# Gene Ontology Prediction Using Compression Based Distances and Alignment Scores on Both Amino Acid Sequence and Secondary Structure

*Aslı Filiz*[1], *Zehra Çataltepe*[2]
***İstanbul Teknik Üniversitesi*** [1]***Bilgisayar Bilimleri Programı,*** [2]***Bilgisayar Mühendisliği Bölümü***
**Maslak, İstanbul, Türkiye**
{filizas,cataltepe}@itu.edu.tr

## Abstract

*Normalized Compression Distance (NCD) is a compression based pairwise distance measure. NCD has been shown to perform well in different domains, such as music, biological sequence and text classification. In this study, we use NCD distance together with Smith-Waterman (SW) alignment scores of protein sequences for gene ontology prediction. We find out that, using secondary structure in addition to the amino acid sequence increases the prediction performance when using NCD or SW alignment scores alone. The best contribution ratio of secondary structure for SW alignment scores is 0.25, while it is 0.50 for NCD scores. We also investigate using both NCD and SW together with the amino acid and secondary structure. We find out that this combination results in better prediction than NCD alone, but worse prediction than SW alone.*

*Keywords: Normalized Compression Distance, Smith-Waterman alignment score, amino acid sequence, secondary structure, Gene Ontology.*

## 1. Introduction

Protein function prediction is one of the most important and difficult problems in bioinformatics.

Usually, alignment scores between amino acid or secondary structure sequences are used to predict protein function. One of the most frequently used alignment algorithms is the Smith-Waterman (SW) alignment which is a local alignment algorithm suitable for detecting remote protein similarities. The normalized compression distance (NCD) is another measure of distance that can be used between protein sequences as well as other kinds of data, such as music, text or images. SW alignment scores and NCD have already been used for function prediction and it has been shown that NCD performs worse than SW alignment, while combination of NCD and alignment scores outperforms alignment scores only [Kocsor et.al. 2005].

In this study, secondary structure is incorporated into SW alignment scores and NCD scores and it is shown that incorporating secondary structure improves function prediction. It is found out that, unlike [Kocsor *et al.*, 2005] where combination of SW and NCD on amino acid sequences was found to help for classification, when using secondary structure in addition to amino acid sequence, using SW and NCD together does not give better results than SW incorporating secondary structure.

The rest of the paper is organized as follows. Section 2 summarizes the previous work on protein function prediction and NCD. Section 3 introduces SW and NCD and how they are used when both amino acid sequence and secondary structure are available. Section 4 includes information about the dataset used in the experiments. Section 5 summarizes the experimental results. The paper ends with the conclusions in Section 6.

## 2. Previous Work

Especially with the increasing amount of proteins whose sequences are known but functions not known, automated function prediction have gained more importance. The Gene Ontology is one of the most frequently used definitions of protein function. In January 2008 GO contains over 24,500 annotations which include the GO ID, a unique alphanumerical string, the common name and the definition of the protein. GO provides three first level branches: biological process, cellular component and molecular function. The Gene Ontology Annotation [Camon et al., 2004] database provides annotations for proteins of the UniProt Knowledgebase [Butler, 2002] using the Gene Ontology (GO). GOA includes many organisms such as human, mouse, rat, arabidopsis, zebra fish, chicken and cow.

A protein can be represented in four different levels: the amino acid sequence, secondary structure, tertiary structure and quaternary structure. All of these representations are used for bioinformatics applications. The Protein Data Bank (PDB) [Berman et. al, 2000] is an online storage for the three-dimensional structures of proteins, nucleic acids and protein-nucleic acid complexes. The PDB contained 50,480 structures in April 2008. For each structure, sequence details, atomic coordinates, crystallization conditions, 3-D structure neighbors computed using various methods, derived geometric data, structure factors, 3-D images and a variety of links to other resources are available in PDB.

Because they are less costly to evaluate and hence available more, the most frequently used protein representations are the amino acid sequence and then the secondary structure. Amino acid sequences have been used by numerous people for protein fold recognition, while secondary structure has been used in addition to the amino acid sequence, for example, by [Wallqvist *et al.* 2000], [Cheng and Baldi 2006].

Alignment-based methods such as Smith-Waterman [Smith and Waterman, 1981] or Needleman-Wunsch [Needleman and Wunsch, 1970] are frequently used for bioinformatics applications. Since it can detect partial matchings between sequences, Smith Waterman alignments can detect remote protein similarities and hence is more useful for function prediction. Previously [Liao and Noble, 2003] built pairs of sequences in the data set and obtained pairwise alignment scores by aligning these. They showed that using the pairwise alignment scores as features to input to support vector machine classifiers is a straight-forward method that outperforms many previous work, e.g. the SVM-Fisher method [Jaakkola *et al.,* 2000], the PSI-BLAST algorithm [Altschul *et al.,* 1997], SAM [Krogh *et al.,* 1994] and FPS [Grundy, 1998], especially when working with large data sets.

Both amino acid sequence and secondary structure were used simultaneously by [Wallqvist *et. al.,* 2000] where they showed that secondary structure improves fold recognition. Similarly, based on Wallqvist's amino acid and secondary structure combination idea, [Aygün *et al.,* 2008a] showed that secondary structure can help with gene ontology prediction.

The normalized compression distance (NCD) is another measure of distance which is shown to perform quite well in different domains. [Keogh *et al.,* 2004] present a successful application in pattern recognition, [Cilibrasi *et al.,* 2004] and [Cataltepe *et al.,* 2006] used NCD in music domain for composer and genre classification. Cilibrasi and Vitanyi [Cilibrasi and Vitanyi, 2005] provide examples of successful uses of NCD in many areas. Different distance metrics that use compression are compared in [Sculley and Brodley, 2006]. These metrics are called Chen-Li metric (CLM), the compression-based dissimilarity measure (CDM), compression-based cosine (CosS). Sculley and Brodley show that NCD outperforms all the other metrics. [Nevill-Manning and Witten, 1999] argued that proteins cannot be compressed which was answered by [Hategan and Tabus, 2004] stating that proteins can be compressed using appropriate compression algorithms. Use of LZ78 algorithm for compressing proteins was suggested by [Freschi and Bogliolo, 2005]. The success of NCD was shown in bioinformatics by [Li and Vitanyi, 1997] and [Li *et al.,* 2001]. [Ferragina *et al.,* 2007] provides another use of NCD on biological data.

The normalized compression distance was developed by [Cilibrasi and Vitanyi, 2005] based on Kolmogorov complexity which cannot be computed, but only approximated. It is a universal, parameter-free (dis)similarity metric which does not depend on the compressor type used. It computes the distance between two sequences, based on their lengths when they are compressed individually or together.

Success of alignment score and compression based methods for protein sequence classification has been examined and compared with each other in [Kocsor *et al.,* 2005]. They show that using alignment scores only outperforms using NCD only. However, they suggest a new similarity metric which is a combination of alignment scores and compression scores and report that this new combined metric has a better performance than alignment or compression only.

## 3. Measuring Similarity or Distance Between Proteins

### 3.1. Smith-Waterman alignment scores

Smith-Waterman [Smith and Waterman, 1981] is a local and pairwise alignment algorithm. It finds similar regions in longer sequences which do not have to be totally similar and also which may have varying sequence length. Therefore, it is suitable for detecting the similarity of distantly related proteins. Pairwise alignment scores between a pair of proteins, in addition to other measures of similarity between them, have been used in [Cheng and Baldi, 2006].

Alignment scores to all available training sequences have also been used as inputs in [Liao and Noble, 2003] and also in this study. "SVM-pairwise" [Liao and Noble, 2003] takes all sequence pairs in the database and aligns them to each other using the Smith-Waterman local alignment algorithm. This is based on the idea that two proteins belonging to the same class can be aligned similarly to a set of proteins containing both positive and negative instances. Alignment scores are then used as the constant-sized feature vector for a protein. For a training set of N sequences, every protein is aligned to all N sequences, including itself, and it has N features. These features are the input to the classification algorithm. Liao and Noble indicate that this method is not only easy to use, but also superior to similar algorithms due to its low complexity and more accurate because it learns from both positive and negative examples.

Although it is usually used with amino acid sequences, Smith-Waterman algorithm can also align sequences according to their secondary structure. Balign [Aygün and Çataltepe, 2008] produces Smith-Waterman or Needleman-Wunsch alignment scores calculated using both amino acid sequence and secondary structure. The tool allows including secondary structure according to a parameter $\alpha$ chosen by the user between 0 (use amino acid sequence only) and 1 (secondary structure only). The Smith-Waterman alignment score including both the amino acid and the secondary structure is computed according to [Wallqvist *et.al.,* 2000]:

$$\mathrm{SW}_\alpha(\mathrm{p},\mathrm{q}) = \mathrm{SW}(\mathrm{p}_{\mathrm{AA}},\mathrm{q}_{\mathrm{AA}}) + \alpha\,\mathrm{SW}(\mathrm{p}_{\mathrm{SS}},\mathrm{q}_{\mathrm{SS}}) \quad (1)$$

where $p$ and $q$ are the proteins to be aligned, $p_{AA}$ and $q_{AA}$ are the amino acid and the secondary structure sequences for the protein p respectively, $SW(p_{AA},q_{AA})$ is the Smith-Waterman alignment score computed from the amino acid sequences and $SW(p_{SS},q_{SS})$ is the Smith-Waterman alignment score computed from their secondary structure. Gap open and extension penalties are also included in the alignment score.

Balign produces two types of alignment scores, the percent identity and the bit score. The bit score is the sum of the substitution matrix entries for matches minus gap penalties, normalized with respect to the statistical parameters of the scoring system and is therefore comparable between different alignments. In this study we use conservation score which is the normalized version of bit score:

$$cons(x,y) = \frac{bitscore(x,y)}{\max\left(bitscore(x,x), bitscore(y,y)\right)} \quad (1)$$

Conservation score gives a better measure of similarity due to normalization and is therefore preferred to raw Smith-Waterman alignment scores in this study.

### 3.2. Normalized Compression Distance

Normalized compression distance (NCD) is a parameter-free, universal metric for sequence similarity developed by [Cilibrasi and Vitanyi, 2005]. The normalized compression distance, *NCD(x, y),* between two sequences $x$ and $y$, is the normalized version of the compression distance $C(x, y)$ involving the normal compressor $C$ and is defined as follows [Cilibrasi and Vitanyi, 2005]:

$$NCD(x,y) = \frac{C(x,y) - \min\{C(x), C(y)\}}{\max\{C(x), C(y)\}} \quad (2)$$

The NCD does not need any background knowledge about the data set and is also robust because it is defined independently from the compressor type.

NCD can be computed for amino acid sequences or secondary structure sequences separately, using the corresponding strings for $x$ and $y$. Just as in the case of Smith-Waterman alignment scores, in this study, we aim to compute NCD between two proteins using both their amino acid and secondary structure sequences. We achieve this purpose using a linear combination of NCD scores for the amino acid ($NCD(p_{AA},q_{AA})$) and secondary structure ($NCD(p_{SS},q_{SS})$) sequences:

$$\mathrm{NCD}_\beta(\mathrm{p},\mathrm{q}) = (1-\beta).\,\mathrm{NCD}(\mathrm{p}_{\mathrm{AA}},\mathrm{q}_{\mathrm{AA}}) + \beta\,\mathrm{NCD}(\mathrm{p}_{\mathrm{SS}},\mathrm{q}_{\mathrm{SS}}) \quad (3)$$

In this formula $\beta \in [0:1]$ is a parameter similar to α in Equation (1) and it controls the contribution of secondary structure to the NCD score.

### 3.3. Combination of NCD and Smith Waterman scores

In their study on comparison of alignment-based and compression-based classification of proteins, Kocsor *et. al.* report that alignment-based classification outperforms the compression-based classification [Kocsor *et al.* 2005]. On the other hand, combination of alignment and compression scores outperforms both. They suggest using the combination score $F(x, y)$ for the sequences $x$ and $y$ as:

$$F(x,y) = \left(1 - \frac{SW(x,y)}{SW(x,x)}\right) NCD(x,y) \quad (4)$$

This is suggested for amino acid sequences only, and it needs to be extended to incorporate the secondary structure. In equation (1) $SW_\alpha(p,q)$ and in equation (3) $NCD_\beta(p,q)$ show how to incorporate the secondary structure into the alignment and compression scores. Combination of them similar to equation (4) leads to a new combined measure of similarity between two proteins that incorporate both amino acid sequence and secondary structure as well as Smith-Waterman and NCD scores:

$$f_{\alpha\beta}(p,q) = \left(1 - \frac{SW_\alpha(p,q)}{SW_\alpha(p,p)}\right) NCD_\beta(p,q) \quad (5)$$

Upon examination of the NCD and SW scores, we found out that the agreement between SW and NCD scores depends on the sequence length. In order to find out this dependence, we could not use direct comparison

of scores, since one shows similarity and the other distance and their distributions are quite different from each other. Instead the following procedure is applied: First of all, all proteins are binned according to their sequence lengths. SW and NCD amino acid scores of proteins in the same bin are computed. In order to measure the disagreement between the scores, for each protein, the Smith-Waterman alignment scores to all other proteins are sorted from highest to lowest. The NCD scores of the same protein and all the other proteins are also sorted. The number of inversions [Kleinberg and Tardos, 2006] between these two protein sequences is taken to be a measure of disagreement between Smith-Waterman and NCD scores. In order to normalize for the effect of sequence length, the number of inversions is divided by the expected number of inversions n*(n-1)/4 where n is the number of proteins that are in the same length bin. The average number of normalized inversions for all proteins in each bin is computed. The higher the number of inversions, the lower the agreement between NCD and Smith Waterman scores. It is found out that as the sequence length increases, both NCD and SW become more consistent and agree on distance/similarity of a protein to other proteins. On the other hand, for small sequence lengths, NCD and SW may not agree. The reason for this may be the fact that a compression algorithm is used for NCD and for small sequence lengths, since there is not enough data, compression can not be performed efficiently.

In Eqn. 5, the Smith-Waterman and NCD scores are combined and the formulation does not include the length of the sequences. We think that the contribution of compression scores to the overall similarity needs to include the sequence length. In order to accomplish this, $f_{\alpha\beta}$ (equation 5) is modified:

$$f_{\alpha\beta\varphi\delta}(p,q) = \left(1 - \frac{SW_\alpha(p,q)}{SW_\alpha(p,p)}\right)\left((\varphi+\delta)NCD_\beta(p,q)\right) \quad (6)$$

where $\varphi$ is the normalization factor with:

$$\varphi = \left(\frac{\min\{|x|,|y|\}}{\max\{|s|, s \in dataset\}}\right) \quad (7)$$

(6) enables the NCD score to have more effect as sequence length increases. Parameter δ is included so that constant additives can also be taken into account.

## 4. Data Set

We use a data set generated from Gene Ontology hierarchy. In the data set there are 27 GO terms. In GO hierarchy, a protein may be associated with more than one term if it is known that it has multiple functions, therefore in our dataset a protein may be associated with more than one function. If two proteins have a high global alignment score (above 40 percent similarity), then they have the same function with high probability and hence function prediction is a quite easy task. The function prediction becomes an interesting task when the test sequence does not show high similarity to the training data. To remove sequence homologs, PDB's scheme of using BLASTClust algorithm for 40% sequence identity is applied. To obtain a well-balanced class distribution, classes with less than 100 or more than 550 sequences are eliminated which resulted in a dataset with 27 classes. Please see [Filiz et. al. 2008, and Filiz 2008] for more details on the data set.

Among the GO classes considered, Class 14 has to be considered as an outlier due to its average sequence length which is around 3.5 times shorter than the closest class and the ratio of its beta sheet regions which is around half of the closest class and which possibly affects the classification results.

The amino acid sequences and secondary structures of every protein are downloaded via PDB web service. The dataset obtained from the GO database includes secondary structure in DSSP representation. The secondary structure sequences are converted to HEL (H: alpha helix, E: beta sheet, L: loop) representation according to: H:G,H,I ; E: B,E ; L: C,S,T.

## 5. Experiment Details

The Smith Waterman alignment scores used in the experiments are the conservation scores produced by Balign program as explained above.

For the computation of NCD scores, the CompLearn Toolkit [Cilibrasi, 2003] developed by Cilibrasi, Cruz and De Rooij is used. Complearn is an open source toolkit built based on Vitanyi and Li's work on compression-based learning algorithms. The compression algorithm used for NCD is the LZMA (Lempel-Ziv-Markov chain-Algorithm). LZMA is a variation of the LZ77 [Lempel and Ziv, 1997]. LZMA compresses very fast and its compression ratio is 30% more than that of gzip, another LZ77 variation, and 15% more than bzip2, also another LZ77 variation, therefore it is preferred in this study.

For the classification of proteins according to the scores computed, the k-nearest neighbor (kNN) method is used. kNN is a supervised learning algorithm that classifies the test instance to the class to which the majority of the *k* nearest train instances belong [Alpaydin, 2004]. "Nearest" means having the smallest

distance computed with a certain distance measure, e.g. Euclidean or cosine. The training instances can be both positive and negative, so the kNN algorithm enables learning from negative examples, too. kNN is a quite straight-forward algorithm with low complexity and surprisingly good performance [Kocsor *et al.,* 2005]. In our experiments, we also tried SVMs, however we found out that they performed worse than kNN for this problem and are much slower. So, we prefer kNN in this study. We used k=1, which is also called 1NN.

Since the data set is multi-labeled (i.e. one protein may belong to more than one GO classes), one-against-all classification is used. In this method, the data set is divided into two subsets, the first subset is the class to be predicted and the second subset is made up by all the other classes. The aim is to correctly isolate one class from the others.

In order to get a good estimate of test performance, 10-fold cross validation is used. The positive and negative number of examples in both training and validation sets are kept proportional.

There are a number of criteria that could be used to measure performance of classification with each similarity metric. Since the data sets are unbalanced (there are many more negative instances and a much smaller number of positive instances), instead of accuracy, the AUC (Area under the ROC Curve) is used as a measure of success [Alpaydın, 2004]. The AUC value for a class is the mean of the AUC values obtained for validation data on each of its 10 folds.

## 6. Results

We first compare the performance of using SW or NCD scores by themselves on amino acid sequences only. The AUC scores for all 27 classes are shown in Figure 1. These results are consistent with that of Kocsor, where they showed that SW (average AUC=0.90) always performed better than NCD (average AUC=0.66).

Next, we determine the best amount of secondary structure contribution when using $SW_\alpha$ or $NCD_\beta$ scores. Although for each of the 27 GO classes, the best value of $\alpha$ or $\beta$ differed, according to the mean of the AUC values over all 27 classes, $\alpha=0.25$ (average AUC=0.92) and $\beta=0.5$ (average AUC=0.71) gave the best results when using SW and NCD respectively. Therefore, incorporating secondary structure increases the performance when either SW or NCD are used.

As the last set of experiments, we used formula (6) and hence all SW, NCD, amino acid sequence and secondary structure. As seen in figure 2, the AUC values are better than using NCD, however using SW results in the best performance. So, best prediction including δ is obtained by setting to 4.

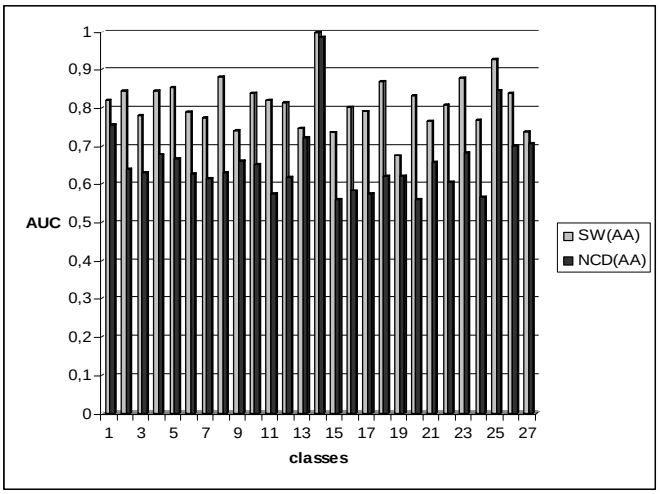


*Figure 1: Comparison of AUC values for $SW_{AA}$ and $NCD_{AA}$.*

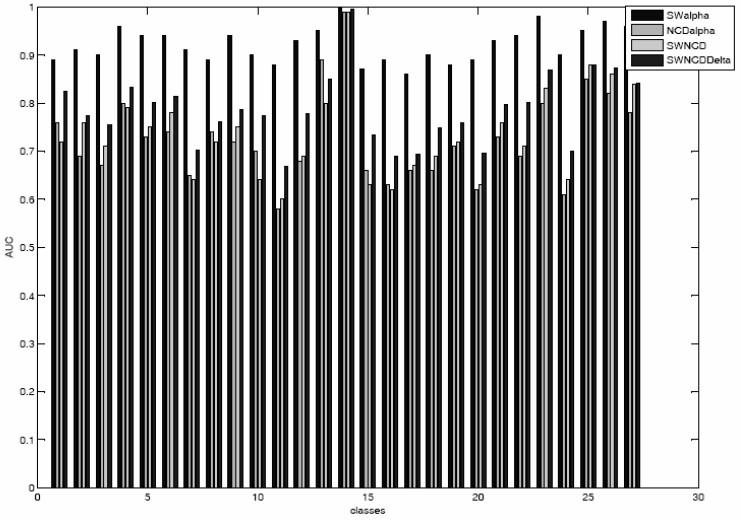


*Figure 2: Comparison of AUC values for $SW_{AA}$ and $NCD_{AA}$.*

## 7. Conclusions

We extended formulation of Kocsor *et.al.* to use both SW and NCD scores for protein function prediction, to the case when amino acid sequence and secondary structure are available. We found out that the inclusion of secondary structure improves the classification performance for both NCD and SW by themselves. However, unlike Kocsor *et.al.,* for our data set using NCD and SW together did not give better performance than using SW alone. SW was superior when we extended Kocsor *et al.*'s [2005] formulation to use NCD and SW together, to take into account the string lengths. We are in the process of finding more suitable

feature/classifier combination algorithms for this problem.

## Acknowledgements

Filiz was supported by the 2228 Tubitak scholarship program and Cataltepe is supported by Tubitak EEEAG research project 10E164. Authors would like to thank Assoc. Prof. Uluğ Bayazıt of Istanbul Technical University for useful discussions.

## References

**Alpaydın, E.,** 2004. Introduction to Machine Learning, The MIT Press, Massachusetts.

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

**Aygün, E., Kömürlü, C., Aydın, Z. and Çataltepe, Z.,** 2008. Protein Function Prediction with Amino Acid Sequence and Secondary Structure Alignment Scores, *HIBIT 2008*, Istanbul, Turkey, May 18-20.

**Aygün, E., and Çataltepe, Z.,** 2008. balign, *in preparation*.

**Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N. and Bourne, P.,** 2000. The Protein Data Bank, *Nucleic Acids Research*, **28(1),** 235-242.

**Camon, E., Magrane, M., Barrell, D., Lee, V., Dimmer, E., Maslen, J., Binns, E., Harte, N., Lopez, R. and Apweiler, R.,** 2004. The Gene Ontology Annotation (GOA) Database: sharing knowledge in Uniprot with Gene Ontology, *Nucleic Acids Research*, **32(1)**, D262-D266.

**Cataltepe, Z., Yaslan, Y. and Sönmez, A.,** 2006. Music genre classification using midi and audio features, *Journal of Applied Signal Processing*, 2006.

**Cheng, J. and Baldi, P.,** 2006. A machine learning information retrieval approach to protein fold recognition, *Bioinformatics*, **22(12)**, 1456–1463.

**Cilibrasi, R.,** 2003. The CompLearn Toolkit, *(Online)*. Available: http://complearn. sourceforge.net/

**Cilibrasi, R., Vitanyi, P.M.B. and Wolf, R.,** 2004. Algorithmic clustering of music based on string compression, *Computer Music J.,* **28(4)**, 49–67.

**Cilibrasi, R. and Vitanyi, P.M.B.,** 2005. Clustering by compression, *IEEE Trans. Inform. Th.*, **51(4)**, 523–1545.

**Ferragina, P., Giancarlo, R., Greco, V., Manzini, G. and Valiente, G.,** 2007. Compression-based classification of biological sequences and structures via the Universal Similarity Metric: experimental assessment, *BMC Bioinformatics*, **8**, 252.

**Filiz A.,** 2008. *Using Compression Based Distances for Protein Function Prediction,* M.Sc. Thesis, Istanbul Technical University Computer Engineering Dept., Turkey.

**Filiz, A., Aygün, E., Keskin, O., Çataltepe, Z.,** 2008. Importance of secondary structure elements for prediction of GO Annotations, HIBIT'08, Istanbul, Turkey.

**Freschi, V. and Bogliolo, A.,** 2005. Using sequence compression to speedup probabilistic profile matching, *Bioinformatics*, **21(10),** 2225-2229.

**Grundy, N.,** 1998. Family-based homology detection via pair-wise sequence comparison, Proc. 2nd Ann. Int. Conf. Computational Molecular Biology, 94–100.

**Hategan, A. and Tabus, I., 2004.** Protein is compressible, *Proceedings of the Northern Signal Processing Symposium*.

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

**Keogh, E., Lonardi, S. and Rtanamahatana, C.A.,** 2004. Toward parameter free data mining, *In Proceedings of the 10th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Seattle, WA, 206–215, Aug 22-25.

**Kleinberg, J. and Tardos, E.,** 2006. Algorithm Design, Pearson Education, Inc..

**Kocsor, A., Kertesz-Farkas, A., Kajan, L. and Pongor, S.,** 2005. Application of compression-based distance measures to protein sequence classification: a methodological study, *Bioinformatics*, **22(4)**, 407–412.

**Krasnogor, N. and Pelta, D.A.,** 2004. Measuring the similarity of protein structures by means of the universal similarity metric, *Bioinformatics,* **20(7)**, 1015-1021.

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

**Li, M. and Vitanyi, P.M.B.,** 1997. An Introduction to Kolmogorov complexity and its Applications. Springer Verlag, NY.

**Li, M., Badger, J.H., Chen, X., Kwong, S., Kearney, P. and Zhang, H.** , 2001. An information-based sequence distance and its application to whole mitochondrial genome phylogeny, *Bioinformatics*, **17**, 149–154.

**Liao, L. and Noble, W.S.,** 2003. Combining pairwise sequence similarity and support vector machines for detecting remote protein evolutionary and structural relationships, *Journal of Comp. Biology*, **10(6)**, 857-868.

**Nevill-Manning, C.G. and Witten, I.H.,** 1999. Protein is incompressible, *DCC '99 Data Compression Conference*, 257.

**Needleman, S. and Wunsch, C.,** 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins, *J Mol Biol*., **48(3)**, 443-53.

**Sculley, D. and Brodley, C.E.,** 2006. Compression and Machine Learning: A New Perspective on Feature Space Vectors, *Proceedings of the Data Compression Conference (DCC'06)*.

**Smith, T.F. and Waterman, M.S.,** 1981. Identification of Common Molecular Subsequences. *Journal of Molecular Biology,* **147,** 195-197.

**Wallqvist, A., Fukunishi, Y., Murphy, L.R., Fadel, A. and Levy R. M.,** 2000. Iterative sequence/secondary structure search for protein homologs: comparison with amino acid sequence alignments and application to fold recognition in genome databases, *Bioinformatics,* **16(11)**, 988-1002.

**Ziv, J. and Lempel, A.,** 1977. A universal algorithm for data compression, *IEEE Transactions on Information Theory,* **23(3),** 337-343.