A novel method for classifying subfamilies and sub-subfamilies of G-protein coupled receptors

Majid Beigi, Andreas Zell

University of Tübingen
Center for Bioinformatics Tübingen (ZBIT)
Sand 1, D-72076 Tübingen, Germany
{majid.beigi,andreas.zell}@uni-tuebingen.de

Abstract. G-protein coupled receptors (GPCRs) are a large superfamily of integral membrane proteins that transduce signals across the cell membrane. Because of that important property and other physiological roles undertaken by the GPCR family, they have been an important target of therapeutic drugs. The function of many GPCRs is not known and accurate classification of GPCRs can help us to predict their function. In this study we suggest a kernel based method to classify them at the subfamily and sub-subfamily level. To enhance the accuracy and sensitivity of classifiers at the sub-subfamily level that we were facing with a low number of sequences (imbalanced data), we used our new synthetic protein sequence oversampling (SPSO) algorithm and could gain an overall accuracy and Matthew's correlation coefficient (MCC) of 98.4 % and 0.98 for class A, nearly 100% and 1 for class B and 96.95% and 0.91 for class C, respectively, at the subfamily level and overall accuracy and MCC of 97.93% and 0.95 at the sub-subfamily level. The results shows that Our oversampling technique can be used for other applications of protein classification with the problem of imbalanced data.

1 Introduction

G-protein coupled receptors (GPCRs) are a large superfamily of integral membrane proteins that transfer signals across the cell membrane. Through their extracelluar and transmembrane domains they respond to a variety of ligands, including neurotransmitters, hormones and odorants. They are characterized by seven hydrophobic regions that pass through the cell membrane (transmembrane regions) [1], as shown in Fig. 1. Each GPCR has an amino terminal (NH2 or N-terminal) region outside of the cell, followed by intracellular and extracellular loops, which connect the seven transmembrane regions, and also an intracellular carboxyl terminal (COOH- or C-terminal) region. GPCRs are involved in signal transmission from the outside to the interior of the cell through interaction with heterotrimeric G-proteins, or proteins that bind to guanine (G) nucleotides. The receptor is activated when a ligand that carries an environmental signal binds to a part of its cell surface component.A wide range of molecules is used as

the ligands including peptide hormones, neurotransmitters, pancrine mediators, etc., and they can be in many forms: e.g., ions, amino acids, lipid messengers and protease [2].

The function of many GPCRs are unknown and understanding the signaling pathways and their ligands in laboratory is expensive and time-consuming. But the sequence of thousands of GPCRs are known [3]. Hence, if we can develop an accurate predictor of the class (and so function) of GPCRs from their sequence it can be of great usefulness for biological and pharmacological research. According to the binding of GPCRs to different ligand types they are classified into different families. Based on GPCRDB (G protein coupled receptor data base) [3] all GPCRs have been divided into a hierarchy of 'class', 'subfamily', 'sub-sub-family' and 'type' (Fig. 2).

Because of the divergent nature of GPCRs it is difficult to predict the classification of GPCRs by means of sequence alignment approaches. The standard bioinformatics approach for function prediction of proteins is to use sequence comparison tools such as PSI-BLAST [4] that can identify homologous proteins based on the assumption of low evolutionary divergence, which is not true for GPCRs families. Here, we are facing a more difficult problem of remote homology detection, where classifiers must detect a remote relation between unknown sequence and training data.

There have been several recent developments to the classification problem specific to the GPCR superfamilies. Moriyama and Kim [5] developed a classification method based on discriminant function analysis using composition and physicochemical properties of amino acids. Elrod and Chou [6] suggested a covariant discriminant algorithm to predict GPCR's type from amino acid composition. Qian et al. [7] suggested a phylogenetic tree based profile hidden Markov model (T-HMM) for GPCR classification. Karchin et al. [8] developed a system based on support vector machines built on profile HMMs. They generated fisher score vectors [9] as features for SVM classifier form those profile HMMs. They showed that classifiers like SVMs that are trained on both positive and negative examples can increase the accuracy of GPCRs classification compared with only HMMs as generative method.

To increase the accuracy of remote homology detection by discriminative methods, researchers also focused on finding new kernels, which measure the similarity between sequences, as main part of SVM based classifiers. So after choosing an appropriate feature space, and representing each sequence as a vector in that space, one takes the inner product between these vector-space representations. Spectrum kernel [10], Mismatch kernel [11] and Local alignment kernel [12] are examples of those kernels and it has been shown that they have outperformed previous generative methods for remote homology detection.

In our study we want to classify GPCRs at the subfamily and sub-subfamily level. In this case, a problem in classification of GPCRs is the number of proteins at the sub-subfamily level. At this level in some sub-subfamilies we have only a very low number of protein sequences as positive data (minor class) compared with others (major class). In general, with imbalanced data, the SVM classifier

tends to perform best for classifying the majority class but fails to classify the minority class correctly. Because of that problem some researchers have not considered those GPCRs families, or if they have included them in their classifier they did not get as good results for them as for other families with enough data [13]. We used a new oversampling technique for protein sequences, explained in [24] to overcome that problem. Based on that method at first we make a HMM profile of those sequences and then try to increase the number of sequences in that family synthetically considering the phylogenetic tree of that family and also the distribution of other families near to that family. For classification, we use the local alignment kernel (LA kernel) that has been shown to have better performance compared with other previously suggested kernels for remote homology detection when applied to the standard SCOP test set [15]. It represents a modification of the Smith-Waterman score to incorporate sub-optimal alignments by computing the sum (instead of the maximum) over all possible alignments. Using that kernel along with our oversampling technique we could get better accuracy and Matthew's correlation coefficient for the classification of GPCRs at the subfamily and sub-subfamily level than other previously published method.

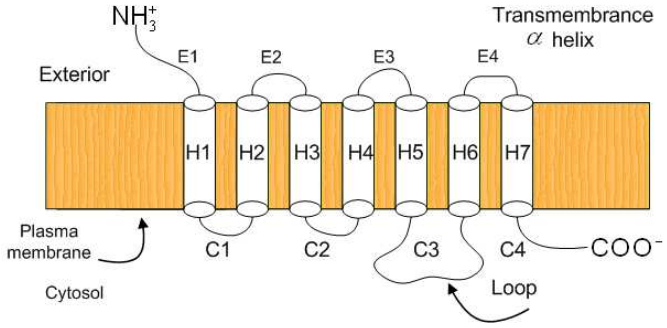

Fig. 1. Schematic representation of GPCR shown as seven transmembrane helices depicted as cylinders along with cytoplasmic and extracellular hydrophilic loops.

2 Materials

The dataset of this study was collected from GPDRDB [3] and we used the latest dataset of GPCRDB (June 2005 release, http://www.gpcr.org/7tm/). The six main families are: Class A (Rhodopsin like), Class B (Secretin like), Class C (Metabotropic glutamate/pheromone), Class D (Fungal pheromone), Class E (cAMP receptors) and Frizzled/Smoothened family. The sequences of proteins in GPCRDB were taken from SWISS-PROT and TrEMBL data banks [14]. All six families of GPCRs (5300 protein sequences) are classified in 43 subfamilies

and 99 sub-subfamilies. The three largest classes are the rhodopsin-like receptors, the secretion-like receptors and the metabotropic glutamate receptors (class A, B, and C). The rhodopsin-like family is the largest and most studied with approximately 90 percent of all receptors (4737 out of 5300).

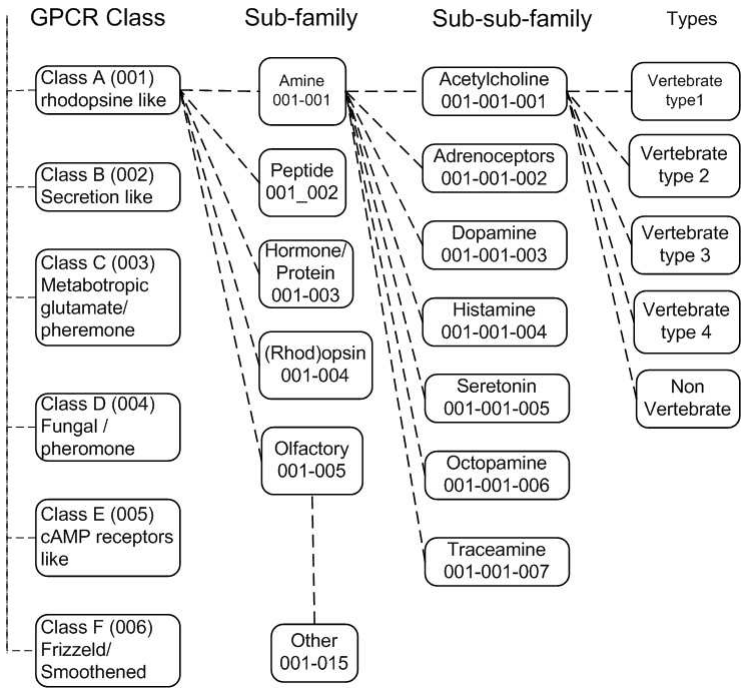

Fig. 2. GPCR family tree according to GPCRDB nomenclature.

3 Algorithms

3.1 Kernel Function

In discriminative methods, a classifier learns a rule to classify unlabelled sequences into a class of proteins by using both sequences belonging to this class (positive examples) and sequences known as not belonging to that class (negative examples). Given a set of positive training sequences χ_+ and a set of negative training sequence χ_- an SVM learns a classification function $f(x)$ of the form:

$$f(x) = \sum_{i;x_i \in \chi_+} \lambda_i K(x, x_i) - \sum_{i;x_i \in \chi_-} \lambda_i K(x, x_i) \quad (1)$$

where non-negative λ_i weights are computed during training by maximizing a quadratic objective function and $K(.,.)$ is the kernel function. Given this function, a new sequence X is predicted to belong to positive dataset if the value of $f(x)$ is positive, otherwise it belongs to the negative dataset.

On the other hand, variable length protein sequences must be converted to fixed length vectors to be accepted as input to a SVM classifier. These vectors should exploit prior knowledge of proteins belonging to one family and enable us to have maximum discrimination for unrelated proteins. So the kernel function is of great importance for SVM classifiers in learning the dataset and also in exploiting prior knowledge of proteins and mapping data from input space to feature space. The Smith Waterman (SW) alignment score between two protein sequences tries to incorporate biological knowledge about protein evolution by aligning similar parts of two sequences but it lacks the positive definiteness as a valid kernel [15]. The local alignment kernel mimics the behavior of the Smith Waterman (SW) alignment score and tries to incorporate the biological knowledge about protein evolution into a string kernel function. But unlike the SW alignment, it has been proven that it is a valid string kernel. We used this kernel for our classification task, so we give a brief introduction to that algorithm: If K_1 and K_2 are two string kernels then the convolution kernel $K_1 \star K_2$ is defined for any two strings x and y by:

$$K_1 \star K_2(x,y) = \sum_{x_1x_2=x, y_1y_2=y} K_1(x_1,y_2)K_2(x_2,y_2) \tag{2}$$

Based on work of Haussler [16] if K_1 and K_2 are valid string kernels, then $K_1 \star K_2$ is also a valid kernel. Vert et al. [12] used that point and defined a kernel to detect local alignments between strings by convolving simpler kernels.The local alignment kernel (LA) consists of three convolved string kernels. The first kernel models the null contribution of a substring before and after a local alignment in the score:

$$\forall (x,y) \in \chi^2, \;\; K_0(x,y) = 1 \tag{3}$$

The second string kernel is for alignment between two residues:

$$K_\alpha^{(\beta)}(x,y) = \begin{cases} 0 \text{ if } |x| \neq 1 \text{ or } |y| \neq 1 \\ \exp[\beta s(x,y)] \quad \text{otherwise,} \end{cases} \tag{4}$$

where $\beta \geq 0$ controls the influence of suboptimal alignments in the kernel value and $s(x,y)$ is a symmetric similarity score or substitution matrix, e.g. BLOSUM62.

The third string kernel models affine penalty gaps:

$$K_g^{(\beta)}(x,y) = \exp\left\{\beta\left[g\left(|x|\right) + g\left(|y|\right)\right]\right\} \tag{5}$$

$g(n)$ is the cost of a gap of length n given by:

$$\begin{cases} g\left(0\right) = 0 & \text{if } n = 0, \\ g\left(n\right) = d + e\left(n-1\right) & \text{if } n \geq 1, \end{cases} \tag{6}$$

where d and e are gap opening and extension costs. After that the string kernel based on local alignment of exactly n residues is defined as:

$$K_n^{(\beta)}(x,y) = K_0 * \left(K_\alpha^{(\beta)} * K_\alpha^{(\beta)}\right)^{(n-1)} * K_\alpha^{(\beta)} * K_0. \tag{7}$$

This kernel quantifies the similarity of two strings x and y based on local alignments of exactly n residues. In order to compare two sequences through all possible local alignments, it is necessary to take into account alignments with different numbers n of aligned residues:

$$K_{LA}^{(\beta)} = \sum_{i=0}^{\infty} K_{(i)}^{(\beta)}. \tag{8}$$

The implementation of the above kernel can be done via dynamic programming [12].

3.2 Synthetic Protein Sequence Oversampling (SPSO)

In classification of GPCRs at the subfamily and specially sub-subfamily level we are facing an imbalanced dataset. There have been two types of solutions to this problem. The first type, as exemplified by different forms of re-sampling techniques, tries to increase the number of minor class examples (oversampling) or decrease the number of major class examples (undersampling) in different ways. The second type adjusts the cost of error or decision thresholds in classification for imbalanced data and tries to control the sensitivity of the classifier [17–20]. In protein classification problems the second type of those approaches has been applied more and a class-depending regularization parameter is added to the diagonal of the kernel matrix: $K'(x,x) = K(x,x) + \lambda n/N$, where n and N are the number of positive (or negative) instances and the whole dataset, respectively.

In GPCR classification, even with that method we could not get good results, especially at the sub-subfamily level. One important issue with imbalanced data is that making the classifier too specific may make it too sensitive to noise specially with highly imbalanced datasets, having a ratio of 100 to 1 and more, the classifier often treats positive data as noise and considers it as negative data and we also have instabilities in the classifier. It means the cost that we consider for an error can be an important issue, and sometimes choosing a value near the optimum value can give unsatisfying results. Then, in this case, only using a different error cost method (DEC) [19] is not suitable. We found out that if we can add synthetic sequences (oversampling) at the sub-subfamily level (minority class) in a way that those added sequences are related to that class and away from other classes (majority class), the accuracy of a classifier will be increased. For that, we used our newly developed algorithm named synthetic protein sequence oversampling (SPSO) technique [24] in which the minority class in the data space is oversampled by creating synthetic examples. It considers the distribution of residues of the protein sequence using a hidden Markov model profile of the minority class and also one of the majority class and then synthesizes protein

sequences which can precisely increase the information of the minor class. We used this method along with the DEC method to increase the sensitivity and stability of the classifier.

4 Results

In this study we used the local alignment kernel (LA kernel) to generate vector from protein sequences. For this, we divided the data into training and test data and then build a kernel matrix K for the training data as shown in Fig. 3. Each cell of the matrix is a local alignment kernel score between protein i and protein j. After that we normalized the kernel matrix via $K_{ij} \leftarrow K_{ij}/\sqrt{K_{ii}K_{jj}}$. We used the SPSO algorithm, explained above, for each subfamily or sub-subfamily whose number of sequences in the training set was less than 50 and more than 4, to synthetically increase the number of data up to 800 percent (depending on the number of sequences). Each subfamily or sub-subfamily is considered as positive training data and all others as negative training data. After that the SVM algorithm with RBF kernel is used for training and for highly imbalanced data (after oversampling) we also use the DEC (different error cost) method. For testing, we create feature vectors by calculating a local alignment kernel between the test sequence and all training data.

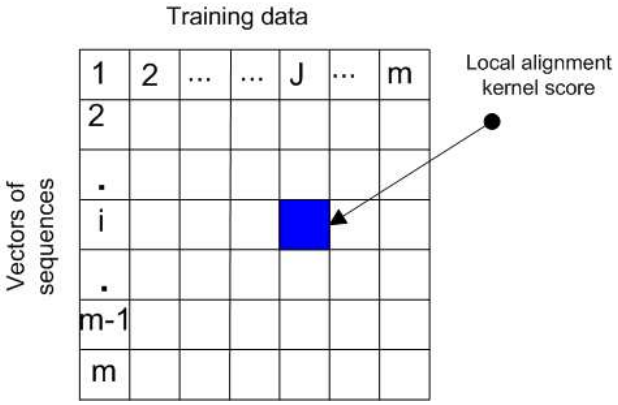

Fig. 3. Calculating the kernel matrix of the training data.

In subfamily classification we randomly partitioned the data in two non-overlapping sets and used a two-fold cross validation protocol. The training and testing was carried out twice using one set for training and the other one for testing. The prediction quality was then evaluated by Accuracy (ACC), Matthew's correlation coefficient (MCC), overall Accuracy ($\overline{ACC}$)and overall MCC ($\overline{MCC}$) as follows:

$$ACC = \frac{TP + TN}{(TN + FN + TP + FP)} \tag{9}$$

$$MCC = \frac{TP \times TN - FN \times FP}{\sqrt{(TN + FN)(TP + FN)(TN + FP)(TP + FP)}} \tag{10}$$

$$\overline{ACC} = \sum_{i=1}^{N} \frac{ACC(i)}{N} \tag{11}$$

$$\overline{MCC} = \sum_{i=1}^{N} \frac{MCC(i)}{N} \tag{12}$$

(TP = true positive, TN = true negative, FP = false positive , FN = false negative, N=number of subfamily or sub-subfamily)

In our study, we used the Bioinformatics Toolbox of MATLAB to create the HMM profiles of families and the SVMlight package [23], to perform SVM training and classification.

Tables 1, 2 and 3 show the results of subfamily classification for classes A,B and C of GPCRs. We see that even when the number of sequences is low, the accuracy of our method is high. The overall accuracy for families A, B and C is 98.94%, 99.94% and 96.95%, respectively, and overall MCC for families A, B and C is 0.98, 0.99 and 0.91, respectively. The results show that almost all of the subfamilies are accurately predicted with our method.

Table 1. The performance of our method in GPCRs subfamily classification (Class A).

Class A subfamilies	Accuracy (%)	MCC
Amine	99.9	0.99
Peptide	97.8	0.97
Hormone protein	100.0	1.00
(Rhod)opsin	99.6	0.99
Olfactory	99.9	0.99
Prostanoid	99.9	.98
Nucleotide-like	100.0	1.00
Cannabinoid	100.0	1.00
Platelet activating factor	100.0	1.00
Gonadotropin-releasing hormone	100.0	1.00
Thyrotropin-releasing hormone	100.0	1.00
Melatonin	100.0	1.00
Viral	87.0	0.8
Lysosphingolipid	100.0	1.00
Leukotriene	100.0	1.00
Overall	98.4	0.98

Class B subfamilies	Accuracy (%)	MCC
Calcitonin	100.0	1.00
Corticotropin releasing factor	100.0	1.00
Glucagon	100.0	1.00
Growth hormone-releasing hormone	100.0	1.00
Parathyroid hormone	100.0	1.00
PACAP	100.0	1.00
Secretin	100.0	1.00
Vasoactive intestinal polypeptide	100.0	1.00
Diuretic hormone	99.1	0.91
EMR1	100.0	1.00
Latrophilin	100.0	1.00
Brain-specific angiogenesis inhibitor	100.0	1.00
Methuselah-like proteins (MTH)	100.0	1.00
Cadherin EGF LAG (CELSR)	100.0	1.00
Overall	≈ 100	0.99

Table 2. The performance of our method in GPCRs subfamily classification (Class B).

Class C subfamilies	Accuracy (%)	MCC
Metabotropic glutamate	92.1	0.84
Calcium-sensing like	94.2	0.82
Putative pheromone receptors	98.7	0.93
GABA-B	100.0	1.00
Orphan GPRC5	97.1	0.96
Orphan GPRC6	100.0	1.00
Taste receptors (T1R)	97.2	0.81
Overall	96.95	0.91

Table 3. The performance of our method in GPCRs subfamily classification (Class C).

For sub-subfamily classification we used 5-fold cross validation. Table 4 shows the results for the sub-subfamily level. We see that in this level also the accuracy is high and we could classify most of GPCRs sub-subfamilies. We could obtain an overall accuracy of 97.93% and a MCC of 0.95 for all sub-subfamilies. At this level we could increase the accuracy, especially when the number of sequences in the positive training data was less than 10, and there was no example in which with our oversampling method the accuracy decreases. Table 5 shows the result of classification in some sub-subfamilies that we used only DEC (different error cost) compared with DEC along with the SPSO method. We tried to find optimum value for both rate of oversampling and error costs. We used the numbers to show the level of family, subfamily and sub-subfamily. For example 001-001-002 means the sub-subfamily Adrenoceptors that belongs to subfamily of Amine (001-001) and class A (001) (as shown in Fig. 2). We see that with

our method the MCC in general increases and, especially when the number of sequences is low, the efficiency of our method is apparent.

Table 4. The performance of our method in GPCRs sub-subfamily classification for Class A,B and C.

Class A	subfamilies	Overall Accuracy (%)	Overall MCC
	Amine	97.1	0.91
	Peptide	99.9	0.93
	Hormone protein	100.1	1.00
	(Rhod)opsin	96.6	0.95
	Olfactory	98.9	0.92
	Prostanoid	98.0	0.94
	Gonadotropin-releasing hormone	96.1	0.93
	Thyrotropin-releasing hormone	91.2	0.94
	Lysosphingolipid	98.4	1.00
Class B	Latrophilin	100.0	1.00
Class C	Metabotropic glutamate	98.1	0.96
	Calcium-sensing like	97.2	0.93
	GABA-B	100.0	1.00
Overall		97.93	0.95

5 Discussion and conclusion

GPCR family classification enables us to find the specificity for ligand that binds to the receptor and also to predict the function of GPCRs. Our aim in this study was to develop an accurate method for classification of GPCRs at the sub-subfamily level, at which we have the problem of imbalanced data. We chose a local alignment kernel(LA kernel) as suitable kernel for our classification task. Compared with HMMs, the LA kernel takes more time during the training phase, but according to results of other researchers, the accuracy of discriminative methods with that kernel is higher than with a generative method like HMMs [8–10]. To solve the problem of imbalanced data we used the SPSO algorithm that can be used along with DEC (different error cost). It makes the classifier less sensitive to noise (here negative data) and increases its sensitivity. Based on our experiments (not showed here) in classifying sub-subfamilies of a subfamily, we get more accurate results if we select all other sub-subfamilies as negative data rather than only sequences in that subfamily, despite the fact that the learning step of the SVM classifier takes more time, because of the higher dimension of the kernel matrix. But the problem of imbalanced data in this case is severe and we tried to solve it with DEC along with the SPSO algorithm. Our study shows again that a discriminative approach for protein classification of GPCRs is more accurate than a generative approach. At the subfamily level we compared our method with that of Bhasin et al. [21]. They used an SVM-based

method with dipeptide composition of protein sequences as input. The accuracy and MCC values of our method outperform theirs. For example in classification of subfamily A, the overall accuracy and MCC of their method were 97.3% and 0.97 but ours are 98.4% and .98, respectively. They did a comparison with other previously published methods like that of Karchin et al. [8] and showed that their method outperformed the others. To the best of our knowledge there is only one study which has been done for sub-subfamily classification [13]. Their approach is based on bagging a classification tree and they achieved 82.4% accuracy for sub-subfamily classification, which is less accurate than ours (97.93% with MCC of 0.95) despite the fact that they had excluded families with less than 10 sequences (we only excluded families with less than 4 sequences).

Table 5. The result of sub-subfamily classification with and without SPSO oversampling for subfamilies of Peptide(Class A).

		DEC		DEC+SPSO	
sub-subfamily	Number of sequence	Accuracy(%)	MCC	Accuracy(%)	MCC
001-002-002	17	99.7	0.81	99.9	0.97
001-002-003	19	99.9	0.94	100.0	1.00
001-002-005	12	99.9	0.91	100.0	1.00
001-002-021	20	99.8	0.66	99.9	0.91
001-002-024	4	99.7	0.38	100.0	1.00
001-002-025	5	99.9	0.79	100.0	1.00

References

[1] T.K Attwood, M. D. R Croning and A. Gaulton. Deriving structural and functional insights from a ligand-based hierarchical classification of G-protein coupled receptors. Protein Eng 15:7-12, 2002.

[2] T. E. Herbert and M. Bouvier. Structural and functional aspects of G protein-coupled receptor oligomerization. Biochem Cell Biol , 76:1-11, 1998.

[3] F. Horn, E. Bettler, L. Oliveira L, F. Campagne, F. E. Cohhen and G. Vriend. GPCRDB information system for G protein-coupled receptors. Nucleic Acids Res. 31(1):294-297,2003.

[4] S. F. Altschul, T. L. Madden, A. A. Schaffer, Z. Zhang, W. Miller W and D. J. Lipman. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. Nucleaic Acids Res. 25:3389-3402, 1997.

[5] 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):767775, 2000.

[6] D. W. Elrod and K. C. Chou. A study on the correlation of G-protein-coupled receptor types with amino acid composition. Protein Eng., 15, 713715, 2002.

[7] B. Qian, O. S. Soyer and R. R. Neubig. Depicting a protein's two faces: GPCR classification by phylogenetic tree-based HMM. FEBS Lett.554, 95, 2003.

[8] R. Karchin, K. Karplus, and D. Haussler. Classifying G-protein coupled receptors with support vector machines. Bioinformatics, 18(1):147159, 2002.
[9] T. Jaakkola,M. Diekhans, and D. Haussler. A discriminative framework for detecting remote protein homologies. Journal of Computational Biology, 7(1-2):95114, 2000.
[10] C. Leslie, E. Eskin, and W. S. Noble. The spectrum kernel: A string kernel for SVM protein classification. In R. B. Altman, A. K. Dunker, L. Hunter, K. Lauderdale, and T. E. Klein, editors, Proceedings of the Pacific Symposium on Biocomputing, pages 564575, New Jersey, World Scientific, 2002.
[11] C.Leslie, E. Eskin, A. Cohen, J. Weston, and W.S. Noble. Mismatch string kernel for SVM protein classification. Advances in Neural Information Processing System 15, pages 1441-1448, 2003.
[12] J.-P.Vert, H. Saigo, and T.Akustu. Convolution and local alignment kernel. In B. Schölkopf, K. Tsuda, and J.-P.Vert(Eds.), Kernel Methods in Compuatational Biology. The MIT Press.
[13] Y. Huang, J. Cai, Y. D. Li, Classifying G-protein coupled receptors with bagging classification tree. Computationa Biology and Chemistry 28:275-280, 2004.
[14] A. Bairoch, R. Apweiler. The SWISS-PROT protein sequence data bank and its supplement TrEMBL. Nucleaic Acids res. 29, 346-349, 2001.
[15] H.saigo, J. P. Vert,N. Ueda and T.akustu. Protein homology detection using string alignment kernels. Bioinformatics, 20(11): 1682-1689, 2004
[16] D. Haussler. Convolution kernels on discrete structures. Technical Report UCSC-CRL-99-10, Department of Computer Science, University of California at Santa Cruz, 1999
[17] M. Pazzini, C. Marz, P. Murphi, K. Ali, T. Hume and C. Bruk. Reducing misclassification costs. In proceedings of the Eleventh International Conference on Machine Learning, 217-225, 1994
[18] N. Japkowicz, C.Myers and M. Gluch. A novelty detection approach to classification. In Proceeding of the Fourteenth International Joint Conference on Artificial Intelilligence, 10-15, 1995.
[19] N. Japkowicz. Learning from imbalanved data sets:A Comparison of various strategies. In Proceedings of Learning from Imbalanced Data, 10-15, 2000.
[20] K. Veropoulos, C. Campbell and N. Cristianini. Controlling the sensitivity of support vector machines. Proceedings of the International Joint Conference on AI, 55-60, 1999.
[21] M. Bhasin and G. P. S. Raghava. GPCRpred: an SVM-based method for prediction of families and subfamilies of G-protein coupled receptors. Nucleaic Acids res. 32, 383-389,2004.
[22] J. D. Thompson and D. G. Higgins, and T. J. Gibson. CLUSTALW: improving the sesitivity of progressive multiple sequence alignment through sequence weighting, positions-specific gap penalties and weight matrix choice. Nucleic Acids Res. 22, 4673-4680, 1994.
[23] T. Joachims. Macking large scale svm learning practical. Technical Report LS8-24, Universitat Dortmond, 1998.
[24] M. Beigi and A. Zell. SPSO: Synthetic Protein Sequence Oversampling for imbalanced protein data and remote homilogy detection. VII international symposium on Biological and Medical Data Analysis ISBMDA, 2006.