Document Classification Approach Leads to a Simple, Accurate, Interpretable G Protein Coupled Receptor Classifier

Betty Yee Man Cheng YMCHENG@CS.CMU.EDU

Jaime G. Carbonell JGC@CS.CMU.EDU

Judith Klein-Seetharaman JUDITHKS@CS.CMU.EDU

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213, USA

Abstract[1]

The need for accurate, automated protein classification methods continues to increase as advances in biotechnology uncovers new proteins at a fast rate. G-protein coupled receptors (GPCRs) are a particularly difficult superfamily of proteins to classify due to the extreme diversity among its members; yet, they are an important subject in pharmacological research being the target of approximately 60% of current drugs (Muller, 2000). A comparison of BLAST, k-NN, HMM and SVM with alignment-based features by Karchin *et al.* (2002) has suggested that classifiers at the complexity of SVM are needed to attain high accuracy in GPCR subfamily classification. Here, analogous to document classification, we applied Decision Tree and Naïve Bayes classifiers with chi-square feature selection on n-gram counts to the GPCR family and subfamily classification task. Using the dataset and evaluation protocol from the previous study, we found the Naïve Bayes classifier surpassing the reported accuracy of SVM by 4.8% and 6.1% in level I and II subfamily classification with an accuracy of 93.2% and 92.4% respectively. The Decision Tree, while inferior to SVM, still outperforms HMM in both level I and II subfamily classification. Moreover, the n-grams selected by chi-square feature selection show evidence of biological importance. Thus, the document classification approach has resulted in a simpler, more accurate and interpretable classifier.

Keywords: GPCR, protein family classification, chi-square, Naïve Bayes, decision tree

1 Introduction

Advances in biotechnology have drastically increased the rate at which new proteins are being uncovered, creating a need for automated methods of classification of proteins. A variety of computational methods have been developed for this task. These methods can be divided into five categories: based on sequence alignments (categories 1-3, Table 1), based on motifs (category 4) and based on machine learning approaches that do not center as much on alignment (category 5, Table 3).

[1] Much of the material in this paper has since been published in PROTEINS: Structure, Function and Bioinformatics journal.

1.1 Classification of Proteins

The first category of methods (Table 1A) searches a database of known sequences for one most similar to the query sequence, and assigns the classification of the best-scoring known sequence to the query sequence. The similarity search is done by performing a pair-wise sequence alignment between the query sequence and every sequence in the database, with the help of a contingency matrix called a similarity matrix quantifying how similar two sequences of amino acids are. The Smith-Waterman (1981) and Needleman-Wunsch (1970) algorithms make use of dynamic programming to perform the alignment and are guaranteed to find the optimal local and global alignment respectively. However, they are extremely slow and thus impossible to use in a database-wide search. A number of heuristic algorithms have been developed, of which BLAST (Altschul *et al.*, 1990) is the most prevalent. Often, a heuristic algorithm would be used to do an initial search of the database for sequences with a similarity scores above a specified threshold, and then either the Smith-Waterman or Needleman-Wunsch algorithm would be used to search among the much smaller set of similar sequences. The second category of methods also searches against a database of known sequences, but instead of comparing the query sequence against the known sequences directly, these methods first align multiple sequences from the same protein superfamily, family or subfamily, and create a consensus sequence to represent the particular category. Then, the query sequence is compared against each of the consensus sequences using a pair-wise sequence alignment tool and is assigned the protein superfamily, family or subfamily represented by the consensus sequence with the highest similarity score. Some of these methods are listed in Table 1B. The third category of methods uses profile HMM as an alternative to consensus sequences, but is otherwise identical to the second category of methods. Representative methods are listed in Table 1C. There are a number of databases of profile HMM available on the Internet (Table 2).

The fourth category of methods to protein classification searches for the presence of known motifs in the query sequence from a database. Motifs are short amino acid sequence patterns that capture the conserved regions of a protein superfamily, family or subfamily. They are often the binding site of a protein-protein interaction specific to the protein category, which is the reason for their being conserved through evolution. They are written as regular expressions because biochemically similar amino acids can act as substitutes for one another in these motifs. Motifs are often captured by multiple sequence alignment tools, but some pattern detection methods have been attempted as well (Smith *et al.*, 1990; Neuwald and Green, 1994). These four categories of classification methods are preferred by biologists because they are "white-box" classifiers, giving an indication of why certain sequences are related and providing clues to the cause of those proteins' functions. Table 1 lists some of the alignment tools, profile HMM tools, and databases used in these popular "white-box" classifiers.

A. Pair-wise Sequence Alignment Tools	
Tool	**Reference**
BLAST	Altschul *et al.*, 1990
FASTA	Pearson, 2000
ISS	Park *et al.*, 1997
Needleman-Wunsch	Needleman and Wunsch, 1970
PHI-BLAST	Zhang *et al.*, 1998
PSI-BLAST	Altschul *et al.*, 1997
Smith-Waterman	Smith and Waterman, 1981
B. Multiple Sequence Alignment Tools	
Tool	**Reference**
BLOCKMAKER	Henikoff *et al.*, 1995
ClustalW	Thompson *et al.*, 1994

DIALIGN	Morgenstern *et al.*, 1998; Morgenstern, 1999
MACAW	Schuler *et al.*, 1991
MULTAL	Taylor, 1988
MULTALIGN	Barton and Sternberg, 1987
Pileup	Wisconsin Package, v. 10.3
SAGA	Notredame *et al.*, 1996.
T-Coffee	Notredame *et al.*, 2000
C. Profile HMM Tools	
Tool	**Reference**
GENEWISE	GENEWISE, 2002
HMMER	HMMER, 2003
HMMpro	HMMpro, v. 2.2
META-MEME	Grundy *et al.*, 1997
PFTOOLS	Bucher *et al.*, 1996
PROBE	Neuwald *et al.*, 1997
SAM	Krogh *et al.*, 1994

Table 1. List of tools used in common protein classification methods.

Motifs / Profiles Databases	
Database	**Reference**
BLOCKS+	Henikoff *et al.*, 1999; Henikoff *et al.*, 2000
eMOTIF	Huang and Brutlag, 2001
Pfam	Bateman *et al.*, 2002
PRINTS	Attwood *et al.*, 2002
ProDom	Servant *et al.*, 2002; Corpet *et al.*, 2000
PROSITE	Sigrist *et al.*, 2002
SMART	Ponting *et al.*, 1999; Letunic *et al.*, 2002
Superfamily	Gough *et al.*, 2001.
SWISSPROT	Apweiler *et al.*, 1997; Boeckmann *et al.*, 2003

Table 2. List of databases providing protein family classification information. A list with links to the web-addresses can be found at the authors' project website flan.blm.cs.cmu.edu.

One commonality among the first four categories of classification methods is the use of alignment. However, there are inherent limitations with using sequence alignment in classification — sequence alignment assumes contiguity is conserved between homologous segments in the protein sequence (Vinga and Almeida, 2003). This contradicts with the genetic recombination and re-shuffling that occur in evolution (Lynch, 2002; Zhang *et al.*, 2002). As a result, when sequence similarity is low, aligned sequence segments are often short and due to chance occurrence. In particular, sequence alignments become unreliable when the sequences have less than 40% similarity (Wu *et al.*, 2003), and unusable below 20% similarity (Pearson, 1996; Pearson, 1998). This has sparked interest in classification methods that do not make use of alignments – the fifth category of methods. The majority of this work has occurred in the past two decades with most reports published in the past 5 years. Two main directions have evolved; methods based on "word" frequency, and methods that do not require transforming the sequence into fixed length word segments (Vinga and Almeida, 2003). The first direction makes use of various machine-learning algorithms (Baldi and Brunak, 2001). Popular tools include Markov models and k-nearest neighbors (k-NN) classifiers, but Deshpande and Karypis (2002) showed that SVM-based approaches can attain a higher accuracy than those classifiers in protein classification. Neural networks, clustering, and approaches that make use of information-theory

based measures instead of statistical distances between frequency vectors have also been applied (Vinga and Almeida, 2003). Table 3 shows some of the work in this area. The second direction in this category makes use of Kolmogorov complexity and Chaos Theory (Vinga and Almeida, 2003). Compared to the first, these classification schemes are much more recent and just beginning to be explored.

Classifier	**Features**	**Reference**
Bayesian inference using Gibbs sampling	Number of conserved columns, the size and number of classes and the motifs in them	Qu *et al.*, 1998
Bayesian Neural Networks	Bi-gram counts, presence and significance of motifs found using an automated tool *Sdiscover*	Wang *et al.*, 2000
Clustering	Digraph representation of the sequence space where the weight of each edge between 2 sequences is the similarity score of the sequences from Smith-Waterman, BLAST and FASTA	Yona *et al.*, 1999
	Sequence and topological similarity	Mitsuke *et al.*, 2002
Discriminant function analysis (non-parametric, linear)	Frequency of each amino acid, average periodicity of GES hydropathy scale and polarity scale, variance of first derivative of polarity scale	Kim *et al.*, 2000
Neural Networks	N-gram counts with SVD	Wu *et al.*, 1995
	Matrix patterns derived from bi-grams	Ferran & Ferrara, 1992
Sparse Markov Transducers	All subsequences of the protein inside a sliding window	Eskin *et al.*, 2000
Support Vector Machines	Fisher scores with Fisher kernel	Jaakkola *et al.*, 1999 & 2000; Karchin *et al.*, 2002
	Set of all possible *k*-grams (fixed *k*) with Spectrum kernel and Mismatch kernels	Leslie *et al.*, 2002a; Leslie *et al.*, 2002b
	String subsequence kernel	Vanschoenwinkel *et al.*, 2002

Table 3. List of some machine learning approaches on protein classification.

In summary, there is a belief that the simpler, yet interpretable classifiers based on sequence alignments are inherently limited in performance due to the failure in accurate alignment of sequences with low sequence identity and that better classification accuracy can be achieved by applying more complex classifiers on features that may or may not be derived from sequence alignments (Deshpande and Karypis, 2002; Karchin *et al.*, 2002). However, the latter methods generally trade off interpretability of the results for their high accuracy. While classifiers at the higher end of complexity are currently being explored by several groups (see Table 3 for some examples), the classifiers at the lower end have been neglected. In particular, the simplest classifier that has been attempted on protein classification is the k-NN classifier, but there are several other even simpler classifiers that are particularly popular for text classification which – to the best of our knowledge – have not been tried previously on the protein classification task. Here, we describe the application of two of the simplest classifiers, Naïve Bayes and Decision Trees to the protein family classification task.

1.2 G-Protein Coupled Receptors

With the enormous amount of proteomic data now available, there are a large number of datasets which can be used in the protein family classification task. We have chosen the G-protein coupled receptor (GPCR) family in our experiments because they are an important topic in pharmacology research and they present one of the most challenging datasets for protein classification. GPCRs are the largest superfamily of proteins found in the body (Gether, 2000) and function in mediating the responses of cells to various environmental stimuli including hormones, neurotransmitters and odorants, to name just a few of the chemically diverse ligands that GPCRs respond to. As a result, they are the target of approximately 60% of approved drugs currently on the market (Muller, 2000). Reflecting the diversity in ligands, the GPCR superfamily is also one of the most diverse protein families (Moriyama and Kim, 2003). Sharing no overall sequence homology (Kolakowski, 1994), the only feature common to all GPCRs is their seven transmembrane alpha helices separated by alternating extracellular and intracellular loops, with the amino terminus (N-terminus) on the extracellular side and the carboxyl terminus (C-terminus) on the intracellular side. The GPCR protein superfamily is composed of five major families (classes A through E) and several putative and "orphan" families (Horn *et al.*, 1998). Each family is divided into level I subfamilies and then further into level II subfamilies based on pharmacological and sequence identity considerations. The extreme divergence among GPCR sequences is the primary reason for the difficulty in classifying them and the diversity has prevented further classification of a number of known GPCR sequences at the family and subfamily levels — these sequences are designated as "orphan" or "putative/unclassified" GPCRs (Moriyama and Kim, 2003). Moreover, since subfamily classifications are often defined chemically/pharmacologically rather than by sequence homology, many subfamilies share strong sequence homology with other subfamilies, making subfamily classification extremely difficult (Karchin *et al.*, 2002).

1.3 Classification of G-Protein Coupled Receptor Sequences

A number of classification methods have been studied on the GPCR dataset. Lapinsh *et al.* (2002) extracted physical properties of amino acids and used multivariate statistical methods, specifically principal component analysis (PCA), partial least squares (PLS), autocross-covariance transformations (ACC's) and z-scores, to classify GPCR proteins at the level I subfamily level. Levchenko (2001) used hierarchical clustering on similarity scores computed with the SSEARCH program[2] to classify GPCR sequences in the human genome belonging to the peptide level I subfamily into their level II subfamilies. Liu and Califano (2001) used unsupervised, top-down clustering in conjunction with a pattern-discovery algorithm, a statistical framework for pattern analysis, and hidden Markov models to produce a hierarchical decomposition of GPCRs down to the subfamily levels. A systematic comparison of performance of different classifiers ranging in complexity has been carried out recently by Karchin *et al.* (2002) for GPCR classification at the superfamily level (that is, whether or not a given protein is a GPCR) and level I and II subfamily levels. Note that family-level classification was not examined by this study. The methods tested include a simple nearest neighbor approach (BLAST), a method based on multiple sequence alignment generated by a statistical profile hidden Markov model (HMM), a nearest neighbor approach with protein sequences encoded into Fisher Score Vector space (kernNN), and support vector machines. In the HMM method, a model is built for each class in the classification and a query sequence is assigned to the class whose model has the highest probability of generating the sequence. Two implementations of

[2] The SSEARCH program is a rigorous and computationally expensive program that searches for similarity between a query sequence and a group of sequences. It uses William Pearson's implementation of the Smith and Waterman algorithm. Compared to the more popular similarity search programs, BLAST and FASTA, it can be very slow. (SSearch, 2002)

support vector machines were investigated, SVM and SVMtree, where the latter is a faster approximation to a multi-class SVM. Fisher Score Vectors were also used with SVM and SVMtree. To derive the vectors, Karchin *et al.* built a profile HMM model for a group of proteins and then computed the gradient of the log likelihood that the query sequence was generated by the model. A feature reduction technique based on a set of pre-calculated amino acid distributions was used to reduce the number of features from 20 components per matching state in the HMM to 9 components per matching state. Both SVM and the kernNN method made use of a radial basis kernel functions. The results from this study are reproduced in Table 4. The study concluded that while simpler classifiers (specifically HMM) perform better at the superfamily level, the computational complexity of SVM is needed to attain "annotation-quality classification" at the subfamily levels. However, the simplest classifiers, such as Decision Trees and Naïve Bayes, have not been applied. In this study, we investigated in further detail the performance of simple classifiers in the task of GPCR classification at the family and subfamily levels. We first optimized these simple classifiers using feature selection and then compared our results against those reported in the study by Karchin *et al.* (2002). To our surprise, using only a simple classifier on counts of n-grams in conjunction with a straight-forward feature-selection algorithm, chi-square, was sufficient to outperform all of the classifiers investigated by Karchin *et al.* (2002).

Superfamily Classification	
Method	**Accuracy at the MEP (%)**
SAM-T99 HMM	99.96
SVM	99.78
FPS BLAST	93.18
Level I Subfamily Classification	
Method	**Accuracy at the MEP (%)**
SVM	88.4
BLAST	83.3
SAM-T2K HMM	69.9
kernNN	64.0
Level II Subfamily Classification	
Method	**Accuracy at the MEP (%)**
SVM	86.3
SVMtree	82.9
BLAST	74.5
SAM-T2K HMM	70.0
kernNN	51.0

Table 4. Classification results reported in a previous study on complexity needed for the GPCR classification task (Karchin *et al.*, 2002). Karchin *et al.* reported their results in terms of "average errors per sequence" at the minimum error point (MEP). Through e-mail correspondence with the first author, we verified that "average errors per sequence" is equivalent to the error rate. Thus, the accuracy results shown above are converted from those in their paper by the formula "1 – average errors per sequence".

2 Approach

In this section, we will describe the classifiers we used and the method in which we extracted and selected the features for our classifiers using chi-square. Section 4 will explain the datasets used in our study, while section 5 will present our results.

2.1 Decision Tree

Decision tree is one of the simplest classifiers in machine learning. One of its advantages lies in its ease of interpretation as to which are the most distinguishing features in a classification problem. It has been used previously with biological sequence data in classifying gene sequences (Yuan *et al.*, 2003). We used C4.5 implementation of decision tree by J. R. Quinlan (C4.5, release 8). The features given to the classifier were counts of n-grams extracted from the amino acid sequences. Instead of using only n-grams of a single fixed length n, we used n-grams of length 1, 2 … n. The features were declared as continuous attributes rather than discrete attributes to the decision tree classifier. Although we examined the effect of different confidence level in pruning the decision tree, because the difference in accuracy is 1.2% at most, we remained largely with the default 75% confidence level in our experiments and present only those results here.

2.2 Naïve Bayes

Naïve Bayes is another example of a simple classifier in machine learning. Its naïve assumption that all of its features are independent clearly did not hold when we allowed overlaps in extracting n-grams of length greater than 1 from a sequence. Nonetheless, the classifier worked remarkably well in our task as described below.

We used the Rainbow implementation of the Naïve Bayes classifier by Andrew K. McCallum[3]. The features given to the classifier were counts of n-grams extracted from the protein amino acid sequence. Because Rainbow has been originally developed for document classification applications, the software expects its training and testing instances to be documents of words from which it can count the number of occurrences of each word. To transform a protein sequence into a document, we explicitly stated in the document all of the occurring n-grams of the desired sizes. For instance, for the sequence "ACWQRACW" and n-grams of size 2 and 3, the corresponding document would be "AC CW ACW WQ CWQ QR WQR RA QRA AC RAC CW ACW". Rainbow as a document classification tool also excludes all words of length 1, which in our case, are unigrams of amino acids from the protein sequence. Since turning off the stop-list feature and any tokenization procedures in Rainbow did not change this result, we used only n-grams of length 2, 3 … n in our experiments with the Naïve Bayes classifier. The n-gram counts were treated as multinomial attributes in the classifier without any normalization. Laplace smoothing was used.

2.3 Chi-Square

Most machine-learning algorithms do not scale well to high-dimensional feature spaces (Sebastiani, 1999), and the decision tree and Naïve Bayes classifiers are no exceptions. Thus, it is desirable to reduce the dimension of the feature space without sacrificing classification accuracy by removing non-informative or redundant features (Yang and Pedersen, 1997). A large number of feature selection methods have been developed for this task, including document frequency, information gain, mutual information, chi-square, and term strength. We have chosen to use chi-square in our study because it is one of most effective feature selection methods in document classification (Yang and Pedersen, 1997).

The chi-square statistic measures the lack of independence between a given binary feature x and a classification category c by computing the difference between the "expected" number of objects in c with that feature and the observed number of objects in c actually having that feature. By "expected", we mean that if the feature were not dependent on the category and had a uniform

[3] The Rainbow implementation is part of Bow, a toolkit for statistical language modeling, text retrieval, classification and clustering (Bow, 2002).

distribution over all the categories instead, how many instances of *c* would we find with feature *x*. Thus, the formula for the chi-square statistic for each feature *x* is as follows,

$$\chi^2(x) = \sum_{c \in C} \frac{(\mathrm{e}(c,x) - \mathrm{o}(c,x))^2}{\mathrm{e}(c,x)}$$

where *C* is the set of all categories in our classification task, and *e(c,x)* and *o(c,x)* are the "expected" and observed number of instances in category *c* with feature *x* respectively. The "expected" number *e(c,x)* is computed as

$$\mathrm{e}(c,x) = n_c \times \frac{t_x}{N}$$

where n_c is the number of objects in category *c*, *N* is the total number of objects, and t_x is the number of objects with feature *x*.

To obtain binary features from counts of n-grams, we divided each n-gram feature into multiple binary features by considering whether the n-gram has occurred at least *i* times in the sequence, where *i* is the first 20 multiples of 5 (that is, 5, 10, 15 … 100) for unigrams and the first 20 multiples of 1 (that is, 1, 2, 3 … 20) for all other n-grams. Then, we computed the chi-square statistic for each of these binary features. For instance, for the tri-gram DRY, we computed the chi-square statistic for DRY occurring at least 1, 2, 3 … 20 times. The expected number of protein sequences in class *c* having at least *i* occurrences of the n-gram DRY is the product of the number of sequences in class *c* and the ratio of the number of sequences with at least *i* occurrences of the n-gram DRY to the total number of sequences. The chi-square statistic for DRY occurring at least *i* times is the square of the difference between the expected and observed number of sequences in each class having at least *i* occurrences of DRY, normalized by the expected number and summed over all classes.

Next, for each n-gram *j*, we found the value i_{max} such that the binary feature of having at least i_{max} occurrences of n-gram *j* has the highest chi-square statistic out of the 20 binary features associated with n-gram *j*. The n-grams were then sorted in decreasing order according to the chi-square value at their associated binary feature of having at least i_{max} occurrences of the n-gram. The top *K* n-grams were selected as input to our classifiers, where *K* is a parameter that can be tuned to achieve maximum accuracy. Our results showed that the accuracy increases as *K* increases until a maximum is achieved, after which the accuracy slowly decreases as *K* continues to increase. In our study, we also investigated the effect on accuracy from giving the classifier the binary feature of having at least i_{max} occurrences of each selected n-gram *j* versus giving the classifier the count of n-gram *j* as mentioned in sections 4.2 and 4.3.

3 Datasets & Evaluation

In this study, we examined classification of GPCRs at the family level and level I and II subfamily levels. Family-level classification was used to develop our classification protocol, while the subfamily-levels classification were used to compare the performance of our protocol against those classifiers, particularly SVM, studied by Karchin *et al.* (2002).

3.1 Family-Level Classification

In family-level classification, we made use of all GPCR sequences and bacteriorhodopsin sequences with SWISS-PROT entries found in the September 15, 2002 release of GPCRDB (Horn *et al.*, 1998). GPCRDB is an information system specifically for GPCRs, containing all

known GPCR sequences, classification information, mutation data, snake-plots, links to various tools for GPCRs, and other GPCR-related information. It contains both sequences with SWISS-PROT entries and those with TREMBL entries. These entries contain important information such as the protein's classification, function and domain structure. SWISS-PROT entries are computer-generated annotations that have been reviewed by a human, while TREMBL entries have not yet been reviewed. For this reason, we have chosen to use only those sequences with SWISS-PROT entries in our evaluation.

According to GPCRDB, the GPCR superfamily is divided into 12 major and putative families. Bacteriorhodopsin is a non-GPCR family of proteins that are often used as a structural template for the three-dimensional structure of GPCRs (Pardo *et al.*, 1992). Thus, we have decided to include them into our dataset as a control. Hence, there were 13 classes in our family classification dataset — 12 GPCR families and 1 non-GPCR family, the largest of which comprise 80% of the dataset. A ten-fold cross-validation was used as our evaluation protocol.

3.2 Level I Subfamily Classification

Since we are using the results of the various classifiers studied by Karchin *et al.* (2002) as the baseline for our subfamily classifications, we have used the same datasets and evaluation protocol in our evaluation at the level I and II subfamily classification. In level I subfamily classification, 1269 sequences from subfamilies within Classes A and C, as well as 149 non-GPCR sequences from archaea rhodopsins and G-alpha proteins were used. The non-GPCR sequences were grouped together as a single class of negative examples for our classifier evaluation. The majority class, Peptide subfamily in Class A, comprises 27% of the dataset. We performed a two-fold cross-validation using the same training-testing data split as in the study by Karchin *et al.* (2002). The dataset and training-testing data split is available at http://www.soe.ucsc.edu/research/compbio/gpcr/subfamily_seqs.

3.3 Level II Subfamily Classification

In level 2 subfamily classification, we used 1170 sequences from Classes A and C, and 248 sequences from arachaea rhodopsins, G-alpha proteins, and GPCRs with no level II subfamily classification or in a level II subfamily containing only 1 protein. As before, the 248 sequences were grouped together as a single class of negative examples and a two-fold cross-validation was performed using the same training-testing data split as in the study by Karchin *et al.* (2002). The majority class is the set of negative examples which comprises 17.5% of the dataset.

4 Results and Discussion

In the following, we first present the result of our classifier as we attempt to optimize it using the family-level classification dataset. We then compare the performance of our classifier on the subfamily-level classification datasets against several classifiers of varying computational complexity presented in the study by Karchin *et al.* (2002). Finally, we examine the significance of the features selected by chi-square.

4.1 Family-Level Classification

We began by running the decision tree and Naïve Bayes classifiers on all the n-grams of a particular size — 1, 2 … n for decision tree and 2, 3 … n for Naïve Bayes. The maximum n for the decision tree was 3 ("tri-grams") due to the limitations in the number of features allowed by the C4.5 software. For Naïve Bayes, n up to 5 were included. The results are shown in Table 5. The addition of larger-sized n-grams as features for the decision tree had little effect on its accuracy. In contrast, the Naïve Bayes classifier performed significantly better with bi-grams and trigrams together than with bi-grams alone. Addition of n-grams of length greater than 3 decreased the accuracy. Based on these results, we employed chi-square feature selection on the

set of all unigrams, bi-grams and tri-grams for the decision tree classifier, and on the set of all bi-grams and tri-grams for the Naïve Bayes classifier.

Decision Tree		Naïve Bayes	
N-grams Used	**Accuracy**	**N-grams Used**	**Accuracy**
1-gram	89.4 %	2-grams	80.7 %
1, 2-grams	89.5 %	2, 3-grams	96.3 %
1, 2, 3-grams	89.3 %	2, 3, 4-grams	95.6 %
		2, 3, 4, 5-grams	94.8 %

Table 5. Result of ten-fold cross-validation on GPCR classification at the family level using decision tree and Naïve Bayes classifier on all n-grams of the specified sizes.

To determine the optimal number of features, *K*, the chi-square algorithm needs to select for each of the classifier, we measured the accuracy of the classifier as a function of *K*. We investigated both using the binary features of having at least i_{max} occurrences of each selected n-gram *j* and using the count of n-gram *j* as mentioned in section 2.3. Overall, the accuracy of the classifier increases with *K* until a maximum accuracy is reached, after which the accuracy drops as *K* continues to increase. Using the count of the selected n-grams instead of their binary features resulted in a higher accuracy with the decision tree, while no significant difference resulted with the Naïve Bayes classifier.

The effect of chi-square feature selection on classification accuracy is reported in Table 6. While chi-square feature selection improved the accuracy of the decision tree with unigrams, bi-grams and tri-grams, it had little effect on the Naïve Bayes classifier with bi-grams and tri-grams. Despite the improvement in accuracy of the decision tree, it is still lower than the accuracy of the Naïve Bayes classifier. With both classifiers, chi-square feature selection reduces the number of features needed for them to achieve their respective optimal accuracy.

GPCR sequences vary significantly in length. For example, the rhodopsin sequence in Class A is one of the shortest GPCR sequences with 348 amino acids only, while the longest GPCR sequences having several thousands of amino acids belong to Class B. We therefore investigated whether the protein sequence length is a useful feature in GPCR classification by plotting a histogram of the sequence length, separated by the GPCR family and the control group, bacteriorhodopsins. While this plot confirms that there is significant variation in sequence length, it also shows that the range of sequence length within each GPCR family overlaps significantly, leading to the confusion of the classifier. We confirmed this by training a decision tree on the sequence length in addition to the counts of all unigrams, bi-grams and tri-grams. The resulting tree gave an improvement of 0.1% in test set accuracy over using the decision tree with only those n-grams. Moreover, the sequence length appeared as a node of the decision tree in only one of the ten trials in a ten-fold cross validation, and the node was at the 11^{th} level. Both histogram and experiment result therefore suggest that sequence length is not a distinguishing feature in GPCR classification.

4.2 Level I Subfamily Classification

The experiments on family-level classification described above demonstrated that chi-square feature selection is beneficial, not only in reducing the number of features needed but also in improving the classification accuracy. We therefore tested if a similar improvement may be obtained at the subfamily levels. As before, we measured the classification accuracy as a function of the number of features, *K*, using unigrams, bi-grams and tri-grams with the decision tree, and bi-grams and tri-grams with the Naïve Bayes classifier. The accuracy was computed from a two-fold cross-validation using the same dataset and training-testing data split as in the study by Karchin *et al.* (2002) for ease of comparison to the classifiers presented in their study.

Similar to the family-level classification, the accuracy increases as *K* increases until a maximum is reached, after which the accuracy decreases. Therefore, an improvement can be obtained by using only a subset of the features selected by chi-square. The results are shown in Table 6, along with a reproduction of the results reported by Karchin et al. (2002) on the same dataset and using the same evaluation procedure.

As Table 6 clearly shows, chi-square feature selection improves the accuracy of level I subfamily classification as well. In both family-level and level I subfamily classification, employing the decision tree and the Naïve Bayes classifier on the counts of n-grams selected by chi-square instead of the selected binary features reduces the number of features needed to achieve their respective maximum accuracy.

Table 6 compares the performance of our two simple classifiers, the decision tree and the Naïve Bayes classifier, against those classifiers in the study by Karchin *et al.* (2002). The Naïve Bayes classifier outperforms all other classifiers in level I subfamily classification, including SVM, a much more complicated classifier whose computational complexity was claimed to be needed to achieve "annotation-quality" accuracy in GPCR subfamily classification (Karchin *et al.*, 2002).

4.3 Level II Subfamily Classification

Next, we repeated the above experiments for level II subfamily classification. Plotting the accuracy of the decision tree and the Naïve Bayes classifier as a function of the number of features *K* using a two-fold cross validation with the same training-testing data split as in the study by Karchin *et al.* (2002) produced graphs similar to those in level I subfamily classification (data not shown). The accuracy of our classifiers with and without chi-square feature selection is shown in Table 6, along with a reproduction of the results reported by Karchin *et al.* (2002).

Here, using binary features selected by chi-square with the Naïve Bayes classifier was more effective than using the counts of the corresponding n-grams, giving an improvement of 10.5% in accuracy. Comparison with the previously studied classifiers shows that the Naïve Bayes classifier outperformed all other classifiers with an improvement of 6.1% over SVM, the best out of the previously studied classifiers. While the decision tree performed worse than SVM, with the aid of chi-square feature selection, it still outperformed HMM and kernNN. This result shows that the computational complexity of SVM, which has been previously claimed to be necessary for high accuracy in GPCR level II subfamily classification (Karchin *et al.*, 2002), can be avoided by using the simple feature selection algorithm chi-square.

Classifier	# of Features	Type of Features	Accuracy
Family Classification			
Decision Tree	All (9723)	N-gram counts	89.3 %
	1100	Binary	89.2 %
	500	N-gram counts	90.2 %
Naïve Bayes	All (9702)	N-gram counts	94.7 %[4]
	3900	Binary	95.1 %
	6900	N-gram counts	95.0 %
Level I Subfamily Classification			
Decision Tree	All (9723)	N-gram counts	77.2 %

[4] The accuracy of using all bigrams and trigrams reported here differs from that in Table 5 because results in Table 6 were generated using the cross-validation option provided in Rainbow toolkit, which allows overlaps between the folds (that is, it performs replacement before drawing samples for the test set at each fold). This gives a small increase in accuracy over the cross-validation method used here where no overlaps are allowed when the class size is small (such as the Nematode Chemoreceptors) where there may be instances of the class in the test set but not in the training set.

	2700	Binary	78.0 %
	700	N-gram counts	78.4 %
Naïve Bayes	All (9702)	N-gram counts	90.0 %
	7400	Binary	93.2 %
	6300	N-gram counts	90.9 %
SVM	9 per match state in the HMM	Gradient of the log-likelihood that the sequence is generated by the given HMM model	88.4 %
BLAST	Local sequence alignment		83.3 %
SAM-T2K HMM	A HMM model built for each protein subfamily		69.9 %
kernNN	9 per match state in the HMM	Gradient of the log-likelihood that the sequence is generated by the given HMM model	64.0 %
Level II Subfamily Classification			
Decision Tree	All (9723)	N-gram counts	66.0 %
	2300	Binary	70.2 %
	1200	N-gram counts	70.8 %
Naïve Bayes	All (9702)	N-gram counts	81.9 %
	8100	Binary	92.4 %
	5600	N-gram counts	84.2 %
SVM	9 per match state in the HMM	Gradient of the log-likelihood that the sequence is generated by the given HMM model	86.3 %
SVMtree	9 per match state in the HMM	Gradient of the log-likelihood that the sequence is generated by the given HMM model	82.9 %
BLAST	Local sequence alignment		74.5 %
SAM-T2K HMM	A HMM model built for each protein subfamily		70.0 %
kernNN	9 per match state in the HMM	Gradient of the log-likelihood that the sequence is generated by the given HMM model	51.0 %

Table 6. Comparison of the accuracy of various classifiers at GPCR level II subfamily classification. Unigrams, bi-grams and tri-grams are used with the decision tree, while bi-grams and tri-grams are used with the Naïve Bayes classifier. Results of SVM, BLAST, HMM and kernNN from the study by Karchin *et al.* (2002) are reproduced above for ease of comparison.

4.4 Significance of Selected Features

In light of the successful results in classification, we were curious whether there is biological significance to the n-grams selected by chi-square. However, examining the importance of bigrams and trigrams is difficult because they are likely to occur multiple times in a sequence. Moreover, biologists are interested in whether properties characteristic of a single class rather than multiple classes. Thus, we extracted all 4-grams in addition to the bigrams and trigrams and employed a modified chi-square feature selection to select the 20 most important n-grams for the Class B family. The modification is, in essence, the removal of the summation across all classes so that a significance measure is computed for each n-gram-class pair. We plotted the top 20 n-grams on the snake-plots of several GPCRs. An example is shown in Figure 1 with PTRR_HUMAN. From the snake-plots, we observed that most of the selected n-grams lie in the

cytoplasmic domains of the receptors. This is consistent with the finding by Vriend and co-workers that conserved regions often lie in the G-protein activating domains. A number of the other selected n-grams are in helix 3 and 7, both known to be important for signal transduction. Ongoing work is being conducted to investigate the biological interpretation of these selected n-grams further.

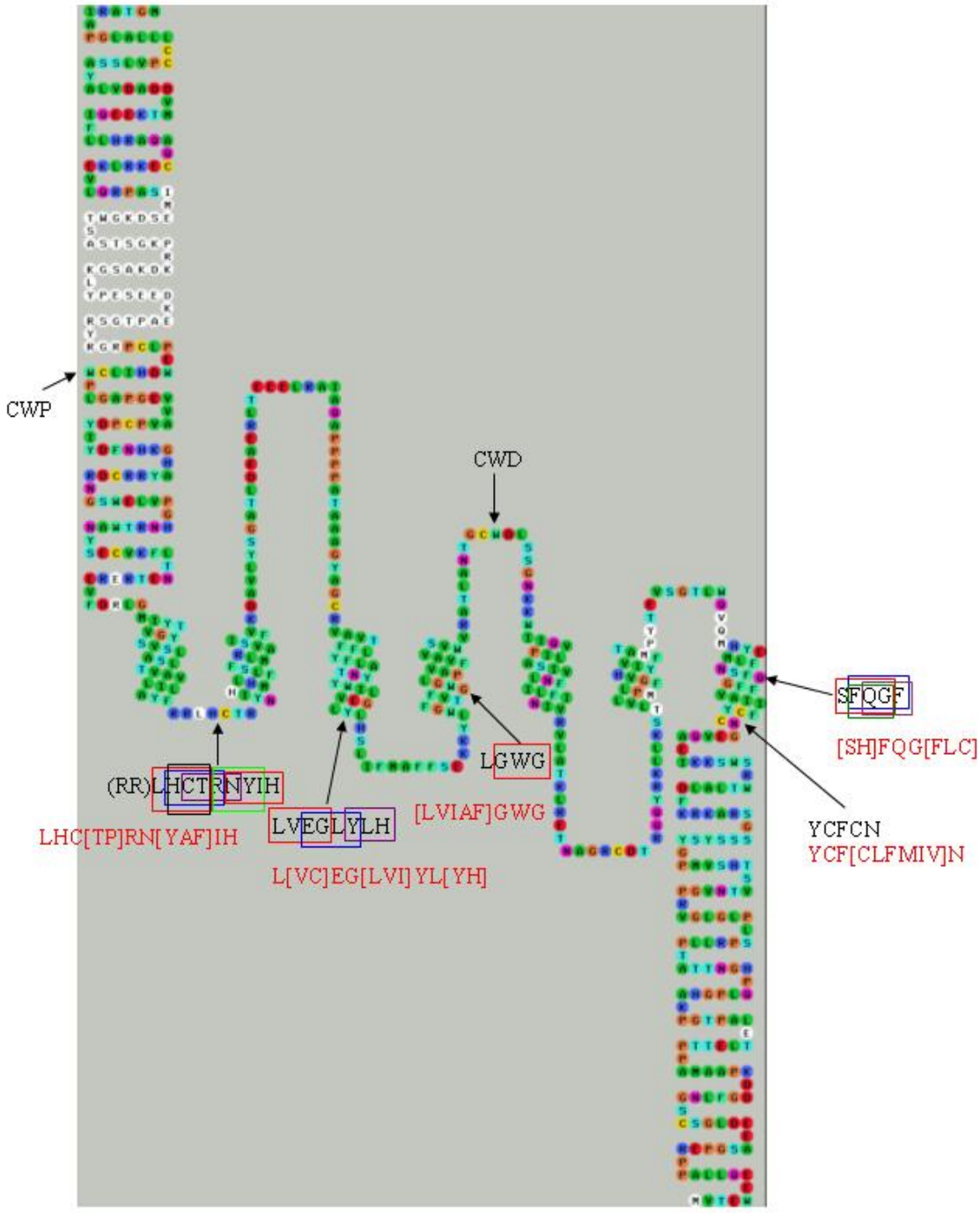

Figure 1. Snake-plot of PTRR_HUMAN in Class B GPCRs with the 20 most significant n-grams as selected by modified chi-square feature selection labeled.

5 Conclusions and Future Work

In this study, we evaluated the performance of simple classifiers in conjunction with feature selection against more complicated classifiers in the task of protein sequence classification. We chose to use the superfamily of G-protein coupled receptors as our dataset because of its biological importance, particularly in pharmacology, and the known difficulty it presents in the classification task due to the extreme diversity among its members. In analogy to document classification in the human language technologies domain, we used the decision tree and Naïve Bayes classifier, and began our experiments with classification at the family level. We first optimized our classification procedure with feature selection using this dataset. In document classification, chi-square feature selection has proven to be highly successful (Yang and Pedersen, 1997), not only in reducing the number of features necessary for accurate classification, but also in increasing classification accuracy via the elimination of "noisy features". We applied chi-square feature selection to the GPCR family classification task and found that chi-square was successful in this task as well. Specifically, using chi-square feature selection, the accuracy increased with the number of features until a maximum accuracy was reached, after which the accuracy dropped. Thus, an improvement in accuracy can be attained by using chi-square to reduce the dimensionality of the feature space to the point at which the maximum accuracy occurs.

We then applied our method to the GPCR level I and II subfamily classification tasks studied previously by Karchin *et al.* (2002) in a systematic comparison of classifiers of varying complexity. For comparability, we used the same dataset and evaluation procedure as published in the previous study. First, we note that subfamily classifications are much more difficult to predict than family level classifications, as shown by the decrease in accuracy of both the decision tree and Naïve Bayes classifier. This observation is consistent with the fact that subfamilies are defined to a greater extent than families are by chemical and pharmacological criteria as opposed to sequence homology. Because of these difficulties, the previous study (Karchin *et al.*, 2002) had concluded that at the subfamily levels, more complex classifiers are needed to maintain high classification accuracy. In particular, the accuracies of BLAST, k-nearest neighbors in conjunction with Fisher Score Vector space, profile HMM and SVM in level I and II subfamily classification had been studied with alignment-based features, and SVM was found to be required to attain "annotation-quality" accuracy. Using SVM, an accuracy of 88.4% and 86.3% had been achieved in level I and II subfamily classification (Karchin *et al.*, 2002), see Table 6. In level I subfamily classification, we found that the Naïve Bayes classifier using the counts of all bi-grams and tri-grams can outperform SVM by 1.6%. Moreover, a greater improvement of 4.8% over SVM can be achieved if chi-square feature extraction is used in conjunction with the Naïve Bayes classifier, leading to a final accuracy of 93.2%. In level II subfamily classification, the Naïve Bayes classifier with the aid of chi-square feature selection surpassed SVM by 6.1% and achieved an accuracy of 92.4%. The comparison of results from the classifiers in our study and that of Karchin *et al.* (2002) showed that the decision tree cannot match the performance of the Naïve Bayes classifier and SVM in either level I or II subfamily classification. However, chi-square improves the accuracy of the decision tree to the extent that it outperforms HMM in both of these tasks.

One interesting observation in our level I subfamily classification results (Table 6) is that while the Naïve Bayes classifier performed better with the help of chi-square feature selection, it outperformed all other classifiers even on its own using counts of all bi-grams and trigrams. This suggests that the difference in performance between the Naïve Bayes classifier and SVM may be due to the different features used. In particular, n-grams may be a better set of features than alignment-based features for protein classification. Biologically, this means that one reason for the improvement in classification accuracy may be the use of small peptide fragments (n-grams) which do not require the sequential arrangement necessary in a sequence alignment. Although

sequence alignment has dominated the field for many years because of its intuitive nature in understanding the evolutionary origin of protein families and subfamilies, relaxing the requirement for consecutive features is more in tune with the hallmark of protein structures. Protein structures are functional because of their arrangement in three-dimensional space, bringing about important contact between amino acids that may be far apart in the linear amino acid sequence. From our current experiments, we cannot distinguish if the type of features, the feature selection process or the different classifier has caused the significant improvement of our simple Naïve Bayes classifier over the SVM classifier. Further experiments using SVM on n-gram features alone and in conjunction with chi-square feature selection are ongoing to determine whether the improvement in accuracy is due to the specific classifier (that is, the Naïve Bayes classifier versus SVM), the feature set or chi-square feature selection.

From the study presented here, we conclude that complicated classifiers at the complexity of SVM are not necessary to attain high accuracy in protein classification, even for the particularly challenging GPCR subfamily classification task. A simple classifier, the Naïve Bayes classifier, in conjunction with chi-square feature selection, applied to n-gram counts can perform better than SVM on alignment-based features. Another simple classifier, Decision Tree with chi-square feature selection, while not as powerful as either Naïve Bayes or SVM, can still outperform profile HMM. Moreover, the n-grams selected by chi-square feature selection seem to have biological significance. Further work is being conducted to investigate this further. The methods presented here were all originally applied to the document classification task in human language technologies domain. The successful application of document classification techniques to the protein classification task, together with the conclusion that simple classifiers can outperform complicated classifiers in this task as a result, have important implications. There are many problems in the biology domain that can be formulated as a classification task. Many of these are considered to be more challenging by biologists than the protein classification task. This includes predicting folding, tertiary structure and functional properties of proteins, such as protein-protein interactions. Thus, these important classification tasks are potential areas for applications of human language technologies in modern proteomics.

Acknowledgements

The authors gratefully acknowledge the financial support by National Science Foundation Large Information Technology Research grant NSF 0225656.

References

S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool, *Journal of Molecular Biology*, 215(3):403-410, 1990.

S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389-3402, 1997.

R. Apweiler, A. Gateau, S. Contrino, M. J. Martin, V. Junker, C. O'Donovan, F. Lang, N. Mitaritonna, S. Kappus, and A. Bairoch. Protein sequence annotation in the genome era: the annotation concept of SWISS-PROT + TREMBL. In *Proceedings of 5th International Conference on Intelligent Systems for Molecular Biology*, pages 33-43, Menlo Park, California, 1997.

T. K. Attwood, M. Blythe, D. R. Flower, A. Gaulton, J. E. Mabey, N. Maudling, L. McGregor, A. Mitchell, G. Moulton, K. Paine, and P. Scordis. PRINTS and PRINTS-S shed light on protein ancestry. *Nucleic Acids Research*, 30(1):239-241, 2002.

P. Baldi and S. Brunak. *Bioinformatics: The Machine Learning Approach*. MIT Press, Cambridge, USA, 2001.

A. Bateman, E. Birney, L. Cerruti, R. Durbin, L. Etwiller, S. R. Eddy, S. Griffiths-Jones, K. L. Howe, M. Marshall, and E. L. L. Sonnhammer. The Pfam protein families database. *Nucleic Acids Research*, 30(1):276-280, 2002.

G. J. Barton and J. E. Sternberg. A strategy for the rapid multiple alignment of protein sequences. Confidence levels from tertiary structure comparisons. *Journal of Molecular Biology*, 198(2):327-337, 1987.

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 in 2003. *Nucleic Acids Research*, 31(1):365-370, 2003.

Bow: a toolkit for statistical language modeling, text retrieval, classification and clustering, February 13, 2002 release. http://www-2.cs.cmu.edu/~mccallum/bow/

P. Bucher, K. Karplus, N. Moeri, and K. Hoffman. A flexible motif search technique based on generalized profiles. *Computers and Chemistry*, 20(1):3-24, 1996.

C4.5, release 8. http://www.cse.unsw.edu.au/~quinlan/

F. Corpet, F. Servant, J. Gouzy, and D. Kahn. ProDom and ProDom-CG: tools for protein domain analysis and whole genome comparisons. *Nucleic Acids Research*, 28(1):267-269, 2000.

M. Deshpande and G. Karypis. Evaluation of Techniques for Classifying Biological Sequences. In *Proceedings of the 6th Pacific-Asia Conference on Knowledge Discovery (PAKDD)*, pages 417-431, Taipei, Taiwan, 2002.

E. Eskin, W. N. Grundy, and Y. Singer. Protein Family Classification using Sparse Markov Transducers. In *Proceedings of the 8th International Conference on Intelligent Systems for Molecular Biology*, San Diego, California, 2000.

L. Falquet, M. Pagni, P. Bucher, N. Hulo, C. J. Sigrist, K. Hofmann, and A. Bairoch. The PROSITE database, its status in 2002. *Nucleic Acids Research*, 30(1):235-238, 2002.

E. A. Ferran and P. Ferrara. Clustering proteins into families using artificial neural networks. *Computer Applications in the Biosciences*, 8(1):39-44, 1992. Erratum in *Computer Applications in the Biosciences*, 8(3):305, 1992.

GENEWISE, 2002. http://www.ebi.ac.uk/Wise2/

U. Gether. Uncovering Molecular Mechanisms Involved in Activation of G Protein-Coupled Receptors. *Endocrine Reviews*, 21(1):90-113, 2000.

J. Gough, K. Karplus, R. Hughey, and C. Chothia. Assignment of Homology to Genome Sequences using a Library of Hidden Markov Models that Represent all Proteins of Known Structure. *Journal of Molecular Biology*, 313(4):903-919, 2001.

W. N. Grundy, T. L. Bailey, C. P. Elkan, and M. E. Baker. Meta-MEME: Motif-based Hidden Markov Models of Biological Sequences. *Computer Applications in the Biosciences*, 13(4):397-406, 1997.

J. G. Henikoff, E. A. Greene, S. Pietrokovski, and S. Henikoff. Increased coverage of protein families with the blocks database servers. *Nucleus Acids Research*, 28(1):228-230, 2000.

S. Henikoff, J. G. Henikoff, W. J. Alford, and S. Pietrokovski. Automated construction and graphical presentation of protein blocks from unaligned sequences. *Gene*, 163(2):GC17-26, 1995.

S. Henikoff, J. G. Henikoff, and S. Pietrokovski. Blocks+: A non-redundant database of protein alignment blocks derived from multiple compilations. *Bioinformatics*, 15(6):471-479, 1999.

HMMER, 2003. http://hmmer.wustl.edu/

HMMpro, v. 2.2. http://www.netid.com/html/hmmpro.html

F. Horn, J. Weare, M. W. Beukers, S. Hörsch, A. Bairoch, W. Chen, O. Edvardsen, F. Campagne, and G. Vriend. GPCRDB: an information system for G protein-coupled receptors. *Nucleic Acids Research*, 26(1):275-279, 1998.

J. Y. Huang and D. L. Brutlag. The EMOTIF database. *Nucleic Acids Research*, 29(1):202-204, 2001.

T. Jaakkola, M. Diekhans, and D. Haussler. A discriminative framework for detecting remote protein homologies. *Journal of Computational Biology*, 7(1-2):95-114, 2000.

T. Jaakkola, M. Diekhans, and D. Haussler. Using the fisher kernel method to detect remote protein homologies. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, pages 149-158, Heidelberg, Germany, 1999.

R. Karchin, K. Karplus, and D. Haussler. Classifying G-protein coupled receptors with support vector machines. *Bioinformatics*, 18(1):147-159, 2002.

J. Kim, E. N. Moriyama, C. G. Warr, P. J. Clyne, and J. R. Carlson. Identification of novel multi-transmembrane proteins from genomic databases using quasi-periodic structural properties. *Bioinformatics*, 16(9):767-775, 2000.

L. F. Kolakowski Jr. GCRDb: a G-protein-coupled receptor database. *Receptors Channels*, 2(1):1–7, 1994.

A. Krogh, M. Brown, I. S. Mian, K. Sjolander, and D. Haussler. Hidden Markov models in computational biology: Applications to protein modeling. *Journal of Molecular Biology*, 235(5):1501-1531, 1994.

M. Lapinsh, A. Gutcaits, P. Prusis, C. Post, T. Lundstedt, and J. E. Wikberg. Classification of G-protein coupled receptors by alignment-independent extraction of principal chemical properties of primary amino acid sequences. *Protein Science*, 11(4):795-805, 2002.

C. Leslie, E. Eskin, and W. S. Noble. The spectrum kernel: A string kernel for SVM protein classification. In *Proceedings of the Pacific Symposium on Biocomputing*, pages 564-575, Lihue, Hawaii, 2002a.

C. Leslie, E. Eskin, J. Weston, and W. Noble. Mismatch String Kernels for SVM Protein Classification. In *Proceedings of the Sixteenth Annual Conference on Neural Information Processing Systems*, British Columbia, Canada, 2002b.

I. Letunic, L. Goodstadt, N. J. Dickens, T. Doerks, J. Schultz, R. Mott, F. Ciccarelli, R. R. Copley, C. P. Ponting, and P. Bork. Recent improvements to the SMART domain-based sequence annotation resource. *Nucleic Acids Research*, 30(1):242-244, 2002.

M. E. Levchenko, T. Katayama, and M. Kanehisa. Discovery and Classification of Peptide Family G-Protein Coupled Receptors in the Human Genome Sequence. In *Genome Informatics*, pages 352-353, Tokyo, Japan, 2001.

A. Liu and A. Califano. Functional Classification of Proteins by Pattern Discovery and Top-down Clustering of Primary Sequences. *IBM Systems Journal: Deep Computing for the Life Sciences*, 40(2):379-393, 2001.

M. Lynch. Intron evolution as a population-genetic process. *Proceedings of National Academy of Sciences of U S A*, 99(9):6118-6123, 2002.

H. Mitsuke, Y. Sugiyama, and T. Shimizu. Classification of Transmembrane Protein Families Based on Topological Similarity. *Genome Informatics*, 13:418-419, 2002.

B. Morgenstern. DIALIGN 2: improvement of the segment-to-segment approach to multiple sequence alignment. *Bioinformatics*, 15(3):211-218, 1999.

B. Morgenstern, K. Frech, A. Dress, and T. Werner. DIALIGN: finding local similarities by multiple sequence alignment. *Bioinformatics*, 14(3):290-294, 1998.

E. N. Moriyama and J. Kim. Protein Family Classification with Discriminant Function Analysis. In *Proceedings of Stadler Genetics Symposium*, 2003.

G. Muller. Towards 3D structures of G protein-coupled receptors: a multidisciplinary approach. *Current Medical Chemistry*, 7(9):861-888, 2000.

S. B. Needleman, and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequences of two proteins. *Journal of Molecular Biology*, 48:443-453, 1970.

A. F. Neuwald and P. Green. Detecting patterns in protein sequences. *Journal of Molecular Biology*, 239(5):698-712, 1994.

A. F. Neuwald, J. S. Liu, D. J. Lipman, and C. E. Lawrence. Extracting protein alignment models from the sequence database. *Nucleic Acids Research*, 25(9):1665-1677, 1997.

C. Notredame, and D. G. Higgins. SAGA: sequence alignment by genetic algorithm. *Nucleic Acids Research*, 24(8): 1515-1524, 1996.

C. Notredame, D. Higgins, and J. Heringa. T-Coffee: A novel method for multiple sequence alignments. *Journal of Molecular Biology*, 302(1):205-217, 2000.

J. Park, S. A. Teichmann, T. Hubbard, and C. Chothia. Intermediate sequences increase the detection of homology between sequences. *Journal of Molecular Biology*, 273(1):349-354, 1997.

W. R. Pearson. Effective protein sequence comparison. *Methods Enzymol*, 266:227-258, 1996.

W. R. Pearson. Empirical statistical estimates for sequence similarity searches. *Journal of Molecular Biology*, 276(1):71-84, 1998.

W. R. Pearson. Flexible sequence similarity searching with the FASTA3 program package. *Methods in Molecular Biology*, 132:185-219, 2000.

C. P. Ponting, J. Schultz, F. Milpetz, and P. Bork. SMART: identification and annotation of domains from signalling and extracellular protein sequences. *Nucleic Acids Research*, 27(1):229-232, 1999.

K. Qu, L. A. McCue, C. E. Lawrence. Bayesian protein family classifier. In *Proceedings of the 6th International Conference on Intelligent Systems for Molecular Biology*, pages 131-139, Montreal, Canada, 1998.

G. D. Schuler, S. F. Altschul, and D. J. Lipman. A workbench for multiple alignment construction and analysis. *Proteins*, 9(3):180-190, 1991.

F. Sebastiani. A Tutorial on Automated Text Categorization. In *Proceedings of ASAI*, pages 7-35, Buenos Aires, Argentina, 1999.

F. Servant, C. Bru, S. Carrère, E. Courcelle, J. Gouzy, D. Peyruc, and D. Kahn. ProDom: Automated clustering of homologous domains. *Briefings in Bioinformatics*, 3(3):246-251, 2002.

C. J. Sigrist, L. Cerutti, N. Hulo, A. Gattiker, L. Falquet, M. Pagni, A. Bairoch, and P. Bucher. PROSITE: a documented database using patterns and profiles as motif descriptors. *Briefings in Bioinformatics*, 3(3):265-74, 2002.

H. O. Smith, T. M. Annau, and S. Chandrasegaran. Finding sequence motifs in groups of functionally related proteins. *Proceedings of the National Academy of Sciences*, 87(2):826-830, 1990.

T. F. Smith and M. Waterman. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147:195-197, 1981.

SSearch program, 2002. http://www.biology.wustl.edu/gcg/ssearch.html

W. R. Taylor. A flexible method to align large numbers of biological sequences. *Journal of Molecular Evolution*, 28(1-2):161-169, 1988.

J. D. Thompson, D. G. Higgins, and T. J. Gibson. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Research*, 22(22):4673-4680, 1994.

B. Vanschoenwinkel, J. Reumers, and B. Manderick. A Text Mining and Support Vector Machine Approach to Protein Classification. *Knowledge Discovery meets Drug Discovery workshop (poster)*, Leuven, Belgium, 2002.

S. Vinga and J. Almeida. Alignment-free sequence comparison - a review. *Bioinformatics*, 19(4):513-523, 2003.

J. T. L. Wang, Q. Ma, D. Shasha, and C. H. Wu. Application of neural networks to biological data mining: a case study in protein sequence classification. In *Proceedings of the 6th ACM*

SIGKDD International Conference on Knowledge Discovery & Data Mining, pages 305-309, Boston, MA, 2000.

Wisconsin Package, v. 10.3. http://www.accelrys.com/products/gcg_wisconsin_package/

C. Wu, M. Berry, S. Shivakumar and J. McLarty. Neural Networks for Full-Scale Protein Sequence Classification: Sequence Encoding with Singular Value Decomposition. *Machine Learning*, 21(1):177-193, 1995.

C. H. Wu, H. Huang H, L. L. Yeh, and W. C. Barker. Protein family classification and functional annotation. *Computational Biology and Chemistry*, 27(1):37-47, 2003.

Y. Yang and J. Pedersen. A Comparative Study on Feature Selection in Text Categorization. In *Proceedings of 14th International Conference on Machine Learning*, pages 412-420, Nashville, US, 1997.

G. Yona, N. Linial, and M. Linial. ProtoMap: Automatic classification of protein sequences, a hierarchy of protein families, and local maps of the protein space. *Proteins*, 37:360-378, 1999.

X. Yuan, X. Yuan, B. P. Buckles, and J. Zhang. A Comparison Study of Decision Tree and SVM to Classify Gene Sequence. In *Proceedings of the 19th International Conference on Data Engineering*, Bangalore, India, 2003.

Y. X. Zhang, K. Perry, V. A. Vinci, K. Powell, W. P. Stemmer, and S. B. del Cardayre. Genome shuffling leads to rapid phenotypic improvement in bacteria. *Nature*, 415(6872):644-646, 2002.

Z. Zhang, A. A. Schaffer, W. Miller, T. L. Madden, D. J. Lipman, E. V. Koonin, and S. F. Altschul. Protein sequence similarity searches using patterns as seeds. *Nucleic Acids Research*, 26(17):3986-3990, 1998.