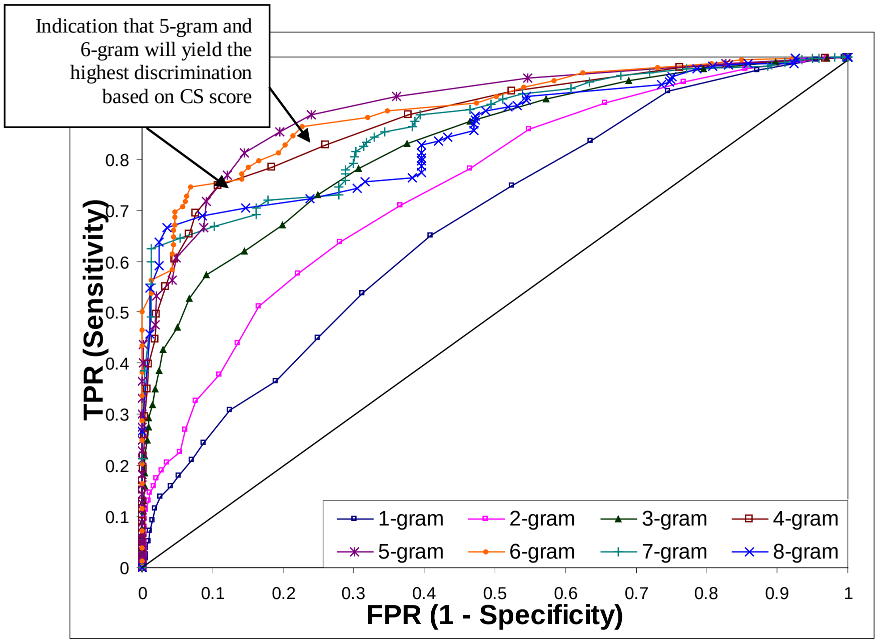

Fig. (1). ROC curves for [1-8]-gram models on Gram-negative data. This figure depicts the ROC curves plotted, where the sensitivity and specificity are based on macro-averaged calculations observed for distinct confidence score (CS) thresholds across each class. We point out that the 5-gram and 6-gram are likely to have the best discriminatory ability based on CS.

ric suggests that the 6-gram model should have slightly better performance (see Table **2**). We are interested in maximizing the number of correct predictions, as opposed to maximizing overall accuracy. Therefore, we chose the 6-gram model for the Gram-negative dataset.

Analyzing the results for the Gram-positive data shows impressive overall performance measures for the 3-gram and 4-gram model, at 91.1% and 92.1%, respectively (see Table **2**). However, it is interesting that the 4-gram model also results in one of the lowest *mac*-F_1 values, at only 0.694. Moreover, if we were to choose either of these models, we would lose significant sensitivity on the smallest classes, compared to higher *n*-gram models. For example, selecting a threshold that allows a generous 90% coverage results in macro-averaged precision values of 86.8% and 71.9% on a 3-gram and 4-gram model, compared with 92.4% and 95.2% on the 5-gram and 6-gram models, respectively (data not shown). The poor performance of the 4-gram model is a result of an absence of predictions for cell wall at this threshold. Moreover, for those predictions achieving a CS below our threshold, only 9% of all sequences localized to the cell wall were predicted correctly with the 4-gram model. We can conclude that the 3-gram and 4-gram models lack an ideal *CSthresh* value that would achieve high coverage while reducing the false positive rate (FPR) across all classes. The ROC curves for the Gram-positive data (see Fig. (**S1**) in the supplementary material) and corresponding AUC (see Table **2**) show that the 6-gram, 7-gram and 8-gram models all perform reasonably well, suggesting that a *CSthresh* value can be chosen that can effectively discriminate between true-positives and false-positives for these models. However, the 7-gram and 8-gram models have comparatively poor *mac*-F_1 values compared to the 6-gram model, suggesting that they will not perform well on the smallest classes. These reasons suggest that the 6-gram model should be the model of choice for the Gram-positive data.

Class-Wise Analysis of the 6-Gram Model

For our next test, we used the 6-gram model to further analyze the performance on individual classes. We ran both datasets separately, with and without specifying a confidence score threshold (*CSthresh*) (see Table **3**). We chose a *CSthresh* that would result in a macro-averaged precision of 98% on the Gram-negative data and 95% on the Gram-positive data. While these are rather restrictive selections, we deemed this to be important in order to use only high-confidence predictions. (We chose a slightly lower threshold for the Gram-positive data because of the extremely small proportion of data localized to the cell wall).

Table **3** shows the effect of selecting a proper CS threshold, which produces a remarkable improvement in the precision measures across all classes. The only drawback to the use of a scoring threshold in this manner is the reduced coverage, i.e., reduction in the number of sequences for which predictions are generated. Positive test sequences with a CS score below the specified CS threshold are considered to be false-negatives, and likewise affect the overall accuracy (reported as micro-averaged sensitivity in Table **3**). This value decreased from 89.8% to 81.6% in the results for Gram-negative data, and from 89.3% to 84.4% for Gram-positive data. Fig. (**2**) shows the ROC curve for each individual class for the Gram-negative data. The bold-line curve is the macro-averaged ROC curve (mac-ROC) over all classes, which is identical to the 6-gram curve in Fig. (**1**). Note that this curve is distinct and distant from the curve for cytoplasm, which is the most prominent class in the data that shows the highest discriminatory performance. This illustrates the importance of using macro-averaged values to indicate performance in the context of unbalanced data. Without averaging, the mac-ROC curve would be very close to the curve for cytoplasmic sequences, giving a false sense that the classifier is near perfect for all classes. We noticed that

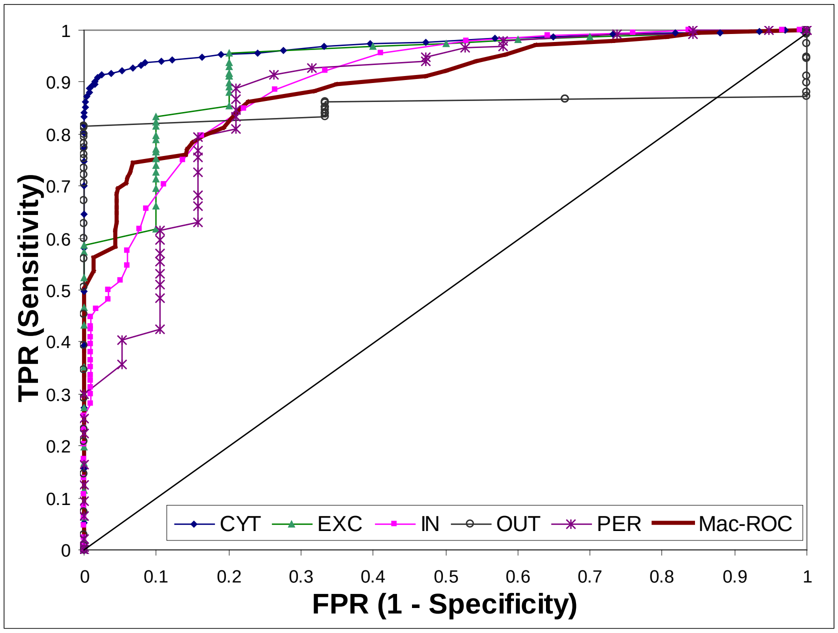

Fig. (2). ROC curves for 6-gram model on Gram-negative data. This figure depicts the ROC curves plotted against a 6-gram model. The Mac-ROC plot is the average ROC curve over each of the individual curves. (CYT = cytoplasmic, EXC = extracellular/secreted, IN = inner/cytoplasmic membrane, OUT = outer membrane, PER = periplasmic, Mac-ROC = macro-averaged ROC curve).

the curve for the outer membrane data looks peculiar; however, we also observed remarkably high precision on these sequences without a CS threshold, at 98.7%. This is an anomaly that may happen when there are very few false predictions generated with relatively higher confidence scores. See the Discussion section for more information pertaining to these and related topics surrounding ROC curves.

Comparison with Other Methods

For comparison purposes, we used a test dataset of Gram-negative bacteria protein sequences that was used by Chou and Shen [8] for comparing their method against PSORTb. This dataset, denoted as *GP-PSORTb*, contained 1114 sequences of which 945 sequences were overlapping with our training dataset for Gram-negative bacteria. The

Table 3. Jack-Knife Performance of ngLOC on Training Data Using a 6-Gram Model

	ngLOC				ngLOC (with CSthresh)			
GN Localization	**Precision**	**Sensitivity**	**Specificity**	**MCC**	**Precision**	**Sensitivity**	**Specificity**	**MCC**
Cytoplasmic	88.7%	99.2%	78.5%	0.82	96.1%	95.6%	89.8%	0.85
Extracellular	94.0%	59.7%	99.8%	0.74	98.6%	53.6%	100.0%	0.72
Inner Membrane	90.8%	83.0%	97.7%	0.83	98.1%	60.1%	99.7%	0.72
Outer Membrane	98.7%	68.0%	100.0%	0.81	99.0%	59.0%	100.0%	0.75
Periplasmic	92.5%	56.1%	99.7%	0.71	98.1%	50.4%	99.9%	0.69
Micro-averaged Results	89.8%	89.8%	97.5%		96.7%	81.6%	99.2%	
Macro-averaged Results	93.0%	73.2%	95.1%		98.1%	63.7%	97.9%	
GP Localization								
Cytoplasmic	88.8%	99.3%	66.8%	0.75	95.9%	96.7%	82.8%	0.81
Extracellular	89.9%	63.7%	99.0%	0.73	97.7%	57.9%	99.8%	0.73
Cell Membrane	94.1%	64.8%	99.3%	0.75	97.1%	49.0%	99.7%	0.66
Cell Wall	78.6%	34.4%	99.9%	0.52	90.9%	31.3%	100.0%	0.53
Micro-averaged Results	89.3%	89.3%	96.4%		96.1%	84.4%	98.7%	
Macro-averaged Results	87.8%	65.6%	91.3%		95.4%	58.7%	95.6%	

This table shows the performance of ngLOC on both the Gram-positive (GP) and Gram-negative (GN) bacterial datasets using a 6-gram model, with and without a CS threshold. We report both micro-averaged and macro-averaged results.

results are reported in Table **4**. The *ngLOC* column indicates the performance of the ngLOC method on the entire *GP-PSORTb* test dataset, and the *ngLOC*(*UO*) column indicates the performance on the 169 non-overlapping (or unobserved in the ngLOC training set) sequences that were unique to the *GP-PSORTb* dataset. We report these results separately to demonstrate that our method also performs well on the portion of the test data that does not overlap with our training data. Specificity is also reported only for the ngLOC method. Finally, not all of the methods that were compared make use of a scoring threshold to improve precision; therefore we report the results without specifying a confidence score threshold.

While the results indicate that all methods outperform PSORTb v1.1, the ngLOC method outperforms the other results presented. (We were unable to obtain results for PSORTb v2.0.) The results presented in the *ngLOC* column represent an unfair comparison due to the substantial number of sequences that exist in both the training data and test data. The *ngLOC*(*UO*) test results offer a more legitimate performance validation because the test eliminates any redundant sequences between the training and testing datasets. Though the overall sensitivity for the non-overlapping sequences (*ngLOC*(*UO*)) drop slightly from 99.2% to 97.0%, these results still indicate that the method is performing remarkably well compared to PSORTb and Gneg-PLoc.

We also use a separate dataset of test sequences, denoted *GP-test*, that was assembled by Chou and Shen [8] for the purpose of independently testing their own method, Gneg-PLoc. This dataset consisted of 637 sequences, of which 512 overlapped with sequences in the ngLOC training data for Gram-negative bacteria. We performed an analysis identical to the previous test and present our results in Supplementary Table **S2**. We used ngLOC to generate predictions for all sequences in the *GP-test* dataset. As with the previous test, we separated out the 125 sequences that did not overlap with any sequence in the ngLOC training data. Again, our method performed extremely well, resulting in an overall accuracy of 99.1% versus 89.1% resulting from the Gneg-PLoc method. While we agree with the fact that these high performance results are partially an artifact of the substantial number of sequences (512) that exist in both the training and test data, we would like to emphasize that our method also shows a 98.4% overall accuracy on the 125 sequences in the *GP-test* dataset that were not in the ngLOC training data (shown under the column ngLOC(UO) in Table **S2**). These results suggest that the ngLOC model has an ability to discriminate between subcellular localizations on bacterial protein sequences, with a predictive performance that surpasses most existing methods today.

Generation of Proteome-Wide Predictions

For our final test, we used the ngLOC method to generate predictions for the entire proteome of ten Gram-negative and five Gram-positive organisms. To generate proteome-wide localization predictions, we trained ngLOC-X [15] using the Gram-negative training data (or Gram-positive, respectively), using a 6-gram model, and a confidence score threshhold (*CSthresh*) value of 25 (30 for Gram-positive) that allowed inclusion of all predictions except those of very low confidence. Both ngLOC and PSORTb v2.0 methods use a scoring threshold to reduce false positives, and therefore predictions are not generated for the entire proteome. There are two reasons why using the CS threshold is important: first, it reduces the chances of generating predictions for sequences targeted to subcellular locations that are not covered by our method. For example, there are other minor subcellular locations to which a protein may be targeted in a Gram-negative cell, such as the flagellum or the fimbrium. Our method does not handle these locations due to the miniscule amount of experimentally determined localization data available for these locations. Second, sequences containing very low homology with respect to any other sequence in the training data will be hard to classify, and will likely result in a low CS. The proteome estimates shown in Tables **5** and **6** are based on the proportion of sequences predicted with a CS of greater than or equal to *CSthresh*.

A large portion of the bacterial proteomes lack experimentally determined localizations, and therefore no real accuracy measure can be calculated at the proteome level. Instead, to establish a measure of credibility of our predictions, we report the fraction of the proteome covered by ngLOC-X and PSORTb v2.0, the fraction of the proteome for which predictions from both methods are available, and the fraction of this overlapping data that have identical predictions. Table **5** shows the results for the Gram-negative organisms, and Table **6** shows the results for the Gram-positive organisms. We observed that ngLOC-X had a substantially larger coverage across all Gram-negative proteomes, with an average of

Table 4. Comparison of ngLOC Against Other Methods Using GP-PSORTb Test Dataset

	Number of Sequences	PSORTb Sensitivity	Gneg-PLoc Sensitivity*	ngLOC Sensitivity	ngLOC Specificity	ngLOC(UO) Sensitivity	ngLOC(UO) Specificity
Cytoplasm	140(39)	83.6%	93.6%	100%	99.6%	100%	96.9%
Extracellular	74(18)	31.1%	90.5%	98.7%	100.0%	94.4%	100.0%
Inner Membrane	687(74)	83.0%	97.5%	99.7%	98.8%	97.3%	98.9%
Outer Membrane	97(21)	82.5%	87.6%	99.0%	99.4%	95.2%	100.0%
Periplasmic	116(17)	81.0%	94.8%	95.7%	100.0%	94.1%	100.0%
Overall Accuracy		79.4%	95.4%	99.2%		97.0%	

This table reports the individual sensitivity measures (TP / (TP + FN)) for each class resulting from the original PSORTb v1.1 method, Gneg-PLoc method, and ngLOC. Specificity is also indicated for the results from the ngLOC method. The *Sequences* column reports the number of sequences in each class. The number in parentheses is the number of sequences that are not in the ngLOC training data. The *ngLOC* column shows the results from the ngLOC method. The *ngLOC*(*UO*) column shows the performance of ngLOC on the 169 unique sequences in the test data that did not exist in the training data. We show the overall accuracy resulting from each method at the bottom of the sensitivity columns.
*(The PSORTb and Gneg-PLoc results are taken from the results reported by Chou and Shen [8]).

74.7% coverage, versus an average of 55.7% coverage by PSORTb. This represents a 34.1% increase in coverage by ngLOC-X, covering an additional 19% of the Gram-negative bacterial proteomes on average. Additionally, our results show that ngLOC-X predictions covered about 79% of the PSORTb predictions, where 82% of these have identical localizations predicted. The results for the Gram-positive bacterial proteomes are similar, with ngLOC-X resulting in an average coverage of 80.6% of the proteome, versus an average coverage of 72.5% by PSORTb, representing an 11.1% increase in coverage by ngLOC-X, covering an additional 8.1% of the proteome. The ngLOC-X predictions covered about 80.7% of the PSORTb predictions, where 70.8% of those have identical localization prediction.

The fractions of subcellular proteomes scaled consistently across most Gram-negative species for both methods (Table **5**). The average subcellular proteome estimates are strikingly similar, with the largest difference observed between the estimates for the outer membrane, with ngLOC-X reporting 1.94% and PSORTb reporting 4.14%, respectively. However, there were significant departures for a few species. The largest anomaly is the ngLOC-X estimate for *E. coli*, with only 48.1% of its proteome estimated to localize into the cytoplasm – a significantly lower estimate than average (63.8%), and also substantially lower than the estimate from PSORTb (60.2%). However, we also observed that 83.2% of the overlapping predictions for this species between these two methods are in agreement. One possible reason for this anomaly is that our training data contains a disproportionately higher number of *E. coli* sequences for non-cytoplasmic locations, resulting in over-prediction of the non-cytoplasmic locations, and vice-versa. The dominance of *E. coli* protein sequences in our training data is likely due to the role of *E. coli* as an established model organism, and to the successful experimental characterization of the subcellular localization of its proteome, compared to any other bacterium. The other significant anomaly was the substantially larger estimate for the outer membrane proteome by PSORTb for *C. crescentus*, *C. jejuni*, and *H. pylori* compared to the ngLOC-X estimate. For the rest of the organelles, both methods were relatively consistent across all locations.

The predicted proportions across different subcellular localizations of Gram-positive bacteria are also highly similar between ngLOC-X and PSORTb, though they are slightly more disproportionate than the Gram-negative predictions (Table **6**). The two primary differences are in the subcellular proteome estimates for cell wall and extracellular space. The ngLOC-X method estimated an average of 4.27% of the proteome to localize to cell wall, versus an average of 1.42% estimated from PSORTb. The estimates for extracellular space were also larger from ngLOC-X, which predicted an average of 9.58% of the proteome to localize here, versus an average of 5.93% estimated from PSORTb. The estimates for the cytoplasm and the cytoplasmic membrane were more proportionate between the methods. The overall estimates for *S. pneumoniae* were strikingly similar between the methods and also offered the most coverage and overlap of predictions between the methods.

DISCUSSION

The ngLOC method uses the naïve Bayes classification method to model the probability that any given protein sequence will be targeted to a specific localization. As with any Bayesian method, the performance of ngLOC is partially dependent on the development of a good estimate for the prior probability distribution of an arbitrary sequence being localized to each localization class. Our method uses the observed distribution in the training data to estimate the parameters of the prior distribution. This accentuates the importance of ensuring that our training data is drawn in a way that is representative of the true underlying distribution of

Table 5. Estimation of the Subcellular Proteome of Ten Gram-Negative Organisms

Species		ngLOC	PSORTb v2.0			ngLOC					PSORTb v2.0				
	Proteome Size	% Coverage	% Coverage	% Overlap	% Agreement	% CYT	% EXC	% IN	% OUT	% PER	% CYT	% EXC	% IN	% OUT	% PER
A. dehalogenans	4346	79.8	57.7	48.3	78.3	76.18	0.46	21.94	0.61	0.81	62.36	0.8	30.3	3.07	3.47
C. crescentus	3737	76.8	53.9	43.1	81.3	71.99	0.63	24.49	0.77	1.08	61.67	0.7	28.95	5.21	3.48
C. jejuni	1628	70.0	55.0	41.6	83.3	62.25	1.23	32.92	1.67	1.93	57.43	0.67	32.18	6.26	3.46
E. coli	5323	71.5	56.3	43.1	83.2	48.11	3.39	35.71	5.59	7.2	60.19	1	29.78	4.2	4.83
G. sulfurreducens	3445	74.9	65.4	50.7	79.0	67.13	0.85	29.19	1.12	1.71	66.06	1.06	28.35	2.13	2.4
H. influenzae	1791	72.6	59.4	46.5	87.1	58.08	1.85	34.85	2.69	2.54	64.85	0.66	27.44	3.01	4.04
H. pylori	1575	67.2	54.6	39.6	84.3	61.38	1.7	33.62	1.23	2.08	61.4	1.4	27.67	8.02	1.51
N. gonorrhoeae	2002	76.0	52.1	42.1	81.4	63.91	1.97	27.55	2.5	1.97	68.55	0.19	25.22	3.45	2.59
R. conorrii	1374	77.2	42.7	34.5	81.2	59.28	1.51	31.1	2.36	1.98	60.48	0.34	34.24	2.73	2.21
T. denitrificans	2827	81.3	59.4	49.6	80.8	69.63	0.48	26.15	0.83	1.83	63.75	0.36	29.35	3.27	3.27
Average		**74.7**	**55.7**	**43.9**	**82**	**63.79**	**1.41**	**29.75**	**1.94**	**2.31**	**62.67**	**0.72**	**29.35**	**4.14**	**3.13**

This table presents the location-wise percentages of the proteome predicted to localize into one of the five distinct organelles for ten Gram negative organisms. The proteomes and corresponding PSORTb v2.0 predictions were downloaded from the PSORTdb online database of subcellular localization predictions for bacteria [18]. We compared our predictions against PSORTb v2.0 for the subset of predictions for which both methods generated predictions. The *% overlap* column indicates the fraction of the proteome in which both methods had a prediction. The *% agreement* column indicates the percent of the overlapping data that had identical predictions from both methods. The ngLOC predictions were generated with a 6-gram model, with a confidence score threshold of 25. CYT, cytoplasm; EXC, extracellular/secreted; IN, inner membrane; OUT, outer membrane; PER, periplasm.

Table 6. Estimation of the Subcellular Proteome of Five Gram-Positive Organisms

		ngLOC	**PSORTb v2.0**			**ngLOC**				**PSORTb v2.0**			
Species	**Proteome Size**	**% Coverage**	**% Coverage**	**% Overlap**	**% Agreement**	**% CYT**	**% EXC**	**% IN**	**% WAL**	**% CYT**	**% EXC**	**% IN**	**% WAL**
M. tuberculosis	4179	77.5	69.3	53.1	71.9	64.29	10.10	21.66	3.95	72.11	7.29	20.28	0.31
S. pneumoniae	2088	86.2	77.8	66.6	69.7	66.72	8.00	23.06	2.22	65.83	7.76	25.25	1.17
B. anthracis	5311	79.3	74.0	58.2	69.1	53.81	10.26	29.46	6.46	62.95	6.36	29.27	1.42
C. acetobutylicum	3848	76.2	70.8	54.7	71.9	60.80	9.34	24.81	5.04	65.31	3.78	29.77	1.14
L. acidophilus	1861	83.8	70.8	59.9	71.7	61.06	10.20	25.08	3.66	62.75	4.48	29.74	3.03
Average		80.6	72.5	58.5	70.8	61.34	9.58	24.81	4.27	65.79	5.93	26.86	1.42

This table presents the location-wise percentages of the proteome predicted to localize into one of the four distinct organelles for five Gram-positive organisms. The proteomes and corresponding PSORTb v2.0 predictions were downloaded from the PSORTdb online database of subcellular localization predictions for bacteria [18]. We compared our predictions against PSORTb v2.0 for the subset of predictions for which both methods generated predictions. The *% overlap* column indicates the fraction of the proteome in which both methods had a prediction. The *% agreement* column indicates the percent of the overlapping data that had identical predictions from both methods. The ngLOC predictions were generated with a 6-gram model, with a confidence score threshold of 30. CYT, cytoplasm; EXC, extracellular/secreted; IN, inner membrane; WAL, cell wall.

the data. Our training data were derived from the subset of sequences in the Swiss-Prot repository that have experimentally determined localizations annotated for bacterial sequences. Observing the class-wise distribution of our training data in Table **1** reveals the highly unbalanced nature of our data. This is expected, because the true distribution of proteins localized to each subcellular location is also unbalanced. Yet, we noticed that the data used by the PSORTb method are balanced [4, 5], which is not representative of the natural, underlying distribution across subcellular classes. As shown in Table **5** and Table **6**, the prediction coverage of the proteome by ngLOC-X for all species was significantly larger than that by PSORTb. We attribute these successful results to using a robust training dataset with a location-wise distribution that is closely representative of the true underlying proportions in a typical bacterial cell.

One significant benefit of the ngLOC method is its foundation on a probabilistic model. We believe that there is a significant lack of methods that generate probabilities or confidence scores with each prediction. It is valuable to know "how true" or "how false" a true or false prediction may be. Although the prediction is based on the model with the highest probability, the probability can also be used as a comparative measure against other classes. Our method and the PSORTb method are among the very few methods available that generate a score associated with each prediction. By using a scoring threshold, the false positive rate can be significantly contained, thereby improving the precision of the classifier (Table **3**).

We demonstrated how a ROC curve could be used as a visual graphical indicator to measure the ability for a classifier such as ngLOC to distinguish between correct and incorrect predictions. It is important to note that a ROC curve does not say anything directly about the overall accuracy of the classifier – this information is inferred. At face value, a ROC curve shows the ability of a classifier to rank positive instances relative to negative instances in the dataset according to a score generated for each instance [25]. It is typically plotted only for binary classification problems, whereas in our classification task, we are working with 4 or 5 distinct classes, depending on the classification problem being addressed. We generate a single ROC curve, denoted as *Mac-ROC*, by macro-averaging the fraction of true positives and false positives over each class observed at distinct CS thresholds (Fig. **2**). We have not observed any other methods that have presented a multi-class ROC curve and analysis in this manner.

For the comparative test against other methods, the initial aim was to compare the ngLOC results with those of the PSORTb v2.0 method, a popular method for bacterial protein subcellular localization prediction [5]. However, numerous high-confidence predictions were observed from ngLOC that disagreed with the annotation in the PSORTb training data. The ngLOC predictions for these proteins were consistent with the annotation in Swiss-Prot. This problem was particularly noticeable with multi-localized proteins, which is partly due to the way that multi-localized proteins are defined in the literature. Although the training data used by PSORTb was originally taken from the Swiss-Prot database, their data was further subject to manual verification based on literature review [4]. This process resulted in discrepancies in the interpretation of multi-localized proteins. For instance, the curators of the PSORTb dataset consider some membrane proteins to be multi-localized if a significant part of a protein (i.e. a domain) lies outside of the membrane [*F. S. Brinkman, Personal communication*]. In contrast, the curators of Swiss-Prot database, which is the basis for our training data, annotate such proteins as ‘integral membrane proteins,’ since they are targeted to membranes. We concur with the annotation as defined by Swiss-Prot because we intend to use and predict localization information at the protein level, not at the domain level (as interpreted by PSORTb). Similar observations have been fully documented by Chou and Shen [8] and thus we do not delve into the topic here. However, due to the numerous inconsistencies in the annotation between our data and that of PSORTb, we determined that a comparative test could not be performed on the PSORTb dataset. The ngLOC method has the ability to use multi-localized proteins for training and generating multi-localized predictions [15]. Nevertheless, there are far fewer bacterial proteins in Swiss-Prot annotated as multi-localized, thus limiting their use. Therefore, we did not attempt to generate any multi-class predictions in this study.

We showed a significant overlap in the proteome-wide predictions that were generated by our method and the PSORTb method. However, we found several low-confidence predictions generated by ngLOC-X for which the PSORTb method generated a high confidence prediction, and vice versa. This demonstrates the value of using such methods in tandem, in conjunction with experimental methods, in order to reduce the likelihood of false positives when understanding the localization of proteins. However, we also stress that functional attributes of protein sequences are annotated in a dynamic process; thus, some disagreements between the methods were likely due to differences in the time frame that the training data was acquired and in the methods used to assemble the training data.

CONCLUSION

Computational methods for predicting protein subcellular localization are vital for functional annotation of proteomes *en masse*. In our study, we curated a new dataset with subcellular annotation for Gram-negative and Gram-positive sequences. Using this new dataset, we extended the ngLOC method [15], a Bayesian classifier that can predict the subcellular localization of a protein. We demonstrated the performance of the ngLOC method through leave-one-out cross validation and direct comparison with similar methods, further validating its usefulness and superior performance. This method shows an overall accuracy of 89.7% and 89.3% for Gram-negative and Gram-positive bacteria protein sequences, respectively. As a probabilistic method, the method can generate a confidence score (CS) that places a measure of credibility on each prediction. We showed how this measure can be used to improve the precision of the method, as well as be used to generate a receiver operating characteristic (ROC) curve for individual classes and a single macro-averaged ROC curve covering all classes. We discussed the importance of micro-averaging and macro-averaging performance results in the context of unbalanced datasets such as those used in this study. We used this method to annotate the subcellular proteomes of ten Gram-negative and five Gram-positive species that resulted in an impressive coverage of 74.7% and 80.6% of the proteomes in these species, respectively. To our knowledge, this study is the first to offer such a wide coverage for the estimation of bacterial subcellular proteomes.

ABBREVIATIONS

ROC	=	Receiver Operating Characteristic
AUC	=	Area Under the ROC Curve
CS	=	Confidence Score
SVM	=	Support Vector Machine

ACKNOWLEDGEMENTS

This work has been supported by the startup funds to CG from the State University of New York (SUNY) at Albany and the graduate student fellowship provided by the GenNYsis Center for Excellence in Cancer Genomics, SUNY at Albany.

SUPPLEMENTARY MATERIAL

This article is accompanied by 3 supplementary files and it can be viewed at www.bentham.org/open/toaij

REFERENCES

[1] I.B. Holland, "Translocation of bacterial proteins - an overview", *Biochem. Biophys. Acta,* vol. 1694, pp. 5-16, 2004.

[2] J.L. Gardy and F. S. L. Brinkman, "Methods for predicting bacterial protein subcellular localization", *Nat. Rev. Microbiol.,* vol. 4, pp. 741-751, 2006.

[3] K. Nakai, and P. Horton, "PSORT: a program for detecting sorting signals in proteins and predicting their subcellular localization", *Trends Biochem. Sci.,* vol. 24, pp. 34-6, Jan 1999.

[4] J.L. Gardy, C. Spencer, K. Wang, M. Ester, G. E. Tusnady, I. Simon, S. Hua, K. deFays, C. Lambert, K. Nakai, and F.S. Brinkman, "PSORT-B: Improving protein subcellular localization prediction for Gram-negative bacteria", *Nucleic Acid Res.,* vol. 31, pp. 3613-7, Jul, 2003.

[5] J.L. Gardy, M.R. Laird, F. Chen, S. Rey, C.J. Walsh, M. Ester, and F.S. Brinkman, "PSORTb v.2.0: expanded prediction of bacterial protein subcellular localization and insights gained from comparative proteome analysis", *Bioinformatics,* vol. 21, pp. 617-23, March, 2005.

[6] M. Bhasin, A. Garg, and G.P. Raghava, "PSLpred: prediction of subcellular localization of bacterial proteins", *Bioinformatics,* vol. 21, pp. 2522-4, May 15 2005.

[7] C.S. Yu, C.J. Lin, and J.K. Hwang, "Predicting subcellular localization of proteins for Gram-negative bacteria by support vector machines based on n-peptide compositions", *Protein Sci.,* vol. 13, pp. 1402-6, May 2004.

[8] K.C. Chou, and H.B. Shen, "Large-scale predictions of gram-negative bacterial protein subcellular locations", *J. Proteome. Res.,* vol. 5, pp. 3420-3428, 2006.

[9] H.-B. Shen, and K.-C. Chou, "Gpos-PLoc: an ensemble classifier for predicting subcellular localization of Gram-positive bacterial proteins", *Protein Eng., Des. Sel..,* vol. 20, pp. 39-46, January 1, 2007 2007.

[10] Z. Lu, D. Szafron, R. Greiner, P. Lu, D.S. Wishart, B. Poulin, J. Anvik, C. Macdonell, and R. Eisner, "Predicting subcellular localization of proteins using machine-learned classifiers", *Bioinformatics,* vol. 20, pp. 547-556, March, 2004..

[11] P. Donnes, and A. Hoglund, "Predicting protein subcellular localization: past, present, and future", *Genomic Proteomics Bioinformatics,* vol. 2, pp. 209-15, Nov, 2004.

[12] S.F. Altschul, W. Gish, W. Miller, E.W. Myers, and D.J. Lipman, "Basic local alignment search tool", *J. Mol. Biol.,* vol. 215, pp. 403-10, Oct, 1990.

[13] B. Boeckmann, A. Bairoch, R. Apweiler, M.C. Blatter, A. Estreicher, E. Gasteiger, M.J. Martin, K. Michoud, C. O'Donovan, I. Phan, S. Pilbout, and M. Schneider, "The SWISS-PROT protein knowledgebase and its supplement TrEMBL", *Nucleic Acids Res.,* vol. 31, pp. 365-70, Jan, 2003.

[14] J. Sprenger, J.L. Fink, and R.D. Teasdale, "Evaluation and comparison of mammalian subcellular localization prediction methods", *BMC Bioinformatics,* vol. 7(Suppl 5), p. S3, 2006.

[15] B.R. King, and C. Guda, "ngLOC: an n-gram-based Bayesian method for estimating the subcellular proteomes of eukaryotes", *Genome Biol.,* vol. 8, p. R68, May, 2007.

[16] "The UniProtKB/Swiss-Prot Protein Knowledgebase, URL: http://www.ebi.ac.uk/uniprot/."

[17] W. Li, and A. Godzik, "Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences", *Bioinformatics,* vol. 22, pp. 1658-1659, July, 2006.

[18] S. Rey, M. Acab, J.L. Gardy, M.R. Laird, K. deFays, C. Lambert, and F.S. Brinkman, "PSORTdb: a protein subcellular localization database for bacteria", *Nucleic Acids Res.,* vol. 33, pp. D164-8, Jan, 2005.

[19] A.K. McCallum, and K. Nigam, "A comparison of event models for naive bayes text classification", AAAI-98 Workshop on Learning for Text Categorization, 1998, vol. 752.

[20] B.W. Matthews, "Comparison of the predicted and observed secondary structure of T4 phage lysozyme", *Biochim. Biophys. Acta,* vol. 405, pp. 442-51, Oct, 1975.

[21] Y. Yang, and X. Liu, "A re-examination of text categorization methods", in 22nd annual international ACM SIGIR conference on Research and development in information retrieval, Berkeley, CA, 1999, pp. 42-49.

[22] C. van Rijsbergen, *Information Retrieval*. London: Butterworths, 1979.

[23] F. Provost, T. Fawcett, and R. Kohavi, “The case against accuracy estimation for comparing induction algorithms”, in Proceedings of the Fifteenth International Conference on Machine Learning, San Francisco, CA, 1998, pp. 445-453.

[24] C.X. Ling, J. Huang, and H. Zhang, “AUC: A Better Measure than Accuracy in Comparing Learning Algorithms”, in Advances in Artificial Intelligence: 16th Conference of the Canadian Society for Computational Studies of Intelligence, AI 2003, Halifax, Canada, 2003, pp. 991-991.

[25] T. Fawcett, “An introduction to ROC analysis”, *Pattern Recognit. Lett.*, vol. 27, pp. 861-874, 2006.

Received: September 23, 2008 Revised: November 24, 2008 Accepted: December 22, 2008