A BETTER GAP PENALTY FOR PAIRWISE SVM

HON NIAN CHUA
NUS Graduate School for Integrative Sciences and Engineering,
Singapore 117597

WING-KIN SUNG
School Of Computing, National University of Singapore, Singapore 119260

SVM-Pairwise was a major breakthrough in remote homology detection techniques, significantly outperforming previous approaches. This approach has been extensively evaluated and cited by later works, and is frequently taken as a benchmark. No known work however, has examined the gap penalty model employed by SVM-Pairwise. In this paper, we study in depth the relevance and effectiveness of SVM-Pairwise's gap penalty model with respect to the homology detection task. We have identified some limitations in this model that prevented the SVM-Pairwise algorithm from realizing its full potential and also studied several ways to overcome them. We discovered a more appropriate gap penalty model that significantly improves the performance of SVM-Pairwise.

1 Introduction

With protein sequences readily available, much challenge lies with understanding the functions and the interactions that proteins are involved in. Current techniques in homology detection have achieved encouraging progress but are far from reliable, especially for proteins with diverged evolutionary relationship where sequence similarities are hardly detectable.

Earlier approaches in homology detection made use of pairwise local alignment search algorithms such as the well-known Smith-Waterman algorithm[1] and its efficient heuristic approximations BLAST [2] and FASTA [3]. Homology is inferred based on sequence similarity between an unknown protein and annotated sequences. These methods have proven very useful. Nonetheless, homologous proteins with remote sequence similarity (less than 25% sequence identity [6]) remain elusive. To detect more subtle similarities, later approaches adopted a superfamily approach. Known proteins are first clustered into different families or superfamilies based on their evolutionary origin, and an unknown protein is compared against each superfamily to detect possible similarities. Several schemes for classifying proteins into families and superfamilies have been established, such as SCOP [21], FSSP [22] and CATH [23]. Techniques that utilized the superfamily concept generally adopted two approaches: generative and discriminative.

Generative techniques construct a statistical model for each protein family from sequences belonging to that family. The probability that an unknown protein belongs to the family is inferred by its similarity to this model. Generative approaches have been shown to be able to infer three times more homologies than simple pairwise alignment[12]. Examples of generative approaches include Position Specific Scoring Matrices (also known as Profiles[7]) and Hidden Markov Models (HMM) [5], and are used by many popular tools such as PFam[10], PROSITE[9], E-MOTIF[8] and eBlocks[11]. Iterative methods such as SAM[5] and PSI-BLAST[4] improve upon the sensitivity of generative approaches by iteratively updating the model with discovered homologues.

Discriminative techniques, on the other hand, try to find features in family members(positive examples) that best distinguishes them from non-members (negative examples). While generative approaches consider only positive examples, discriminative approaches consider both positive and negative examples. Discriminative methods such as Fisher-SVM[13] and SVM-Pairwise[15] that combine Support Vector Machines(SVM) with sequence similarity performed relatively well. SVM takes in a fixed-length feature vector for each training example that models its characteristics. SVM then transforms these vectors using a kernel and finds a hyperplane that best separates transformed feature vectors of positive examples from those of negative ones. A similar transformed feature vector is derived from each test example and it is classified as positive or negative based on which side of the hyperplane this vector resides.

SVM-Pairwise significantly outperforms all preceding methods and is often used by later approaches as a benchmark for performance evaluation. Its edge over previous approaches stems from its inclusion of negative examples during training, its ability to detect motif or domain-sized similarities even when overall sequence similarity is low[15], as well as the inclusion of unrelated dimensions in the feature vector. Subsequent approaches largely adopted the discriminative framework [16-20]. Some studies proposed to find more concise local structural information by using the presence/absence of motifs to derive feature vectors. [17] used motifs from the eBlocks database while SVM-I-Sites[16] used structural motifs from the I-Sites database. Such methods have been shown to perform well, reinforcing the significance of motif-sized similarity. Other works explored new similarity metrics in place of local alignment scores [18-20].

Although SVM-Pairwise has been extensively evaluated and studied, no known work has studied the gap penalty model and parameters that it employs. SVM-Pairwise's improvement over Fisher-SVM depended largely upon its use of local alignment algorithms. Since gap penalty has a fundamental effect on alignment algorithms, the use of an appropriate gap penalty model can be vital to the performance of SVM-Pairwise. In this paper, we study how SVM-Pairwise derived the gap penalty model for its local alignment algorithm and how well this model performs for homology detection. We discovered that with the original gap penalty model, only a single motif-sized local similarity is captured between more distant homologues. To realize the full potential of Pairwise-SVM, we need to consider all possible motif-sized local similarities so that domain-sized similarities can be detected. We proposed some new algorithms and investigated more appropriate gap penalty models. We discovered that simply using a more appropriate gap penalty scheme can significantly improve the performance of SVM-Pairwise.

2 SVM-Pairwise

SVM-Pairwise uses the pairwise local alignment score between a protein and every protein in the dataset to form a feature vector for SVM training. Specifically, the Smith-Waterman algorithm is used to compute the local alignment score. The default parameters for protein sequence alignment are used. That is, an affine gap penalty with gap initiation penalty 11 and gap extension penalty 1, as well as the BLOSUM62 substitution matrix. SVM-Pairwise employs the GIST SVM classifier from the GIST SVM software package, which is provided by the authors of the SVM-Pairwise[15] paper. The kernel is normalized and transformed into a Radial Basis Function as follows:

$$\hat{K}(X,Y) = e^{-\frac{K(X,X)-2K(X,Y)+K(Y,Y)}{2\sigma^2}} + 1$$

where width σ is the median Euclidean distance (in feature space) from any positive training example to the nearest negative example. The constant 1 is added to the kernel in order to translate the data so that the separating hyperplane passes through the origin. An asymmetric soft margin is implemented by adding to the diagonal of the kernel matrix a value 0.02ρ, where ρ is the fraction of training set sequences that have the same label as the current sequence. The trained SVM model produces a discriminant score that is used to rank the members of the test set [15]. For a dataset of *n* proteins, the algorithm generates *n* length-*n* vectors. For a protein X, its corresponding feature vector will be $F_x = (f_{x1}, f_{x2}, \ldots, f_{xn})$ where f_{xi} is the E-value of the Smith-Waterman score between the x-th sequence and the *i*-th sequence in the dataset. The Smith-Waterman algorithm is a dynamic programming algorithm that finds the optimal local alignment between 2 sequences. Implementations of Smith-Waterman and other local similarity algorithms such as BLAST and FASTA typically use the abovementioned gap penalty because of their satisfactory performance in finding the most significant local alignment. They do not take into account any good local alignments that are not part of the best local alignment. To study the effect of such possible limitations in SVM-Pairwise we examine the alignment of sequences taken from the *Nucleic Acid-Binding proteins* superfamily (SCOP 2.38.4.1) in version 1.53 of the Structural Classification of Proteins (SCOP) database. Figure 1 shows pairwise alignments between *Aspartyl-tRNA synthetase*(SCOP 2.38.4.1.1) and some of its family and superfamily members using the default gap penalty.

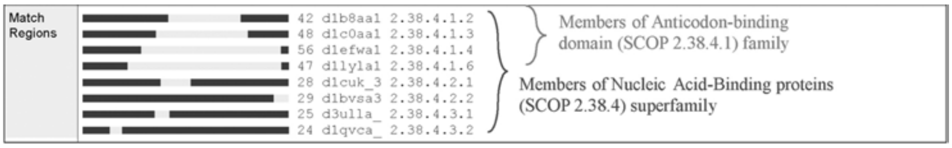

Figure 1. Smith-Waterman Alignments between Aspartyl-tRNA synthetase and members of the Nucleic Acid-Binding proteins superfamily using affine gap penalty with 11 for gap initiation and 1 for gap extension

The aligned regions are indicated by the lighter regions and are shown based on their position in the *Aspartyl-tRNA synthetase* sequence. We can see from the alignments that while the algorithm is able to capture substantial local similarity regions between the sequence and those from its family members (SCOP 2.38.4.1), it does not capture ample local similarity with other superfamily members outside its family. The algorithm is unable to detect domain-sized similarity but rather only motif-sized similarity between these distant homologues. We will see later that distant homologues may have multiple regions of short (motif-sized) local similarities rather then a single substantial (domain-sized) region of local similarity.

3 Multiple local similarity

3.1 Recursive Smith-Waterman

To verify our suspicion that there may be multiple significant motif-sized local similarity regions between remote homologues that escape the detection of SVM-Pairwise, we modified the Smith-Waterman algorithm to recursively capture all significant alignments longer then a user-defined minimum length ε. We will refer to this new algorithm as Recursive SW. Refer to Figure 2 for an illustration of the algorithm.

```
RecursiveSW ( X , Y )
{
        $m$ = length of $X$;  $n$ = length of $Y$;
        Compute the alignment using the Smith-Waterman algorithm. Let S be the alignment
        score and l be the length of matches
        If ($S = 0$ or $l < \varepsilon$) return 0;
        If ($b_x > \varepsilon$ and $b_y > \varepsilon$) $S$ += RecursiveSW ( $X(0, b_x - 1)$ , $Y(0, b_y - 1)$ );
        If ($m - e_y > \varepsilon$ and $n - e_y > \varepsilon$)
                $S$ += RecursiveSW ( $X(e_x + 1 , m - 1)$ , $Y(e_y + 1 , n - 1)$ ) + $h$;
        Return $S$;
}
```

Figure 2. The Recursive SW algorithm.

Given two protein sequences X and Y of length m and n respectively, the algorithm first finds the best local alignment between the two sequence, $X(b_x , e_x)$ and $Y(b_y , e_y)$, where b_i and e_i are the beginning and ending indices of the alignment in the i-th sequence respectively. If the Smith-Waterman score of the alignment is 0 (local alignment scores are never negative) or if the number of aligned residues is less then ε, the score is set to 0 and returned. This is the termination condition of the recursive function. If the condition is not met, the function will recursively call itself to find the best alignment between subsequences not included in the alignment. To preserve sequential ordering of the local alignments, we restrict the recursive alignments to $X(0, b_x - 1)$ with $Y(0, b_y - 1)$ and $X(e_x + 1 , m - 1)$ with $Y(e_y + 1 , n - 1)$. The score from the recursive call is added to the current score. The default gap penalty is used and the gap initiation penalty h is imposed for each recursion. In Figure 3, the new algorithm is used to align the same set of sequences from the Nucleic Acid-Binding proteins superfamily as before to illustrate the improvement in the coverage of local similarities. The alignments showed that Recursive SW captured multiple local similarities between the homologues, affirming our earlier speculation that the Smith-Waterman with default parameters overlooked possible significant similarities.

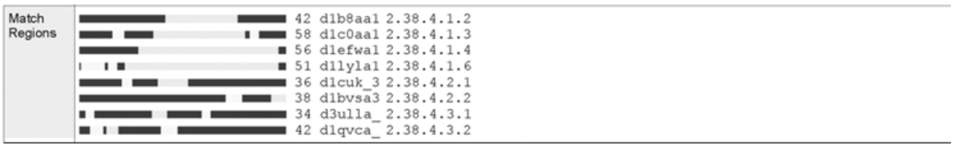

Figure 3. Recursive Smith-Waterman Alignments between Aspartyl-tRNA synthetase and members of the Nucleic Acid-Binding proteins superfamily using affine gap penalty with 11 for initiation and 1 for extension

The Recursive SW assumes a sequential ordering of these motif-sized similarities. To study whether it is significant, we designed another algorithm that allows discovery of non-sequential local similarities. This algorithm performs an initial Smith-Waterman alignment, then concatenates unaligned regions and aligns them again. This is repeated until the alignment is shorter than a minimum length ε. We refer to this algorithm as Non-Sequential Recursive SW.

3.2 Experimental Setup

We evaluate the performance of the new algorithms using sequences from version 1.53 of the SCOP database selected with the Astral database[24] such that the E-value of sequence similarity among sequences are above 10^{-25}. The resulting dataset contains 4352 distinct sequences, grouped into families and superfamilies. For each family, family members are used as test examples while superfamily members that are not in the family are used as training examples. The data set comprises 54 families with at least 10 family members and 5 superfamily members not in the family. Protein domains that do not belong to the superfamily are considered negative examples and are randomly split into training and testing sets in the same ratio as the positive examples. We used raw Smith-Waterman alignment scores in the vectorization step of the SVM-Pairwise method as Karlin-Altschul statistics may not be appropriate for this application. We shall explain this in detail in our journal publication.

To compare the relative performance of different algorithms, we use the Receiver Operating Characteristic (ROC)[25] score, which is the area under the curve derived from plotting true positives as a function of false positives for various thresholds. A higher ROC score indicates a better classifier and the perfect classifier has an ROC score of 1. The dataset is classified using the SVM-Pairwise framework in 3 different setups. The first uses a Smith-Waterman implementation for the vectorization step, with the default gap penalty. The other two employs Recursive SW and Non-Sequential Recursive SW respectively in place of Smith-Waterman in the vectorization step with the same gap penalty and a value of 10 for ε. Figure 4 is obtained by plotting the total number of families for which each method obtains an ROC score that exceeds or equals some threshold h where $h \in [0..1]$. The curve of the setup using Recursive SW dominates that using Smith-Waterman, indicating that it is a better classifier. This reinforced our idea that capturing a more complete local similarity between two homologues can better reflect their relationship. We also observed that the classifier using Recursive SW performed better than that using Non-Sequential Recursive SW. This indicates that the sequential ordering of local similarities may be significant in homology detection.

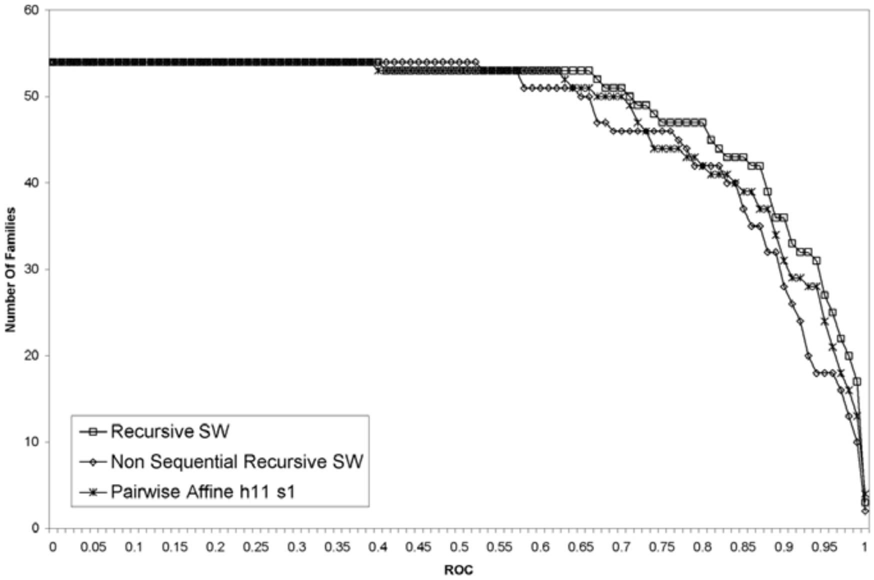

Figure 4. Number of families with ROC scores equals or exceeding different thresholds scores

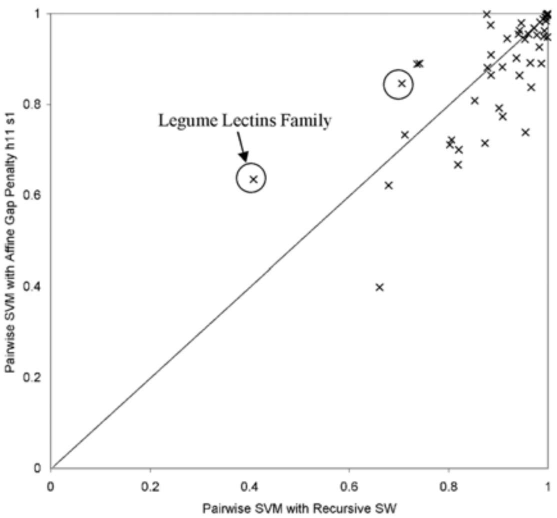

Figure 5. 2D plot of the ROC scores for Pairwise-SVM using Smith-Waterman against Pairwise-SVM using Recursive SW

Figure 5 illustrates a 2D-plot that compares the relative ROC scores for each method family-by-family. It is observed that while Recursive SW performs better then Smith-Waterman for most families, it performs unsatisfactorily for a handful of families (indicated by the arrows). Among the worst is the *Legume lectins* family (SCOP 2.38.1.1) as indicated by the circle in Figure 5.

To see why Recursive SW classified these families so badly, we examine the classification results of the *Legume Lectins* family (SCOP 2.28.1.1). We study one of the positive test examples, *West-central African Legume* (SCOP 2.38.1.1.5) in which the algorithm erroneously classified them as unlikely to belong to the *ConA-like lectins/glucanases* superfamily (SCOP 2.38.1) (it was given a very low discriminant score). Figure 6 shows the alignment of *West-central African Legume* with all the 24 training examples for the *ConA-like lectins/glucanases* superfamily using Recursive SW. The alignments revealed that the algorithm detected very little similarity between them. Using other members of the family yielded similar observations. This lead us to speculate that there may be more subtle similarity between remote homologues that may contain frequent non-contiguous gaps. Such similarities would be undetected by the harsh gap initiation penalty. Another problem with Recursive SVM is that it does not take into account the length of gaps between any two local alignments (the gap initiation penalty is imposed for every recursion independent of the gap length).

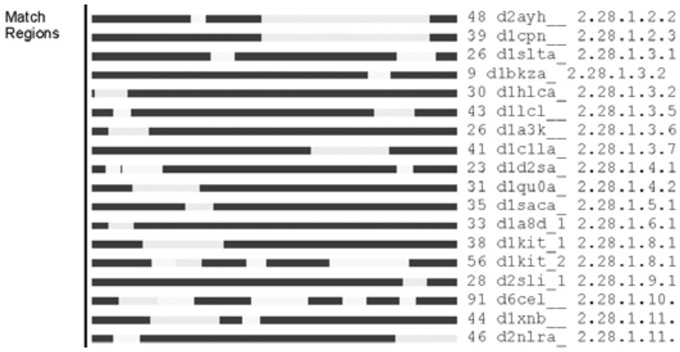

Figure 6. Recursive SW Alignments between *West-central African Legume* (SCOP 2.38.1.1.5) and all 18 training examples from the *ConA-like lectins/glucanases* superfamily (SCOP 2.38.1)

4 Relaxed Gap Penalty

We have seen that despite its relatively better performance, the Recursive SVM algorithm has some possible pitfalls. Based on the above study, we need to find an approach that can capture multiple sequential local motif-sized similarities between any pair of protein sequences. The approach should also allow more general alignment with multiple gaps, and is sensitive to the length of gaps between motif-sized similarities. If we take such gap lengths into consideration, motif-sized local similarities that are too far apart from each other may no longer be significant as a group. Hence we do not have to capture all possible local similarities but only capture those that are relatively close to each other. The most straight forward way to achieve all these characteristics is to use Smith-Waterman algorithm with a more relaxed gap penalty.

We examine the effect of relaxing the gap initiation penalty for the affine gap model to be 4 while retaining the gap extension penalty of 1. Using the same set of sequences as in Figure 4, we examine the effect on the alignment of these sequences. The alignment results are shown in Figure 7. From the alignments, we can see that the more relaxed penalty allows for more general similarity with frequent short gaps and at the same time provides a comprehensive coverage of motif-sized similarities. We observed that the new gap penalty model can capture most of the motif-sized similarities that Recursive SW was able to discover. At the same time, the sequential order of these motif-sized similarities is required for successful alignment. The effect of varying length of gaps between motif-sized similarities is also taken into consideration since the alignment will be penalized according to the gap length.

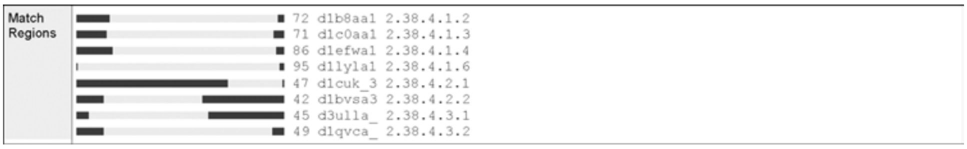

Figure 7. Recursive Smith-Waterman Alignments between *Aspartyl-tRNA synthetase* and members of the *Nucleic Acid-Binding* proteins superfamily using affine gap penalty with 4 for gap initiation and 1 for gap extension

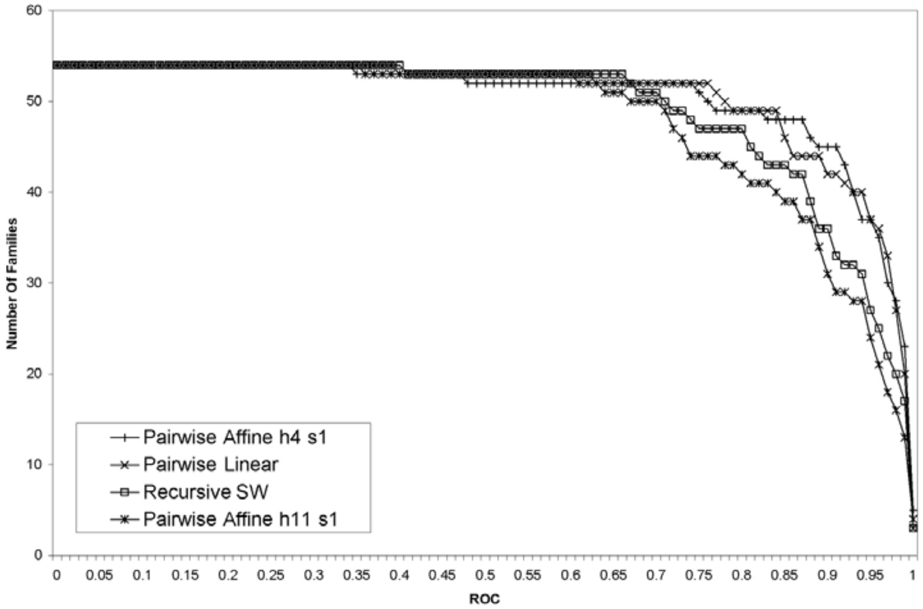

Figure 8. Number of families with ROC scores equals or exceeding different thresholds scores

The same experimental setup in Section 3.2 is used to evaluate the relative performance of the relaxed gap model when used for homology detection. The original Pairwise-SVM is again used as the baseline model. To illustrate the significance of a less restrictive gap penalty, we also run one set of experiment using a simple linear gap penalty model with a gap penalty of 4. Figure 8 is obtained by plotting the total number of families for which each method obtains an ROC score that exceeds or equals some threshold *h* where $h \in [0..1]$. We can see a significant improvement in the classification performance of SVM-Pairwise when a more relaxed gap penalty model is used. A family-to-family comparison with the original SVM-Pairwise method (refer to Figure 10) reveals that both relaxed penalty models can achieve an equal or better prediction then the original gap penalty model for most families. For the linear model, the boost in improvement comes with a bonus – reduction in memory requirement and running time. The linear gap penalty model requires only 1 dynamic programming table compared to 3 for affine gap penalty, reducing memory requirement by a factor of 3. Complexity is also reduced by a factor of 3 since fewer tables are updated, speeding up the execution time.

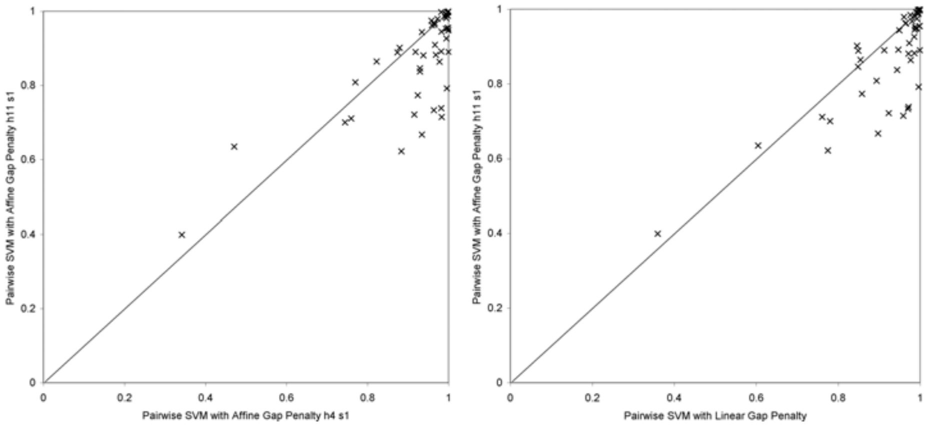

Figure 10. ROC 2D-Plot between SVM-Pairwise and SVM-Pairwise using Affine Penalty of 4 for gap initiation and 1 for gap extension (left) and between SVM-Pairwise and SVM-Pairwise with Linear Penalty (Right)

5 Conclusion

We have studied the gap penalty model used by the SVM-Pairwise method in detail and discovered several limitations, namely that it fails to detect multiple localized motif-sized similarities and it does not capture more subtle similarities with frequent gaps. We also studied several approaches to improve the performance of SVM-Pairwise by rectifying these limitations. Through these studies, we have affirmed our speculation that a more complete similarity assessment between any two sequences should consider multiple motif-sized local similarities that are in sequential order. This also implies that the ordering of motifs in a protein sequence may significantly affect its function. Among the approaches studied, we have found that using a relaxed affine gap penalty with a gap initiation penalty of 4 and a gap extension penalty of 1 works well for the most number of families studied.

References

1. T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *J. Mol. Biol.* 147:195—197, 1981.
2. S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. A basic local alignment search tool. *J. Mol. Biol.* 215:403—410, 1990.
3. W. R. Pearson. Rapid and sensitive sequence comparisions with FASTP and FASTA. *Methods in Enzymology, 183:63—98, 1985.*
4. S. F. Altschul et al. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucl. Acids Res.,* 25, 3389—3402, 1997.
5. K. Karplus, C. Barrett, and R. Hughey. Hidden markov models for detecting remote protein homologies. *Bioinformatics*, 14(10):846—856, 1998.
6. S. Y. Chung and S. Subbiah. A structural explanation for the twilight zone of protein sequence homology. *Structure,* 4(10):1123—1127, 1996.
7. M. Gribskov, A. D. McLachlan, and D. Eisenberg. Profile analysis: detection of distantly related proteins, *Proc. Natl. Acad. Sci. USA*, 84, 4355—4358, 1987.
8. J. Huang and D. Brutlag. The E-Motif Database. *Nucl. Acids Res.,* 29(1),202—204

9. A. Bairoch. PROSITE: A dictionary of sites and patterns in proteins. *Nucl. Acids Res.,* 20:2013—2018, 1992.
10. A. Bateman et al. The Pfam Protein Families Database. *Nucl. Acids Res.,* 30(1) 276—280, 2002.
11. Q. Su et al. eBLOCKS: an automated database of protein conserved regions maximizing sensitivity and specificity *http://motif.stanford.edu/eblocks,* 2003.
12. M. P. S. Brown et al. Knowledge-based analysis of microarray gene expression data using support vector machines. *Proc. Natl. Acad. Sci. USA,* 97(1):262—267, 2000
13. T. Jaakkola, M. Diekhans, and D. Haussler. A discriminative framework for detecting remote homologies. *J. Comput. Biol.,* 7(1-2):95—11, 2000.
14. W. N. Grundy. Family-based homology detection via pairwise sequence comparison. *RECOMB 98, ACM Press,* 1998.
15. L. Liao and W. S. Noble. Combining pairwise sequence similarity and support vector machines for detecting remote protein evolutionary and structural relationships. *J. Comp. Biol.,* 2003.
16. Y.Hou, W. Hsu, M. L. Lee, and C. Bystroff. Efficient Remote Homology Detection Using Local Structure. *Bioinformatic,s* 19(17): 2294—2301, 2003.
17. A. Ben-Hur and D. Brutlag. Remote homology detection: a motif based approach. *Bioinformatics,* 19 Suppl. 1: i26—i33, 2003.
18. C. Leslie, E. Ekin, W. S. Noble. The Spectrum Kernel for SVM protein classification. *Proc. Pacific Symposium on Biocomputing*, 564—575, 2002
19. C. Leslie, E. Ekin, J. Weston, W. S. Noble. Mismatch String Kernels for SVM protein classification. *Neural Information Processing System 15*, 2002.
20. H. Saigo, J. Vert, N. Ueda, T. Akutsu. Protein homology detection using string alignment kernels. *Bioinformatics*, 20(1682—1689), 2004.
21. A. G. Murzin et al. SCOP: A structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.,* 247:536—540, 1995.
22. L. Holm et al. Mapping the protein universe. *Science,* 273, 595—603, 1996.
23. C. S. Orengo et al. CATH–A Hierarchic Classification of Protein Domain Structures. *Structure,* 5(8):1093—1108, 1997.
24. S. E. Brenner, P. Koehl, and M. Levitt. The ASTRAL compendium for sequence and structure analysis. *Nucl. Acids Res.,* 28:254—256, 2000.
25. M. Gribskov and N. L. Robinson. Use of receiver operating characteristic analysis to evaluate sequence matching. *Computers and Chemistry,* 20(1):25—33, 1996.