國立交通大學

生物資訊及系統生物研究所

博士論文

一個基於同義字辭典的蛋白質序列分析與
分類的方法

A SYNONYMOUS DICTIONARY BASED
APPROACH FOR PROTEIN SEQUENCE
ANALYSIS AND CLASSIFICATION

研 究 生：林信男

指導教授：許聞廉 教授

何信瑩 教授

中華民國九十九年十一月

一個基於同義字辭典的蛋白質序列分析與分類的方法

A SYNONYMOUS DICTIONARY BASED APPROACH FOR PROTEIN SEQUENCE ANALYSIS AND CLASSIFICATION

研究生：林信男　　Student: Hsin-Nan Lin

指導教授：許聞廉 博士　Advisors: Dr. Wen-Lian Hsu

何信瑩 博士　　Dr. Shinn-Ying Ho

國立交通大學

生物資訊及系統生物研究所

博士論文

A Dissertation

Submitted to Institute of Bioinformatics and Systems Biology

College of Biological Science and Technology

National Chiao-Tung University

in Partial Fulfillment of the Requirements

for the Degree of Ph.D.

in

Bioinformatics

November 2010

Hsinchu, Taiwan, Republic of China

中華民國九十九年十一月

一個基於同義字辭典的蛋白質序列分析與分類的方法

研究生：林信男

指導教授：許聞廉　博士 與 何信瑩 博士

國立交通大學生物資訊與系統生物研究所

摘　　要

由於蛋白質序列不斷地增加，蛋白質序列的分析與分類在生物資訊中是非常重要的課題。許多的研究顯示蛋白質二級結構對於了解蛋白質的功能及三級結構有很大的幫助，並且透過預測蛋白質在細胞中的定位，有助於分析蛋白質的功能和藥物標靶的發現，此外找出同源蛋白質序列也是另外一個非常重要的課題。藉由偵測同源蛋白質，可以更迅速地了解未知蛋白質可能的功能和屬性。因此在本研究中，我們提出一個基於同義字辭典的蛋白質序列分析與分類的方法，用來預測蛋白質二級結構、蛋白質細胞定位和同源蛋白質偵測等相關重要課題。

在蛋白質序列分析的方法上我們採用了自然語言處理的概念，提出以同義字的方法來擷取一群同源蛋白質之間的區域相似性。一個同義字就是一個 n 字元的胺基酸片段，一組同義字可顯示蛋白質在演化過程中可能發生的序列變化。我們利用PSI-BLAST從一組蛋白質序列中產生了一個與蛋白質相依的同義字字典，以這個字典當作蛋白質序列分析與分類的參考依據。

在蛋白質二級結構預測方面，基於同義字辭典我們發展了 SymPred 與 SymPsiPred 的方法。使用一組序列相似度在 25% 以下的蛋白質序列測試預測效率，SymPred 和 SymPsiPred 平均的 Q_3 分別為 81.0% 和 83.9%。使用兩組 EVA 公用測試資料，SymPred 平均的 Q_3 分別是 78.8% 和 79.2%，預測準確率比現有方法高出 1.4% 至 5.4%。我們分析發現 SymPred 的準確率與已知蛋白質序列的數量有

正相關，這個發現說明 SymPred 和 SymPsiPred 的預測準確率會隨著蛋白質序列的增加而不斷地提高。

在蛋白質細胞定位預測中，基於同義字辭典我們發展了 KnowPred$_{\text{site}}$ 的自動預測方法。KnowPred$_{\text{site}}$ 可同時預測單一胞器定位與多胞器定位。在一組公用的測試資料中，包含了 取自1923個不同物種的 25887 單一胞器定位蛋白質與 2169 多胞器定位蛋白質。實驗結果發現KnowPred$_{\text{site}}$ 的預測準確率高於現有許多蛋白質細胞定位預測方法。在單一胞器定位預測上，KnowPred$_{\text{site}}$ 的準確率為 91.7%，高於 ngLOC 的 88.8%。在多胞器定位預測上，KnowPred$_{\text{site}}$ 的準確率為 72.1%，高於 ngLOC 的 59.7%。此外KnowPred$_{\text{site}}$ 的預測結果是可說明的，KnowPred$_{\text{site}}$ 可呈列預測結果的來源。實驗結果顯示即使序列相似度低，使用同義字辭典仍可以捕捉到有意義的區域序列相似性用來幫助預測。

在同源蛋白質序列的偵測中，基於同義字辭典我們發展了 SymDetector 用來偵測序列相似度很低的同源蛋白質。我們下載了一組公用測試資料，包含了2,476條相似度極低的蛋白質序列。在允許一個 false positive pair 的條件下，SymDetector 可偵測到 5,308 組 true positive pair，然而現有的方法 ConSequenceS及PSI-BLAST 僅能偵測到低於1,000組的 true positive pairs。隨著 false positive pair的提高為100和1000，SymDetector 可分別偵測到6,906及7,666組 true positive pairs，而相同條件下，現有的方法ConSequenceS 僅能偵測到 2,000 及3,500，而 PSI-BLAST 則僅有 ConSequenceS 所偵測到的一半。

A SYNONYMOUS DICTIONARY BASED APPROACH FOR PROTEIN SEQUENCE ANALYSIS AND CLASSIFICATION

Student: Hsin-Nan Lin

Advisors: Dr. Wen-Lian Hsu and Dr. Shinn-Ying Ho

Institute of Bioinformatics and Systems Biology
National Chiao-Tung University

Abstract

With the increasing number of protein sequences, the protein sequence analysis and classification is an important issue in Bioinformatics. Many researches show that protein secondary structure plays an important role in analyzing and modeling protein structures when characterizing the structural topology of proteins because protein secondary structure represents the local conformation of amino acids into regular structures.

The study of protein subcellular localization (PSL) is important for elucidating protein functions involved in various cellular processes. Most of the PSL prediction systems are established for single-localized proteins. However, a significant number of eukaryotic proteins are known to be localized into multiple subcellular organelles. Many studies have shown that proteins may simultaneously locate or move between different cellular compartments and be involved in different biological processes with different roles.

The analysis of novel proteins usually starts from searching homologous proteins in annotated databases. Homologous proteins usually share a common ancestor, and thus often have similar functions and structures. Based on pairwise identities and some specific thresholds, sequence search tools retrieve annotated homologous sequences to infer annotations of the novel sequences. As the number of protein sequences grows, sensitive strategies of homology detection using simply sequence information are still

demanding and of great importance in post-genomic era. Sequence similarity is a frequently used simple metric for homology detection and other annotation transfers. However, sequence itself provides incomplete and noisy information about protein homology. Many improvements on homology searching and sequence comparisons have been developed to overcome the limitation of sequence similarity.

Based on above observation, we propose a general approach based on a synonymous dictionary for protein sequence analysis and classification. We apply it to the problems of protein secondary structure prediction, protein subcellular localization and remote homology detection. We adopt the techniques from natural language processing and use synonymous words to capture local sequence similarities in a group of similar proteins. A synonymous word is an *n*-gram pattern of amino acids that reflects the sequence variation in a protein's evolution. We generate a protein-dependent synonym dictionary from a set of protein sequences.

Protein secondary structure prediction: On a large non-redundant dataset of 8,297 protein chains (*DsspNr-25*), the average Q_3 of SymPred and SymPsiPred are 81.0% and 83.9% respectively. On the two latest independent test sets (*EVA_Set1* and *EVA_Set2*), the average Q_3 of SymPred is 78.8% and 79.2% respectively. SymPred outperforms other existing methods by 1.4% to 5.4%. We study two factors that may affect the performance of SymPred and find that it is very sensitive to the number of proteins of both known and unknown structures. This finding implies that SymPred and SymPsiPred have the potential to achieve higher accuracy as the number of protein sequences in the NCBInr and PDB databases increases.

Protein subcellular localization: We downloaded the dataset from ngLOC, which consisted of ten distinct subcellular organelles from 1923 species, and performed ten-fold cross validation experiments to evaluate KnowPred$_{site}$'s performance. The experiment results show that KnowPred$_{site}$ achieves higher prediction accuracy than ngLOC and Blast-hit method. For single-localized proteins, the overall accuracy of KnowPred$_{site}$ is 91.7%. For multi-localized proteins, the overall accuracy of KnowPred$_{site}$ is 72.1%, which is significantly higher than that of ngLOC by 12.4%. Notably, half of the proteins

in the dataset that cannot find any Blast hit sequence above a specified threshold can still be correctly predicted by KnowPred$_{site}$.

Remote homology detection: We propose a two-stage method called SymDetector for the problem of remote homology detection. We downloaded a benchmark dataset which contains 2,476 protein sequences with mutual sequence identity below 25%. When allowing only one false positive, SymDetector achieves 5,308 true positive pairs while ConSequenceS and PSI-BLAST report less than 1,000 true homologous ones. As the error rate grows, SymDetector can identify 6,906 along with 7,666 sequence pairs given 100 and 1000 false positives permitted separately. Under the same setting, ConSequenceS only reports about 2,000 and 3,500 pairs in the same Fold, which improve PSI-BLAST by 50% in average.

ACKNOWLEDGEMENT

在中研院擔任研究助理與攻讀博士學位的期間，我由衷感謝許聞廉老師與宋定懿老師給予我許多的指導。無論是在研究上或生活上，許老師與宋老師總是如嚴師與慈父、慈母般不吝惜地給予我最大的幫助和教誨。在中研院得有兩位老師如此盡心盡力地指導，是我人生中最大的福氣。許老師與宋老師所展現的嚴謹與認真的研究精神和態度，更是我學習的典範。我還要感謝交大何信瑩老師、黃鎮剛老師、楊進木老師和盧錦隆老師在百忙當中擔任我的口試委員，並且針對我的報告提出許多有待改進的地方，讓我體認到學術研究所需具備的條件，這都是我未來繼續從事研究所要努力的目標。

此外，我還要感謝生物資訊實驗室的所有夥伴。巫坤品學長在我剛進中研院的時候，總是像一個大哥般地照顧我，研究上也給予我許多相當寶貴的意見。與張家銘、邱樺聲、陳鯨太、楊翊文和林晏禕共事期間得到許多的幫助。而林可軒博士、顏義樺、周文奇、崔殷豪、蘇家玉博士、羅光倫博士(Dr. Allan Lo)、邱一原和紀俊男等等這些實驗室好夥伴，不僅是在生活上和研究上給予我許多協助，也讓枯燥的實驗室生活有了許多的歡樂。李政緯博士和戴敏育博士也教導我許多程式上的技巧。

最後要感謝我的家人。我的父母、岳父母在生活上給予我很大的支持，感謝他們對我的栽培，使我在博士求學期間無後顧之憂。還有愛妻永菁的陪伴，讓我有一個堅強和溫軟的依靠。還有在天上的阿嬤，謝謝您一直都這麼疼我，支持我。謝謝大家!

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1 Introduction

1.1 Protein Secondary Structure Prediction

Proteins can perform various functions when they fold into proper three-dimensional structures. Because determining the structure of a protein through wet-lab experiments can be time-consuming and labor-intensive, computational approaches are preferable. To characterize the structural topology of proteins, Linderstrøm-Lang proposed the concept of a protein structure hierarchy with four levels: primary, secondary, tertiary, and quaternary. The primary structure of a protein refers to its amino acid sequence. The secondary structure consists of the coiling or bending of amino acids. The tertiary structure is the folding of a molecule upon itself by disulfide bridges and hydrogen bonds. The quaternary structure refers to the complex structure formed by the interaction of 2 or more polypeptide chains. In the hierarchy, protein secondary structure (PSS) plays an important role in analyzing and modeling protein structures because it represents the local conformation of amino acids into regular structures.

There are three basic secondary structure elements (SSEs): α-helices (H), β-strands (E), and coils (C). Many researchers employ PSS as a feature to predict the tertiary structure [1-4], function [5-8], or subcellular localization [9] of proteins. It is noteworthy that, among the various features used to predict protein function, such as amino acid composition, disorder patterns, and signal peptides, PSS makes the largest contribution

[10]. Moreover it has been suggested that secondary structure alone may be sufficient for accurate prediction of a protein's tertiary structure [11].

Current PSS prediction methods can be classified into two categories: template-based methods and sequence profile-based methods [12]. Template-based methods use protein sequences of known secondary structures as templates, and predict PSS by finding alignments between a query sequence and sequences in the template pool. The nearest-neighbor method belongs to this category. It uses a database of proteins with known structures to predict the structure of a query protein by finding nearest neighbors in the database. By contrast, sequence profile-based methods (or machine learning methods) generate learning models to classify sequence profiles into different patterns. In this category, Artificial Neural Networks (ANNs), Support Vector Machines (SVMs) and Hidden Markov Models (HMMs) are the most widely used machine learning algorithms [13-19].

Template-based methods are highly accurate if there is a sequence similarity above a predefined threshold between the query and some of the templates; otherwise, sequence profile-based methods are more reliable. However, the latter may under-utilize the structural information in the training set when the query protein has some sequence similarity to a template in the training set [12]. An approach that combines the strengths of both types of methods is required for generating reliable predictions irrespective of whether the query sequence is similar or dissimilar to the templates in the training set.

To measure the accuracy of secondary structure prediction methods, researchers often use the average three-state prediction accuracy (Q_3) accuracy or the segment overlap (SOV) measure [20-21]. The estimated theoretical limit of the accuracy of secondary structure assignment from the experimentally determined 3D structure is 88% of the Q_3 accuracy [5, 22], which is deemed the upper bound for secondary structure prediction. However, PSS prediction has been studied for decades and has reached a bottleneck, since the Q_3 accuracy remains at approximately 80 % and further improvement is very difficult, as demonstrated by the CASP competitions. Currently, the most effective PSS prediction methods are based on machine learning algorithms, such as PSIPRED [15], SVMpsi [17], PHDpsi [23], Porter [24] and SPINE [25], which employ ANN or SVM learning models. The two most successful template-based methods are NNSSP [26-27] and PREDATOR [28]. They use the structural information obtained from local alignments among query proteins and template proteins, and their Q_3 accuracy is approximately 70%. Thus, the difference in the accuracy of the two categories is approximately 10%.

In a previous work on PSS prediction [29], we proposed a method called PROSP, which utilizes a sequence-structure knowledge base to predict a query protein's secondary structure. The knowledge base consists of sequence fragments, each of which is associated with a corresponding structure profile. The profile is a position specific scoring matrix that indicates the frequency of each SSE at each position. The average Q_3 accuracy of PROSP is approximately 75%.

In this study, we present an improved version of PROSP called SymPred, which is a dictionary-based method for predicting the secondary structure of a protein sequence.

Dictionary-based approaches are widely used in the field of natural language processing (NLP) [30-32]. We generate synonymous words from a protein sequence and its similar sequences. The definition of a synonymous word is given in the Chapter Two. The major differences between SymPred and PROSP are as follows. First, the constitutions of the dictionary (SymPred) and the knowledge base (PROSP) are different. Second, the scoring systems of SymPred and PROSP are different. Third, unlike PROSP, SymPred allows inexact matching. Our experiment results show that SymPred can achieve 81.0% Q_3 accuracy on a non-redundant dataset, which represents a 5.9% performance improvement over PROSP.

There are significant differences between SymPred and other methods in the two categories described earlier. First, in contrast to template-based methods, SymPred does not generate a sequence alignment between the query protein and the template proteins. Instead, it finds templates by using local sequence similarities and their possible variations. Second, SymPred is not a machine learning-based approach. Moreover, it does not use a sequence profile, so it cannot be classified into the second category. However, like machine learning-based approaches, SymPred could capture local sequence similarities and generate reliable predictions. Therefore, SymPred could combine the strengths of template-based and sequence profile-based methods. The experiment results on the two latest independent test sets (*EVA_Set1* and *EVA_Set2*) show that, in terms of Q_3 accuracy, SymPred outperforms other existing methods by 1.4% to 5.4%.

1.2 Protein Subcellular Localization Prediction

Protein subcellular localization (PSL) is important to elucidate protein functions as proteins cooperate towards a common function in the same subcellular compartment [33]. It is also essential to annotate genomes, to design proteomics experiments, and to identify potential diagnostic, drug and vaccine targets [34]. Determining the localization sites of a protein through experiments can be time-consuming and labor-intensive. With the large number of sequences that continue to emerge from the genome sequencing projects, computational methods for protein subcellular localization at a proteome scale become increasingly important.

Most existing PSL predictors are based on machine learning algorithms. They can be categorized by the feature sets used for building prediction models. A group of methods use features derived from primary sequence [35-39]; some utilize various biological features extracted from literature or public databases [9, 34, 40-44]. Other features are also used in different methods, e.g., phylogenetic profiling [45], domain projection [46], sequence homology [38], and compartment-specific features [47].

A simple and reliable way to predict localization site is to inherit subcellular localization from homologous proteins. Therefore, in [38] a hybrid method was proposed, which combined an SVM based method with a sequence comparison tool to find homology to improve the performance. However, some homologous proteins are not similar in sequences, but in structures. For example, the sequence identity between proteins *1aab* and *1j46* is only 16.7% but they are structurally homologous and classified into the same family (*HMG-box*) in the SCOP classification. For such cases, it is difficult to discover the homologous relationship using sequence comparison methods. Profile-profile alignment methods [48-52] are capable of identifying remote homology; nevertheless, they are relatively slow.

Most of the PSL prediction systems are established particularly for single-localized proteins. A significant number of eukaryotic proteins are, however, known to be localized into multiple subcellular organelles [53-54]. In fact, proteins may simultaneously locate or move between different cellular compartments and be involved in different biological processes with different roles. This type of proteins may take a high proportion, even more than 35% [53]. In addition, the majority of existing computational methods have the following disadvantages [54]: 1) they only predict a limited number of locations; 2) they are limited to subsets of proteomes which contain signal peptide sequences or with prior structural/functional information; 3) the datasets used for training are for specific species, which is not sufficiently robust to represent the entire proteomes. Thus, most of the computational methods are not sufficient for proteome-wide prediction of PSL across various species.

Thus in this study, we propose a synonymous dictionary based approach, called KnowPred$_{\text{site}}$, using local sequence similarity to find useful proteins as templates for site prediction of the query protein. It is designed to predict localization site(s) of single- and multi-localized proteins and is applicable to proteome-wide prediction. Furthermore, it only requires protein sequence information and no functional or structural information is required. Notably, prediction results can be explained by the template proteins which are used to vote for the localization sites. The dictionary based prediction scheme has been shown to be effective in predicting protein secondary structure [29, 38, 55] and local structure [56]. To evaluate our prediction method, we used the ngLOC dataset [54] to perform ten-fold cross validation to compare with existing methods. The dataset consists of ten subcellular proteomes from 1923 species with single- and multi-localized proteins. KnowPred$_{site}$ achieved 91.7% accuracy for single-localized proteins and 72.1% accuracy with both sites correctly predicted for multiple localized proteins.

1.3 Remote Homology Detection

The analysis of novel biological sequences usually starts from searching homologous sequences in annotated databases. Homologous sequences usually share a common ancestor, and thus often have similar functions and structures. Based on pairwise identities and some specific thresholds, sequence search tools retrieve similar annotated sequences for homology inferences, which are crucial in advanced analysis, such as protein structure modeling, function predictions, protein-protein interaction networks analysis, and other property annotations. While structural information assists to increase the understanding of some target proteins, in many situations one has to analyze a protein based on its sequence information only. The advent of whole genome sequencing generates large amounts of protein sequences with undetermined structures and functions.

Many of these newly sequenced proteins, including those related to diseases, have few closely related homologs in annotated databases. In addition, as the number of sequenced genomes and proteins grows, many relationships between distantly related proteins are observed and needed to be studied further for better understanding the complex structure of protein universe. Sensitive strategies for analyzing proteins based on simply sequence information are therefore still demanding and of great importance in genomic era.

Sequence similarity is a frequently used simple metric for homology detection and other annotation transfers. However, sequence itself provides only incomplete and noisy information about the protein. The most similar result may not be the most relevant sequence [57], while some other homologous sequences might be lost in the search results. For example, two sequences are usually identified as homologs if their pairwise similarity is higher than 40%, but the problem becomes rather challenging for sequences sharing similarity between 20% and 35%, i.e., sequences in the twilight zone. Studies showed that even for protein pairs with sequence identity less than 25%, about slightly less than 10% of them still homologous [58]. Thus pairwise sequence similarity has its limit in detecting distant sequence relationships. Using a threshold of pairwise sequence identity to determine homology relationship is arguable since it is hard to determine whether protein pairs having sequence identities lower than this threshold are homologous. Once pairwise similarity of a sequence pair is below a specified threshold, we can hardly distinguish whether the pair of sequences is from homology or not. Therefore many improvements on homology searching and sequence comparisons have been developed to overcome the limitation of sequence similarity [59-60].

To improve sequence-based analysis strategies, we have to determine the strategies to represent proteins and corresponding similarity metrics for such representations. Based on these two issues, homology detection methods can be roughly divided into two categories: generative models and discriminative models. Given a protein sequence, generative models focus on describing a set of known proteins with a probabilistic model, and propose a probabilistic measurement between the query protein and the model. On the other hand, discriminative models focus on differences between two sets of proteins.

Homology search tools of generative models consist of profile-profile comparisons and profile-sequence methods. Since sequence information itself is insufficient, researchers devise probabilistic models to represent the protein sequences, such as PSSM [61] and profiles [62] and profile Hidden Markov Models [63-65]. Some famous packages include HMMER and HMMERHEAD [66] , COMPASS [67-69], COACH [70], HHSearch [71], and profile comparison tools such as PRC [72]. While there might be concerns about the statistical measurement about accuracies for these model-comparison tools [73-74], they provide best available results among generative model methods. These tools, however, are time-consuming. Therefore profile-sequence (sequence-profile) search tools that strike balances between speed and accuracy are de facto standards for large-scale database searching. PSI-BLAST [75] is definitely the Google for bioinformatics community, while CS-BLAST/CSI-BLAST [76] provides more sensitive results based on similar ideas. More detailed comparison could be found in [77].

Discriminative models mainly focus on designing kernel functions based on sequence patterns to distinguish sequences from two different sets. Most of these methods are based on support vector machines, and extract frequent patterns from sequences as their features in the string kernel. The first string kernel might be Fisher's kernel [78]. Some popular string kernels includes, but not limited to, Pairwise kernel[79], Spectrum and the Mismatch kernels [80-81], Local Alignment method [82], and Word Correlation Matrices [83]. Some methods integrate structural and motif information into the feature set, such as I-Sites [84], eMOTIF-database search [85], Profile-Based Mismatch methods [86] and Profile-based direct methods [87]. Readers can find more comprehensive information about discriminative methods in the following materials [88-89].

While discriminative models, especially string kernels methods, achieve better performance than generative models in some comparative studies [79, 81], these results often lack of evidences for interpretations, such as HSPs in general alignment tools. In addition, they may lead to over-fitting due to parameter setting and feature selections. Therefore, many strategies attempt to improve homology detections based on results of generative models, especially on results of PSI-BLAST. RankProt [90] attempts to consider pairwise distances between all the query sequences to construct a relation network, and increase homology detection results based on analyzing the network information. Ku and Yona [91] propose a framework based on similar ideas. Since there are already lots of annotated sequences in current databases, a natural thought is to integrate information from external sequences to boost homology detection.

A simple attempt to integrate external sequence information in homology detection might be intermediate sequence search (ISS) [92-93]. In short, if protein sequences A and B are both homologous to the third sequence C, A and B may be detected as homologs although they share low identities. Improved frameworks based on similar ideas consist of SCOOP[94] and SIMPRO[95]. Moreover, some strategies tend to apply information from the probabilistic models, instead of shared sequences only. Consensus-sequence-based methods are representatives of these kinds of strategies. PHOG-BLAST [96] make sequence profiles discrete, and generate consensus for a query sequence by substituting each residue with the most important amino acids in the original sequence. Recently, Przybylski and Rost generalize such consensus-based concepts for boosting homology search for sequences of low identities [97-98]. For an unknown sequence, they search it against NCBInr to obtain its PSSM. Then the original sequence is transformed to a consensus sequence based on this PSSM. They claim that, by using the informative consensus sequence as the object in comparisons, homology search results would be better than traditional PSI-BLAST searches.

Based on above observation, we aim to design a computational framework for detecting distantly relationships between protein sequences in twilight zone (sequence identities between 25% and 40%) or midnight zone (sequence identities below 25%) with several properties. First, it should deal with sequence relationships among proteins with low sequence identity. Second, the results of the framework should be explainable. That is, we hope the result can provide evidence, and even high quality alignments to support its identification, instead of some profiles or a set of dozens of features. Third, the framework is computationally incremental, and we can easily add or delete sequences in our training set. Besides, this framework should make best use of the power of current homology search tools to make it simple to be implemented. As a result, we use fixed-length protein words as possible homology indicator in this framework. For each word in separate sequences, we use PSI-BLAST to generate its variations. These variations would be integrated to estimate relations between novel sequences and annotated sequences. We demonstrate that this framework achieves high sensitivity in discovering protein homologs even though they share low sequence similarities with annotated sequences.

Chapter 2 Synonymous Words and a Protein-dependent Synonymous Dictionary

2.1 Synonymous Words versus Similar Words

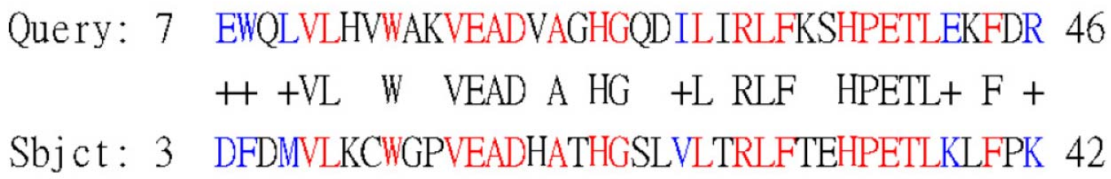
It is well known that a protein structure is encoded and determined by its amino acid sequence. Therefore, a protein sequence can be treated as a text written in an unknown language whose alphabet comprises 20 distinct letters; and the protein’s structure is analogous to the semantic meaning of the text. Currently, we cannot decipher the “protein language” with existing biological experiments or natural language processing (NLP) techniques; thus, the translation from sequence to structure remains a mystery. However, biologists have found that two proteins with a sequence identity above 40% may have a similar structure and function. The high degree of robustness of the structure with respect to the sequence variation shows that the structure is more conserved than the sequence.

In evolutionary biology, protein sequences that derive from a common ancestor can be traced on the basis of sequence similarity. Such sequences are referred to as homologous proteins. In terms of natural language, a group of homologous protein sequences can be treated as texts whose semantic meaning is identical or similar. The homologous relationship between proteins can be always captured by sequence alignment; thus, we assume that two sequence fragments have a similar semantic relation if they can be aligned by a sequence alignment tool, such as BLAST, with a significant e-value, say 0.001. Figure 1 shows an example of a sequence alignment derived by BLAST with an e-value of 0.001. In the alignment, the identical residues are labelled with letters and conserved substitutions are labelled with + symbols. The sequence identity between the two sequence fragments in this example is 50% (=20/40).

The idea of treating n-gram patterns as words has been widely used in biological sequence comparison methods; BLAST is probably the most well known method. BLAST's heuristic algorithm uses a sliding window to generate an initial word list from a query sequence. To further expand the word list, BLAST defines a *similar word* with respect to a word on the list based on the score of the aligned word pair. A word whose alignment score is well above a threshold is called a similar word and is added to the list to recover the sensitivity lost by only matching identical words. However, in BLAST, the length of a word is only 2 or 3 characters (the default size) for protein sequences and short words are very likely to generate a large number of false hits of protein sequences that are not actually semantically related.

In this study, we define synonymous words as follows. Given a protein sequence *p*, we use PSI-BLAST to generate a number of significant sequence alignments, called *high-scoring segment pairs* (HSPs), between *p* and its similar proteins *sp*. All words, i.e., *n*-grams, in *p* and *sp* are generated by a sliding window of size *n*. Given a word *w* in *p*, the *synonymous word* of *w* is defined as the word *sw* in *sp* that is aligned with *w*. Please note that no gap is allowed in either *w* or *sw* since there is no structural information in the gap region. Thus, the major difference between synonymous words and similar words is that synonymous words are based on sequence alignments (i.e., they are context-sensitive), whereas similar words are based on word alignments (i.e., they are context-free). Take the sequence alignment (or High-scoring Segment Pair, HSP) in Figure 1 as an example. The *Sbjct* sequence is a similar protein to the *Query* sequence; therefore, DFDM is deemed synonymous to the word EWQL if the word length is 4, and FDMV is deemed synonymous to the next word WQLV. Based on the observation of the high robustness of structures, if the *Query* is of known structure and the *Sbjct* is of unknown structure, we assume that each synonymous word *sw* adopts the same structure as its corresponding word *w*; i.e., *sw* inherits the structure of *w*.

Moreover, different synonymous words *sw* for a word *w* should have different similarity scores to *w*. To estimate the similarity between *w* and *sw*, we calculate the *similarity level* according to the number of amino acid pairs that are *interchangeable*. If two amino acids are aligned in a sequence alignment, they are said to be *interchangeable* if they have a positive score in BLOSUM62. Since a protein word is an n-gram pattern, the range of the similarity level between the components of a word pair is from 0 to *n*. For example, in Figure 2, the similarity level between DFDM and EWQL is 3, and that between FDMV and WQLV is also 3.



```
Query: 7   EWQLVLHVWAKVEADVAGHGQDILIRLFKSHPETLEKFDR 46
           ++ +VL  W  VEAD A HG  +L RLF  HPETL+ F +
Sbjct: 3   DFDMVLKCWGPVEADHATHGSLVLTRLFTEHPETLKLFPK 42
```

Figure 1 – A local sequence alignment (or High-scoring Segment Pair, HSP) derived by PSI-BLAST. The identical residues are labelled with letters and conserved substitutions are labelled with + symbols. The alignment in this example shows that the sequence fragment from position 7 to position 46 of the query sequence is very similar to that from position 3 to position 42 in the subject sequence. It is assumed that the two sequences have a similar semantic relation because they form a significant sequence alignment.

2.2 Advantages of Synonymous Words

The major advantages of using synonymous words over similar words are as follows. First, since the synonymous words are generated from a group of similar proteins, two irrelevant proteins would use different groups of similar proteins to generate their own synonymous words. Two irrelevant proteins would be unlikely to have common synonymous words, even if their original sequences had contained identical words. This observation implies that synonymous words probably tend to *protein-dependent*.

Second, two remote homologous proteins might be very likely to have common similar proteins because of the transitivity of the homology relationship, so they probably share some synonymous words. Transitivity refers to deducing a possible similarity between protein *A* and protein *C* from the existence of a third protein *B*, such that *A* and *B* as well as *B* and *C* are homologues if the sequence identity between *A* and *B* as well as that between *B* and *C* is above the predefined threshold. Figure 2(a) shows an example of transitivity relationship among protein *A*, protein *B*, and protein *C*. Protein *A* and protein *B* share sequence identity of 34%, and protein *B* and protein *C* share sequence identity of 27%, whereas protein *A* and protein *C* only share sequence identity of 12%. Using the transitivity relationship, remote homologous relationship and local similarity between protein *A* and protein *C* can be detected. In this study, we apply the transitivity concept to peptide fragments instead of the protein sequences to obtain local similarities between remotely homologues. Protein *A* and protein *C* share local similarity if there is a protein word aligned with the words in protein *A* and protein *C*. Figure 2(b) illustrates the idea, in which protein *A* and *C* are aligned with protein *B1* and protein *B2* (*B1* and *B2* can be identical, homologous or non-homologous). If there is a protein word shared by both *B1* and *B2*, the corresponding protein words in protein *A* and protein *C* are inferred as locally similar between protein *A* and protein *C*. The shared synonymous word may represent a possible sequence variation in evolution. Moreover, if protein *A* and protein *C* are remotely homologous, there are likely more shared synonymous words in different protein *B*'s to characterize their similarity.

Third, a synonymous word is given a similarity score (i.e., the similarity level) respective to the word it is aligned with. Therefore, a synonymous word may have different similarity scores depending on which word it is aligned with. Accordingly, a synonymous word is a protein-dependent similar word that may also have a similar semantic meaning in terms of its structure.

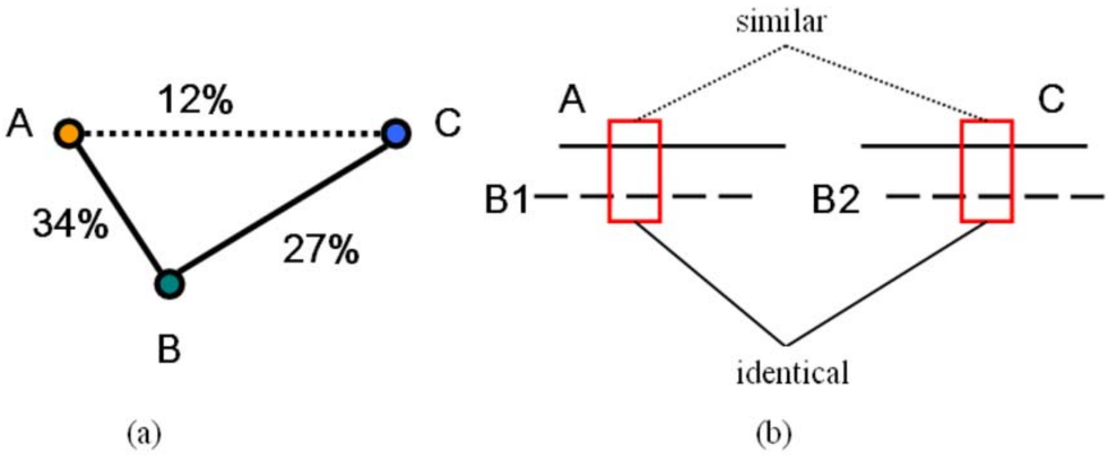


Figure 2 – Two different transitivity relationships. (a) Protein *A* and protein *B* share sequence identity of 34%, and protein *B* and protein *C* share sequence identity of 27%, whereas protein *A* and protein *C* only share sequence identity of 12%. We infer the homologous relationship between *A* and protein *C* through protein *B*. (b) Protein *A* and protein *C* are aligned with protein *B1* and protein *B2*. The peptide fragments of *B1* and *B2* besieged by the rectangles are identical, the two corresponding peptide fragments of *A* and *C* are considered to be similar.

In this study, we construct a protein-dependent synonymous word dictionary that lists possible synonyms for words of a protein sequence in a dataset. We use synonymous words as features to infer structural information for the problems of protein secondary structure prediction, protein subcellular localization prediction, and remote homology detection.

2.3 Construction of a protein-dependent Synonymous Dictionary

Given a query sequence, we use PSI-BLAST to generate a number of significant alignments, from which we acquire possible sequence variations. In general, the similar protein sequences (i.e., the Sbjct sequences) reported by PSI-BLAST share highly similar sequence identities (between 25% and 100%) with the query, which implies that the sequences may have similar structures. Therefore, we identify synonymous words in those sequences.

Using a dataset of protein sequences with known secondary structures, we construct a protein-dependent synonymous dictionary, called *SynonymDict*. For each protein *p* in the dataset, we first extract protein words from its original sequence using a sliding window of size *n*. Each protein word, as well as the corresponding SSEs of the successive *n* residues, the protein source *p*, and the similarity level (here, the similarity level is *n*), are stored as an entry in *SynonymDict*. A protein source *p* represents the structural information provider. We then use PSI-BLAST to generate a number of similar protein sequences. Specifically, to find similar sequences, we perform a PSI-BLAST search of the NCBInr database with parameters *j*=3, *b*=500, and *e*=0.001 for each protein *p* in the dataset. Since the NCBInr database only contains protein sequence information, each synonymous word inherits the SSEs of its corresponding word in *p*. A PSI-BLAST search for a specific query protein *p* generates a number of local pairwise sequence alignments between *p* and its similar proteins. Statistically, an e-value of 0.001 generally produces a safe search and signifies sequence homology [99]. Similarly, each synonymous word and its inherited structure, the protein source *p*, and the similarity level are stored as an entry in *SynonymDict*.

Figure 3 shows the procedure used to extract protein words and synonymous words for a query protein *p*. We use a sliding window to screen the query sequence, as well as all the similar protein sequences found by PSI-BLAST, and extract all words. The query protein *p* is the protein source of all the extracted words. Each word is associated with a piece of structural information of the region from which it is extracted. For example, WGPV is a synonymous word of WAKV. Since it is from a similar protein of unknown structure, it is associated with a piece of structural information of WAKV, which is HHHH.

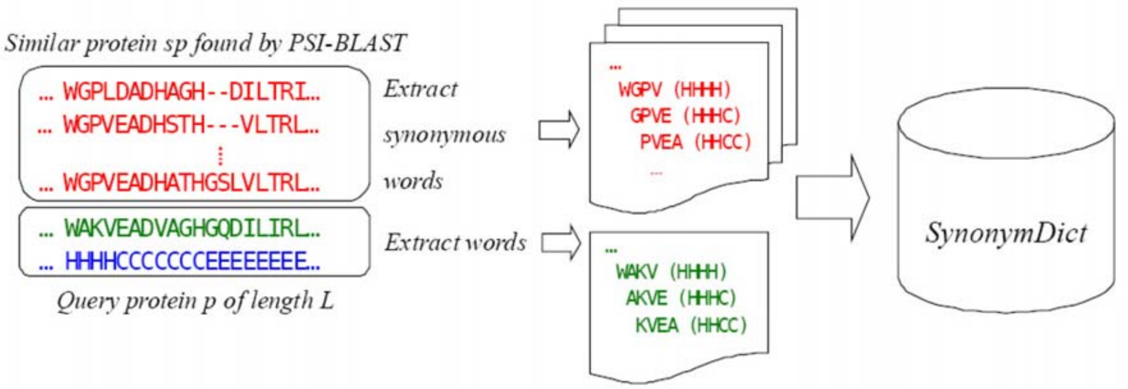

Similar protein sp found by PSI-BLAST
... WGPLDADHAGH--DILTRI...
... WGPVEADHSTH---VLTRL...
... WGPVEADHATHGSLVLTRL...
... WAKVEADVAGHGQDILIRL...
... HHHHCCCCCCEEEEEEEE...
Query protein p of length L
Extract synonymous words
Extract words
...
WGPV (HHHH)
GPVE (HHHC)
PVEA (HHCC)
...
...
WAKV (HHHH)
AKVE (HHHC)
KVEA (HHCC)
SynonymDict

Figure 3 – The procedure used to extract protein words and synonymous words for a query protein p. The procedure used to extract protein words and their synonymous words for a given query protein p (assuming the window size n is 4). We use a sliding window to screen the query sequence and all the similar protein sequences found by PSI-BLAST and extract all words. Each word is associated with a piece of structural information of the region from which it is extracted. The protein source of all the extracted words is the query protein p, since all the structural information is derived from p.

Note that a synonymous word may appear in more than one similar protein when all similar protein sequences are screened. We cluster identical words together and store the frequency in the synonymous word entry. Table 1 shows an example of a synonymous word entry in *SynonymDict*. In the example, WGPV is a synonymous word of proteins *A*, *B* and *C*, since it is extracted from the similar proteins of *A*, *B* and *C*. The synonymous word inherits the corresponding structural information of its source, and we can derive the corresponding similarity levels and frequencies via the extraction procedure. For example, the similarity level of WGPV in terms of protein source *A* is 3 and the frequency is 7. This implies that WGPV has 3 interchangeable amino acids with the corresponding protein word of *A* and it appears 7 times among the similar proteins of *A* found in the PSI-BLAST search result.

In Table 1, we store the inherited secondary structural information for the synonymous word WGPV. We can use the structural information to predict the secondary structure for a given protein sequence. In fact, we can store other protein related information in a synonymous word entry, such as protein subcellular localization sites, protein function labels, or structural classes, etc. In Table 2 we show another example of a synonymous word entry which stores the protein subcellular localization sites. Using the stored information, we can study different protein prediction or classification problems.

Table 1 – An example of a synonymous word entry in *SynonymDict*. An example of a synonymous word entry in SynonymDict (assuming the word length n = 4). WGPV is a synonymous word of proteins A, B and C, since it is extracted from the similar proteins of A, B and C. We record the structural information of protein sources to the corresponding synonymous words, and calculate the corresponding similarity levels and frequencies. For example, the similarity level of WGPV in terms of protein source A is 3 and the frequency is 7.

Synonymous word: WGPV			
Protein Source	Secondary Structure	Similarity Level	Frequency
A	HHHH	3	7
B	HHCH	4	11
C	CHHH	2	3

Table 2 – Another example of a synonymous word entry in *SynonymDict*. Three protein sources with known localization sites contain protein words that are aligned to the word MYSKILL in the corresponding sequence alignments. We store the inherited subcellular localization sites for MYSKILL from the protein sources A, B, and C.

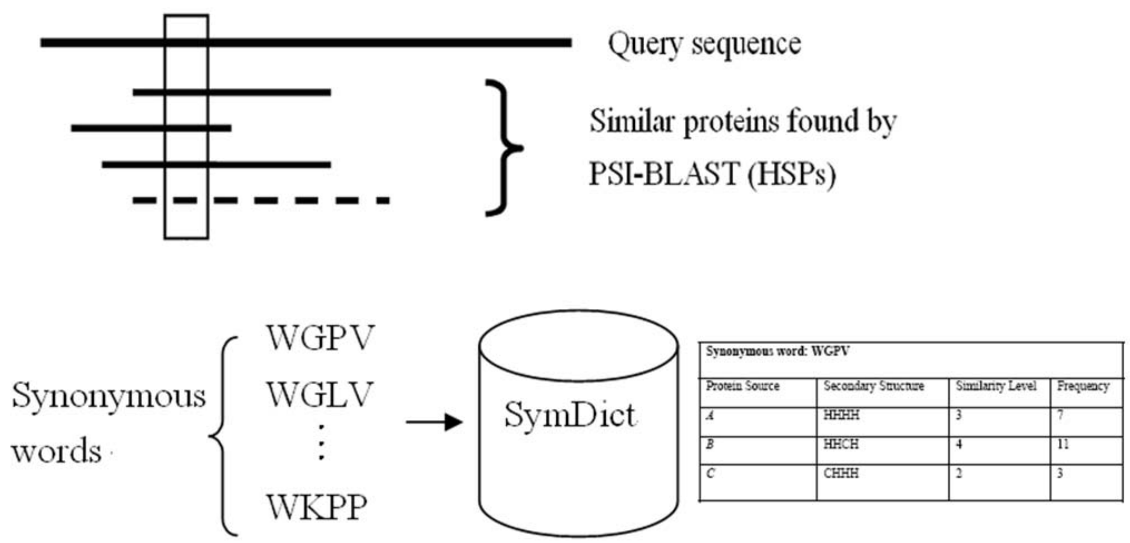
Synonymous word: MYSKILL			
Protein Source	Localization Sites	Similarity Level	Frequency
A	Cytoplasm	5	21
B	Nuclear	4	12
C	Cytoplasm Extracellular	5	17

Chapter 3 Protein Secondary Structure Prediction

3.1 Methods

In this section, we present our synonymous dictionary based approach for protein secondary structure prediction, called SymPred, and a meta-predictor, called SymPsiPred.

3.1.1 SymPred: a PSS predictor based on SynonymDict



Synonymous word: WGPV			
Protein Source	Secondary Structure	Similarity Level	Frequency
A	HHHH	3	7
B	HHCH	4	11
C	CHHH	2	3

Figure 4 – The prediction procedure of SymPred. An HSP represents a high-scoring segment pair which is a significant sequence alignment reported by PSI-BLAST.

Preprocessing

Figure 4 shows the prediction procedure of SymPred. Given a target protein *t*, whose secondary structure is unknown and to be predicted, we perform a PSI-BLAST search on *t* to compile a word set containing its original protein words and synonymous words. The procedure is similar to the construction of *SynonymDict*. We also calculate the frequency and similarity level of each word in the word set.

Exact and inexact matching mechanisms for matching words to *SynonymDict*

Each word *w* in the word set is used to match against words in *SynonymDict*, and the structural information of each protein source in the matched entry is used to vote for the secondary structure of *t*. When matching a word to *SynonymDict*, we consider using straightforward exact matching and a simple inexact matching. Exact matching is rather strict, so we consider a possible relaxation of inexact matching to increase the sensitivity to recover synonymous word matches so that *SynonymDict* can be utilized to more extent than by using exact matching. Our inexact matching allows at most one mismatched character, i.e., allowing a don't-care character (not a gap) in the words. The matched entries are then evaluated by the following scoring function.

The Scoring Function

To differentiate the effectiveness of matched entries, we design a scoring function based on the protein sources in the matched entries and the sum of the weighted scores on the associated structures determines the predicted structure.

Since we use the structural information of protein sources in the matched entries for structure prediction, we define the scoring function based on its similarity level and frequency recorded in the dictionary for the following observation. *The similarity level represents the degree of similarity between a protein word and its synonymous word, and the frequency represents the degree of sequence conservation in the protein's evolution.* Intuitively, the greater the similarity between two words, the closer they are in terms of evolution; likewise, the more frequently a word appears in a group of similar proteins, the more conserved it is in terms of evolution.

To define the scoring function, we consider the similarity level and the frequency of the word in the word set of t, denoted by Sim_t and $freq_t$ respectively, as well as those of a protein source i in its matched entry, denoted by Sim_i and $freq_i$ respectively. Note that Sim_t and $freq_t$ are obtained in the preprocessing stage. To measure the effectiveness of the structural information of the protein source i, we define the voting score s_i as $min(freq_t, freq_i)\times(1+min(Sim_t, Sim_i))$. The structural information provided by i will be highly effective if: 1) w is very similar to the corresponding words of t and i; and 2) w is well conserved among the similar proteins of t and i.

Take the synonymous word WGPV in Table 1 as an example. If WGPV is a synonymous word of t (assuming $freq_t$ is 5 and Sim_t is 4), then the voting score of the structural information provided by protein source A is $min(5, 7)\times(1+min(4, 3)) = 5\times(1+3) = 20$. Similarly, the voting score provided by protein source B is $min(5, 11)\times(1+min(4, 4)) = 5\times(1+4) = 25$, and the score provided by protein source C is $min(5, 3)\times(1+min(4, 2)) =$

3×(1+2) = 9. The structural information provided by protein source *B* has the highest score in this matched entry and therefore has the most effect on the prediction.

Structure determination

The final structure prediction of the target protein *t* is determined by summing the voting scores of all the protein sources in the matched entries. Specifically, for each amino acid *x* in a protein *t,* we associate three variables, *H*(*x*), *E*(*x*), and *C*(*x*), which correspond to the total voting scores for the amino acid *x* that has structures H, E, and C, respectively. For example, if we assume that the above synonymous word WGPV is aligned with the residues of protein *t* starting at position 11, then protein *A's* contribution to the voting score of *H*(11), *H*(12), *H*(13), and *H*(14) would be 20. Similarly, protein *B* would contribute a voting score of 25 to *H*(11), *H*(12), *C*(13), and *H*(14); and protein *C* would contribute a voting score of 9 to *C*(11), *H*(12), *H*(13), and *H*(14). The structure of *x* is predicted to be *H*, *E* or *C* based on *max*(*H*(*x*), *E*(*x*), *C*(*x*)). When two or more variables have the same highest voting score, C has a higher priority than H, and H has a higher priority than E.

Confidence level

A confidence measure of a prediction for each residue is important to a PSS predictor because it reflects the reliability of the predictor's output. To evaluate the prediction confidence on each amino acid *x*, we calculate a *confidence level* to measure the reliability of the prediction. The confidence level on amino acid *x* is defined as follows:

$$ConLvl(x) = 10 \times \frac{H(x) + E(x) + C(x)}{\sum_{i,t} \left\{ \frac{(freq_t + freq_i)}{2} \times max\left[1, \frac{(Sim_t + Sim_i)}{2}\right] \right\}}$$

The product in the denominator represents a normalization factor for the scoring function. Therefore, the confidence level measures the ratio of the voting scores a residue *x* gets over the summation of the normalization factors. The range of *ConLvl*(*x*) is constrained between 0 and 9 by rounding down. In the Results section (Section 3.2), we analyze the correlation coefficient between the confidence level and the average Q_3 accuracy.

3.1.2 SymPsiPred: a secondary structure meta-predictor

SymPred is different from sequence profile-based methods, such as PSIPRED, which is currently the most popular PSS prediction tool. PSIPRED achieved the top average Q_3 accuracy of 80.6% in the 20 methods evaluated in the CASP4 competition [100]. SymPred and PSIPRED use totally different features and methodologies to predict the secondary structure of a query protein. Specifically, SymPred relies on synonymous words, which represent local similarities among protein sequences and their homologies; however, PSIPRED relies on a position specific scoring matrix (PSSM) generated by PSI-BLAST, which is a condensed representation of a group of aligned sequences. Furthermore, SymPred constructs a protein-dependent synonymous dictionary for inquiries about structural information. In contrast, PSIPRED builds a learning model based on a two-stage neural network to classify sequence profiles into a vector space; thus, it is a probabilistic model of structural types.

It has been shown that combining the prediction results derived by various methods, often referred to as a meta-predictor approach, is a good way to generate better predictions. JPred [101] was the first meta-predictor developed for PSS prediction. After examining the predictions generated by six methods it, JPred returned the consensus prediction result and achieved a 1% improvement over PHD, which was the best single method among the six methods. Similar to the concept of the meta-predictor, we have developed an integrated method called SymPsiPred, which combines the strengths of SymPred and PSIPRED.

To combine the results derived by the two methods, we compare the prediction confidence level of each residue from each method and return the structure with the higher confidence. Since SymPred and PSIPRED use different measures for the confidence levels, we transform their confidence levels into Q_3 accuracies. For each method, we generate an accuracy table showing the average Q_3 accuracy for each confidence level, i.e., we use the average Q_3 accuracy of an SSE to reflect the prediction confidence.

For example, suppose SymPred predicts that a residue in a target sequence has structure H with a confidence level of 6, PSIPRED predicts that the residue has structure E with a confidence level of 6, and the corresponding Q_3 accuracies in the accuracy tables are 77.6% and 64.6% respectively. In this case, SymPsiPred would predict the residue as H.

3.2 Results

In this section, we first reported performance evaluation of SymPred and SymPsiPred on a validation dataset, and then compared our methods with existing methods on EVA benchmark datasets.

3.2.1 Datasets used to develop SymPred

We downloaded all the protein files in the DSSP database [102] and generated three datasets, i.e., *DsspNr-25*, *DsspNr-60*, and *DsspNr-90*, based on different levels of sequence identity using the PSI-CD-HIT program [103] following its guidelines. In other words, *DsspNr-25*, *DsspNr-60* and *DsspNr-90* denote the subset of protein chains in DSSP with mutual sequence identity below 25%, 60% and 90%, respectively, and contain 8297, 12975 and 16391 protein chains, respectively.

3.2.2 Performance evaluation of SymPred and SymPsiPred on the validation set DsspNr-25

We used all the protein chains in *DsspNr-25*, *DsspNr-60* and *DsspNr-90* as template pools to construct the synonymous dictionaries *SynonymDict-25*, *SynonymDict-60* and *SynonymDict-90*, respectively. Furthermore, we used *DsspNr-25* as the validation set to determine the parameters of SymPred by leave-one-out cross validation (LOOCV) since LOOCV (also known as *full jack-knife*) has been shown to provide an almost unbiased estimate of the generalization error [104] and makes the most use the data. (SymPred does not need to rebuild model unlike most machine learning methods when using LOOCV.) Once the parameters of SymPred, including the length *n* of a word and the dictionary, were determined, we also used the validation set *DsspNr-25 to* evaluate the performance of SymPred and SymPsiPred by 10-fold cross validation and LOOCV. To avoid over-estimation of SymPred's performance, when testing each target protein in the *DsspNr-25*, we discarded all the structural information of proteins *t* in the template pool if *t* and the target protein share at least 25% sequence identity.

Choosing the word length 8 with inexact matching criterion and using *SynonymDict-60*, we evaluated the performance of SymPred and SymPsiPred on the validation set *DsspNr-25* by LOOCV and 10-fold cross validation as shown in Table 3. SymPred achieved the Q_3 of 80.5% and the SOV of 75.6% in 10-fold cross validation and the Q_3 of 81.0% and the SOV of 76.0% in LOOCV, outperforming PROSP by at least 5.4% in Q_3 and 6.9% in SOV.

PSIPRED achieved the Q_3 of 80.1% and the SOV of 76.9% on the same test set. However, the prediction performance of PSIPRED might be over-estimated using our dataset because PSIPRED was trained separately. Some protein sequences in our dataset might be in the training set of PSIPRED. Therefore, to have a fair comparison with PSIPRED, we use EVA benchmark datasets. We show the prediction performance with existing methods in the sub-section of 3.2.6.

The meta-predictor, SymPsiPred which integrates the prediction power of SymPred and PSIPRED, achieved a further improvement on Q_3 of 83.9% on *DsspNr-25*. This result demonstrates that SymPsiPred can combines the strengths of the two methods and thus yield much more accurate predictions.

It is noteworthy that SymPred can predict helical structure more accurately than others. The **Q_3Ho** is 84.3% which is much better that **Q_3Eo** and **Q_3Co**. Among the three secondary structure elements, strands (beta sheets) are the most difficult ones to be predicted. Because strands are formed by the pairing of multiple strands held together with hydrogen bonds, they involve interactions between linearly distant residues [105]. Using local sequence or structural information could not predict strands very well. This is one of major challenges and limitations of our method. The **Q_3Eo** of SymPred on *DsspNr-25* is 71.6%, which is lower than **Q_3Ho** by 12.7%, and lower than **Q_3Co** by 6.1%. However SymPsiPred can improve **Q_3Eo** to 75.8% by combining the strength of SymPred and PSIPRED.

Table 3 – Performance comparison of SymPred, SymPsiPred, and PROSP on the *DsspNr*-25 dataset. Q_3Ho (Q_3Eo and Q_3Co, respectively) represents correctly predicted helix (strand and coil, respectively) residues (percentage of helix observed). sovH/E/C values are the specific SOV accuracies of the predicted helix, strand and coil, respectively. SymPred* represents the experiment result using leave-one-out cross validation and SymPred^{+} represents the experiment result using 10-fold cross validation.

***DsspNr-25* (8,297 proteins)**	***Q_3***	***Q_3Ho***	***Q_3Eo***	***Q_3Co***	**sov**	**sov H**	**sov E**	**sov C**
SymPred*	81.0	84.3	71.6	77.7	76.0	82.5	76.9	70.7
SymPred^{+}	80.5	84.1	70.9	77.5	75.6	82.3	76.4	70.3
PSIPRED	80.1	78.8	68.8	78.3	76.9	79.2	74.4	72.2
SymPsiPred	83.9	81.5	75.8	83.9	80.2	82.3	80.3	76.5
PROSP	75.1	79.7	67.6	71.3	68.7	77.0	73.0	63.4

The prediction accuracy of SymPred on *DsspNr-25* was obtained by optimized the two factors: (1) the length of protein words and the matching criterion used for searching the synonymous dictionary and (2) the size of the template pool, as mentioned earlier. Below, we analyze the two factors in more detail and the reported accuracies were obtained by LOOCV.

3.2.3 Factor 1: the word length n and the matching criterion

The choice of word length *n* is a trade-off between specificity and sensitivity, i.e., long words tend to have highly specific structural features and short words increase sensitivity by recovering sequence matches. Regarding the matching, in the previous study of PROSP, we adopted exact matching when searching a synonymous dictionary. Since the exact matching criterion is rather strict in terms of matching efficiency, we also compared the performance of SymPred using exact matching against using inexact matching, which allows at most one mismatched character.

We evaluated the performance of SymPred using the smallest *SynonymDict-25* dictionary. Table 4 shows the Q_3 accuracy of SymPred with exact and inexact matching on different word lengths. The results reveal that the Q_3 accuracy is not always increasing along the increasing word length in both matching mechanisms. The best Q_3 accuracies are reported at n=7 for exact matching and n=8 for inexact matching. That is, 7 identical residues yield high specificity for the structural features and a single *don't-care* character increases the sensitivity to recover sequence matches. In summary, we can improve the prediction performance by using the inexact matching criterion when searching a synonymous dictionary and choosing the word length 8.

Table 4 – The Q_3 accuracies of SymPred using exact and inexact matching on different word lengths.

Word length n	**6**	**7**	**8**	**9**
Q_3 (exact matching)	78.2	80.1	78.1	76.2
Q_3 (inexact matching)	74.9	79.2	80.5	79.0

3.2.4 Factor 2: the effect of the dataset size used to compile a dictionary

Although the estimated theoretical limit of the accuracy of secondary structure assignment is 88%, current state-of-the-art PSS prediction methods achieve around 80% accuracy; there is an 8% accuracy gap. What is the major obstacle to achieving 88% accuracy? Rost [22] raised this question, and Zhou et al. [106] suggested that the size of an experimental database is crucial to the performance. However, Rost found that PHDpsi trained on only 200 proteins was almost as accurate as PSIPRED trained on 2000 proteins, i.e., the performance is insensitive to the size of the training dataset. This is both the strength and the weakness of machine learning-based approaches. Machine learning-based approaches can generate satisfactory prediction models using a limited dataset. On the other hand, the benefit of using more instances is also limited. Though SymPred is not a machine-learning approach, we still concern the relationship between its performance and the size of a template pool.

We fist studied the sensitivity of the data set size by compiling the *SynonymDict-25* using different percentages of the protein sequences in *DsspNr-25*. (The following analysis is based on word length of 8 and using inexact matching in SymPred.) Table 5 summarizes the prediction performance of SymPred using different percentages of proteins in the template pool. The performance improves as the number of template proteins increases. The Q_3 accuracies for 10% and 100% usage of template proteins are 70.8% and 80.5%, respectively, a 9.7% improvement. Moreover, SymPred's performance improves between 0.5% and 2.8% each time the number of template proteins is increased by 10%.With more protein sequences in the template pool, the synonymous dictionary can learn more synonymous words from those sequences and their similar protein sequences.

Table 5 – The Q_3 accuracy comparison of SymPred using dictionaries compiled from different percentages of the template proteins. The performance improves as the number of template proteins increases. SymPred's performance improves between 0.5% and 2.8% each time the number of template proteins is increased by 10%.

Percentage of template pool	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**	**100%**
Number of template proteins	830	1660	2490	3320	4150	4980	5809	6638	7467	8297
Q_3 on *DsspNr-25*	70.8	73.6	75.0	76.3	77.3	78.1	78.7	79.3	79.8	80.5
Improvement	-	+2.8	+1.4	+1.3	+1.0	+0.8	+0.6	+0.6	+0.5	+0.7

Since SymPred is sensitive to the size of the template pool, we next evaluated its performance on *SynonymDict-60* and *SynonymDict-90*, which were compiled from much larger template pools. Table 6 shows SymPred's prediction performance using different-sized template pools. Its prediction accuracy reaches 81.0% on *SynonymDict-60*, a 0.5% improvement over using *SynonymDict-25*. We can learn more useful synonymous words from the additional template proteins. The implication is that if protein *A* and protein *B* are similar, say the two share 50% of sequence identity, then PSI-BLAST can find more similar protein sequences by analyzing *A* and *B* together, rather than separately. For example, there might be a protein *C* that is only similar to protein *B*. In such a case, if *A* is the query sequence, PSI-BLAST would not report protein *C* due to the low sequence identity. However, the advantage decreases when a larger number of similar proteins are involved in the template pool, as shown by the result for *SynonymDict-90,* which is comprised of proteins whose sequence identities are below 90%. The sequence conservation rate contracts to highly similar sequences, and this leads to a bias in the weighted scores of the scoring system. Therefore, we adopt *SynonymDict-60* as the primary synonymous dictionary for making predictions.

Table 6 – Comparison of SymPred's prediction performance on different-sized template pools.

Template pool	***DsspNr-25***	***DsspNr-60***	***DsspNr-90***
Number of template proteins	8297	12975	16391
Synonymous dictionary	*SynonymDict-25*	*SynonymDict-60*	*SynonymDict-90*
Q_3 on *DsspNr-25*	80.5	81.0	80.9

3.2.5 Evaluation of the confidence level

Figure 5 shows the utility of our confidence level and PSIPRED's confidence level in judging the prediction accuracy of each residue in the test set. The statistics are based on more than 2 million residues. The correlation coefficient between the confidence levels and Q3 scores for SymPred is 0.992, and that for PSIPRED is 0.976. Thus, both methods provide strong confidence measures for the output. We observe that a confidence level of 7 or above reported by SymPred is attributed to 53% of the residues with more than 81%

of the Q3 accuracy which is comparable to the confidence level of 8 or above reported by PSIPRED. Furthermore, it can be observed that the prediction of SymPred is more reliable when the confidence levels of both methods are low. For example, the average Q3 score of SymPred for the confidence level of 6 is 77.6%, whereas that of PSIPRED is 64.6%.

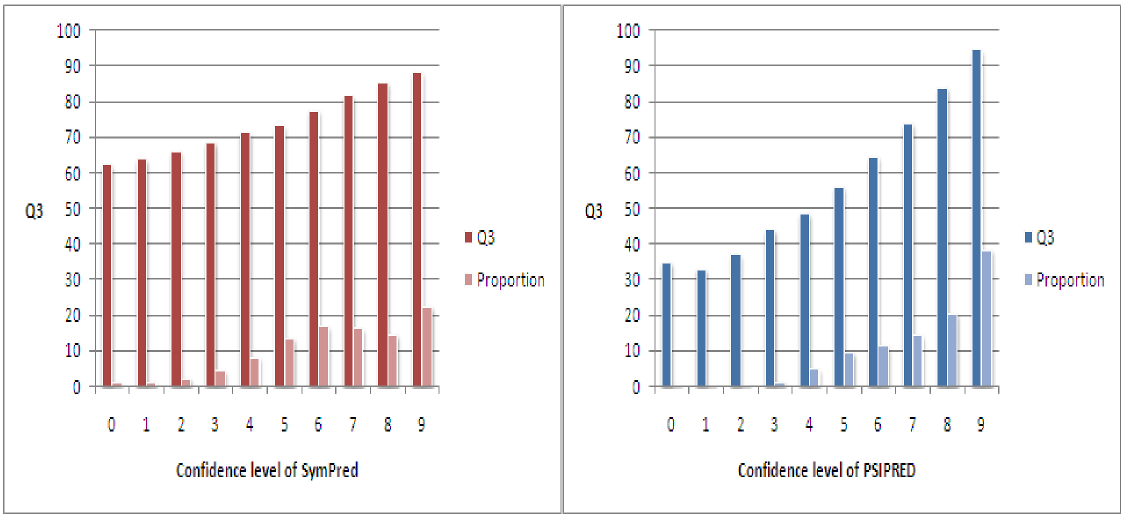


Figure 5 – Relationships between Q_3 accuracy and confidence level on SymPred and PSIPRED. The correlation coefficient between the confidence levels and Q_3 scores for SymPred is 0.992, and that for PSIPRED is 0.976.

3.2.6 Performance comparison with existing methods on EVA benchmark datasets

EVA test sets usually serve as benchmarks of protein secondary structure predictors, particular for CASP competitions [107]. Only proteins without significant sequence identity to previously known PDB proteins were used to test on different existing methods. We downloaded two latest EVA benchmark datasets, called *EVA_Set1* (protein list: http://cubic.bioc.columbia.edu/eva/sec/set_com1.html) and *EVA_Set2* (protein list: http://cubic.bioc.columbia.edu/eva/sec/set_com6.html), the former containing 80 proteins tested on the most number of methods and the latter with the maximum number of proteins (212 proteins). The two datasets serve as independent test sets for performance comparison of SymPred with other existing methods.

For fair comparison, when predicting the secondary structure of each target protein in an independent set, SymPred discarded the structural information of all proteins sharing at least 25% of the sequence identity with the target protein in the template pool, i.e., SymPred used in the template pool the structural information of proteins sharing no more than 25% sequence identity with the target protein.

Table 7 shows the experiment result on the two benchmark datasets, *EVA_Set1* and *EVA_Set2*, where SymPred's results were achieved by using *n*= 8, inexact matching and *SynonymDict-60* It shows that SymPred achieves Q_3 accuracies of 78.8% (SOV=76.4%) and 79.2% (SOV=76.0%), outperforming existing state-of-the-art methods by 1.4% to 5.4%. It can be observed that SymPred performs better than each single predictor on most of performance measurements.

Table 7 – The prediction performance of different methods on the EVA benchmark datasets. sovH/E/C values are the specific SOV accuracies of the predicted helix, strand and coil, respectively. The prediction results of other methods on *EVA_Set1* and *EVA_Set2* are reported at http://cubic.bioc.columbia.edu/eva/sec/common3.html.

EVA_Set1 (80 proteins)	Q_3	*ERRsig* Q_3	sov	*ERRsig* sov	sovH	sovE	sovC
SymPred	78.8	±1.4	76.4	±1.9	85.0	76.5	70.4
SAM-T99sec	77.2	±1.2	74.6	±1.5	80.9	72.5	71.2
PSIPRED	76.8	±1.4	75.4	±2.0	82.1	72.3	69.2
PROFsec	75.5	±1.4	74.9	±1.9	78.3	75.9	71.3
PHDpsi	73.4	±1.4	69.5	±1.9	73.7	73.9	65.2
EVA_Set2 (212 proteins)	Q_3	*ERRsig* Q_3	sov	*ERRsig* sov	sovH	sovE	sovC
SymPred	79.2	±0.9	76.0	±1.2	85.1	77.7	71.3
PSIPRED	77.8	±0.8	75.4	±1.1	80.6	72.6	70.4
PROFsec	76.7	±0.8	74.8	±1.1	79.2	76.2	71.8
PHDpsi	75.0	±0.8	70.9	±1.2	77.0	72.4	67.0

3.3 Discussions

In this section, we analyze the prediction power of SymPred on similar proteins as well as the relationship between the number of synonymous words and the method's prediction performance. We also demonstrate the structure conservation of synonymous words via a case study of a pair of protein sequences that are very dissimilar at the sequence level.

3.3.1 Evaluation on similar proteins

One weakness of machine learning-based methods is that they may under-utilize the structural information in the training set when the query protein has a high sequence similarity to a template in the training set. Therefore, we assess the performance of SymPred when there are sequence similarities between test proteins and proteins in the template pool. Since *SynonymDict-90* contains the largest number of known-structure protein sequences, we conducted an experiment in which we used all the structural information of the template proteins in the dictionary, except the information of the target protein itself. Of the 8297 target proteins, 3585 have similar proteins in the template pool (i.e., the sequence identity ≧25%). SymPred's average Q_3 accuracy on those proteins is 88.1%, which fits the estimated theoretical limit of the accuracy. The result shows that SymPred can utilize the structural information in the template pool effectively when there are sequence similarities to the target protein sequence.

3.3.2 Prediction accuracy affected by enlargement of synonymous words

Although the parameter *b* in PSI-BLAST is set at 500 for searches, not every query protein can have that number of similar proteins in the database used to generate sequence alignments. Because some query proteins are quite unique, PSI-BLAST only reports a few similar proteins at most, and may not report any. In such cases, SymPred would not have enough synonymous words to generate reliable predictions. On the other hand, some query proteins have many highly similar proteins in the database, which results in duplicate synonymous words. Apart from the number of sequence alignments, the number of distinct synonymous words may affect SymPred's performance. Therefore, we analyze the relationship between the number of distinct synonymous words and the SymPred's prediction performance.

To study the relationship, we set different thresholds for selecting corresponding subsets *u* of test protein sequences. The selection criterion is defined as follows. For each test protein *t* in *DsspNr-25*, let *v* denote the number of distinct synonymous words in the word set of *t*, and let *L* be the sequence length of *t* ; then let $e = v/L$, which denotes the multiple of *L* in terms of *v*. If *e* is greater than or equal to a threshold, the protein *t* is added to *u*. We compare the average Q_3 accuracy of proteins in *u* with respect to different thresholds.

Table 8 shows the prediction performance of SymPred and SymPsiPred with respect to different thresholds. The results show that there is a positive correlation between the number of distinct synonymous words and the prediction performance of SymPred and SymPsiPred. For SymPred, the accuracy improves from 81.0% to 83.5% when the threshold increases from $e \geqq 0$ to $e \geqq 150$. It is remarkable that SymPred can predict approximately 75% of the proteins in *DsspNr-25* with 83.1% accuracy, and more than 50% of the protein sequences can be predicted with 83.5% accuracy. For SymPsiPred, the accuracy increases from 83.9% to 85.5% when the threshold increases from $e \geqq 0$ to $e \geqq 150$. The results imply that SymPred and SymPsiPred have the potential to achieve higher accuracy as the number of protein sequences in the NCBInr database increases.

Table 8 – The relationship between the number of distinct synonymous words and the prediction performance. For each test protein *t* of length *L* in *DsspNr-25*, let *v* denote the number of distinct synonymous words of *t*. Define $e = v/L$, the multiplicity of *v* over *L*. If *e* is greater than or equal to a threshold, the protein *t* is selected. The results show that there is a positive correlation between the number of distinct synonymous words and the prediction performance of SymPred and SymPsiPred.

Selection criterion		$e \geqq 0$	$e \geqq 5$	$e \geqq 25$	$e \geqq 50$	$e \geqq 75$	$e \geqq 100$	$e \geqq 125$	$e \geqq 150$
Number of selected proteins		8297	7983	7252	6660	6178	5637	5035	4378
Q_3	SymPred	81.0	81.6	82.3	82.8	83.1	83.3	83.4	83.5
	SymPsiPred	83.9	84.3	84.8	85.1	85.2	85.3	85.4	85.5

3.3.3 Essential Residues

Since the confidence level measures the ratio of voting scores a residue *x* gets to the summation of the normalization factors, it reflects the degree of sequence conservation in protein evolution. We use the confidence levels representing the degrees of importance of residues in determining the structure and function of a protein sequence.

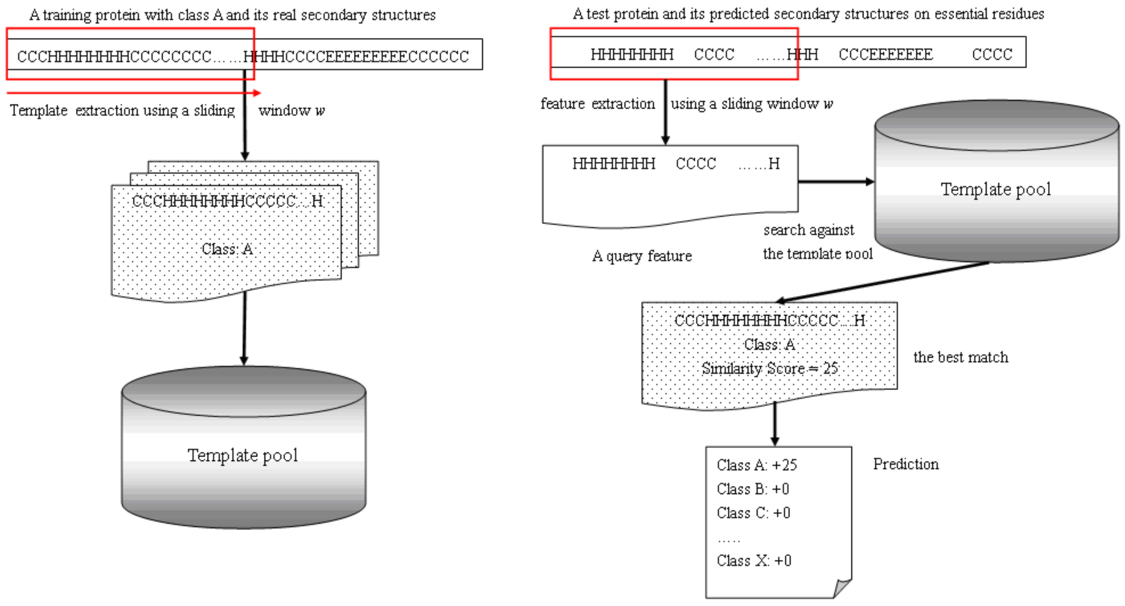
To study the effectiveness of essential residues, we developed a general prediction method, called ProtoPred, which only uses the secondary structural information as the single feature for general proteome prediction problems, such as function prediction and enzyme/non-enzyme classification. The confidence levels are used as weights to indicate the degrees of importance of residues when finding protein templates for the prediction.

ProtoPred: A Prototype of Prediction Method

Figure 6 shows the main algorithm of ProtoPred. ProtoPred is a simple template based method for general prediction problems. It is a standard query-template alignment algorithm that is used frequently in homology modeling or threading methods [108-110].

For the training of ProtoPred, we used a sliding window of size *w* to extract the real secondary structure fragments from each of the training proteins. Each structure fragment carried the related information from its origin, such as function labels or protein classes. These fragments were treated as templates for predictions. For test phase we used the same sliding window to extract the predicted secondary structure fragments from the target protein. Each structure fragment (denoted as *s*) was used to search against the template pool. We compared the similarities between *s* and each template *t* in the template pool. The similarity was estimated as follows.

For each position *x* (from 1 to *w*) if *s*[*x*] was identical to *t*[*x*], then *t* would get a weighted score from *s*, i.e., the confidence level of *s*[*x*]. Each *s* selects the best template *t* with the highest sum of weighted scores (denoted as Sum_{ws}). If the best template *t* was labeled as class *A*, then the target protein would get a score of Sum_{ws} for class *A*. Finally, the target protein would be predicted as the class with the highest score.


A training protein with class A and its real secondary structures
CCCHHHHHHHHHCCCCCCCC……HHHHCCCCEEEEEEEEEECCCCCC
Template extraction using a sliding window w
CCCHHHHHHHHHCCCCC…H
Class: A
Template pool
A test protein and its predicted secondary structures on essential residues
HHHHHHHH CCCC ……HHH CCCEEEEEEE CCCC
feature extraction using a sliding window w
HHHHHHHH CCCC ……H
A query feature
search against the template pool
Template pool
CCCHHHHHHHHHCCCCC…H
Class: A
Similarity Score = 25
the best match
Class A: +25
Class B: +0
Class C: +0
…..
Class X: +0
Prediction

(a) Template extraction

(b) ProtoPred's prediction procedure

Figure 6 – The main algorithm of ProtoPred. (a) Template extraction (b) The prediction procedure.

Experiment Result on Protein Function Prediction Using Essential Residues

The knowledge of protein functions is crucial to the understanding of biological process. Since the experimental procedures for protein function annotation are inherently low throughput, the accurate computational techniques for protein function prediction represent useful tools. Automated protein function prediction methods include direct homology-based and indirect subsequence/feature-based approaches. For the indirect subsequence-based approaches, often only specific subsequences are crucial for the protein to perform its function [109]. This motivated us to use the essential residues in the function predictions.

We downloaded the protein function labels from the Gene Ontology Annotation Database (goa_pdb) [111]. Since we needed to compile a dataset whose protein sequences are not redundant (mutual sequence identity less than 25%) and each of them is of known secondary structure, we then made an intersection set of goa_pdb with *DsspNr-25*. The number of proteins is 2677 and the total number of distinct function labels is 1539. It is worth to note that the function labels contain all GO annotations for the 2677 proteins, including the function labels of biological process, molecular functions, and cellular components. For example, the function labels of protein 1ak6 are 3779 (molecular function: actin binding) and 5622 (cellular component: intracellular).

In this application, we focus on verifying the efficacy of different sources of PSS. These sources are the real secondary structures, the predicted secondary structures of SymPred, and the predicted secondary structure of PSIPRED. ProtoPred predicts the most specific function label among 1539 candidates for a target protein by using one of the sources of secondary structures rather than general functions. The prediction accuracy is 100% if the predicted function label belongs to the target protein, otherwise it is 0%. For example, if we predict 1ak6 as the function 3779 (or 5622) then the accuracy is 100%. The hierarchical structure of GO annotations is not exploited in our prediction method, though it could be used to improve prediction accuracy [6].

ProtoPred extract structure fragments using a sliding window of size w. Table 9 shows the results for several different window sizes. It can be observed that ProtoPred's prediction using the predicted secondary structure of SymPred shows the highest accuracy for all studied window sizes (except the window size of 11 because it is too short to represent the uniqueness of structures for different function classes). For example, for the window size of 51, the prediction accuracies of ProtoPred using the features of real structure, PSIPRED's prediction, and SymPred's prediction are 49.8%, 35.4%, and 57.6% respectively. Notably, the Q_3 of PSIPRED and SymPred on this dataset are 80.3% and 81.1%. Although the performances of PSS prediction of the two methods are similar, the effectiveness is quite different. Moreover, the performance of ProtoPred with SymPred's prediction is also better than that of ProtoPred with real structure. A possible explanation for this discrepancy is that different structures within a protein did not have equal importance for its function. It shows that SymPred could identify the essential residues which are crucial for proteins to perform their functions. Structural identities of low relevance residues dilute the influence of major residues when using the real structure as the feature in the ProtoPred's prediction.

Table 9 – The accuracy (%) of function predictions using different structure sources and different window sizes.

Window Size	11	21	31	41	51	61	71
Real Structure	21.0	21.1	31.5	45.5	49.8	51.8	53.0
PSIPRED	21.0	21.0	23.3	28.9	35.4	40.6	44.0
SymPred	21.0	21.5	39.4	53.8	57.6	58.3	59.1

Experiment Result on Enzyme/non-enzyme classification Using Essential Residues

Many protein function prediction methods focus on only one specific type of functions [112-113]. The problem of enzyme and non-enzyme classifications is a special case of function prediction. We do not have to predict a functional type but only to distinguish between enzyme and non-enzyme. In Dobson and Doig's study, they use multiple features such as secondary structure, amino acid propensities, and surface properties to do the binary classifications. They further divide the features into 52 sub-features and select 36 optimal sub-features for the SVM models to generate the classifier. The overall accuracies are 77.16% and 80.14% for the two different sizes of sub-features, respectively.

We download Dobson and Doig's dataset which contained 1076 proteins. Since SymPred's prediction is the most effective feature among different sources of PSS in the above protein function prediction, ProtoPred uses SymPred's prediction as the input feature for the problem of enzyme and non-enzyme classifications. ProtoPred achieves an overall accuracy 81.8%.

In this application, we only use the secondary structural information for enzyme/non-enzyme classification and achieve a better result. It suggests that the secondary structural information with the essential residue annotation may be sufficient to predict protein functions, which supports the conclusion of Przytycka et al [11].

3.3.4 Sequence alignment by using synonymous words

From the performance of SymPred, we observe that protein-dependent synonymous words possess the property of structure conservation. In other words, the synonymous words show the semantic relationship in terms of protein structures. To further demonstrate the structure conservation property, we compare the synonymous words of two proteins and analyze the shared synonymous words with respect to each residue pair of the two proteins. The distribution of shared synonymous words can help to generate a highly accurate alignment for two protein sequences.

Balibase 3.0 [114], a database that serves as an evaluation resource for sequence alignments, contains manually constructed multiple sequence alignments that are all based on three-dimensional structural superpositions. Therefore, Balibase can be used as a benchmark of sequence alignment tools. We downloaded the first test case (BB11001) and used the first two proteins (1aab and 1j46_A) to demonstrate the structure conservation of synonymous words. The sequence identity of the two proteins is only 16.7%; however, they belong to the same Family (HMG-box) according to the SCOP classification. This indicates that the two proteins are remotely homologous.

Figure 7 shows the distribution of synonymous words shared by the two proteins. The x- and y- axes represent the sequence of 1j46_A and 1aab respectively. A grayscale pixel represents the number of shared synonymous words corresponding to a residue pair (x_i, y_j), where x_i and y_j denote a residue pair comprised of the *i*-th residue of 1j46_A and the *j*-th residue of 1aab respectively. More specifically, if an identical synonymous word *sw* of length *w* is both derived from 1j46_A and 1aab beginning with residue x_i and y_j respectively, then the residue pairs (x_i, y_j), (x_{i+1}, y_{j+1}), …, and (x_{i+w-1}, y_{j+w-1}) are all counted to share *sw*. The darker the pixel, the greater the number synonymous words shared by x_i and y_j.

In Figure 7, Box B is a zoom-in of Box A. We can see that the fourth residue of 1j46_A shares some synonymous words with the first residue of 1aab, the fifth residue of 1j46_A shares more synonymous words with the second residue of 1aab, and so on. It is noteworthy that the Box C shows some residues of 1j46_A shares synonymous words with multiple and continuous residues of 1aab. Since the experiment results suggest that synonymous words are likely expressing similar structures, the Box C implies a possible tolerance of deletions in protein 1aab.

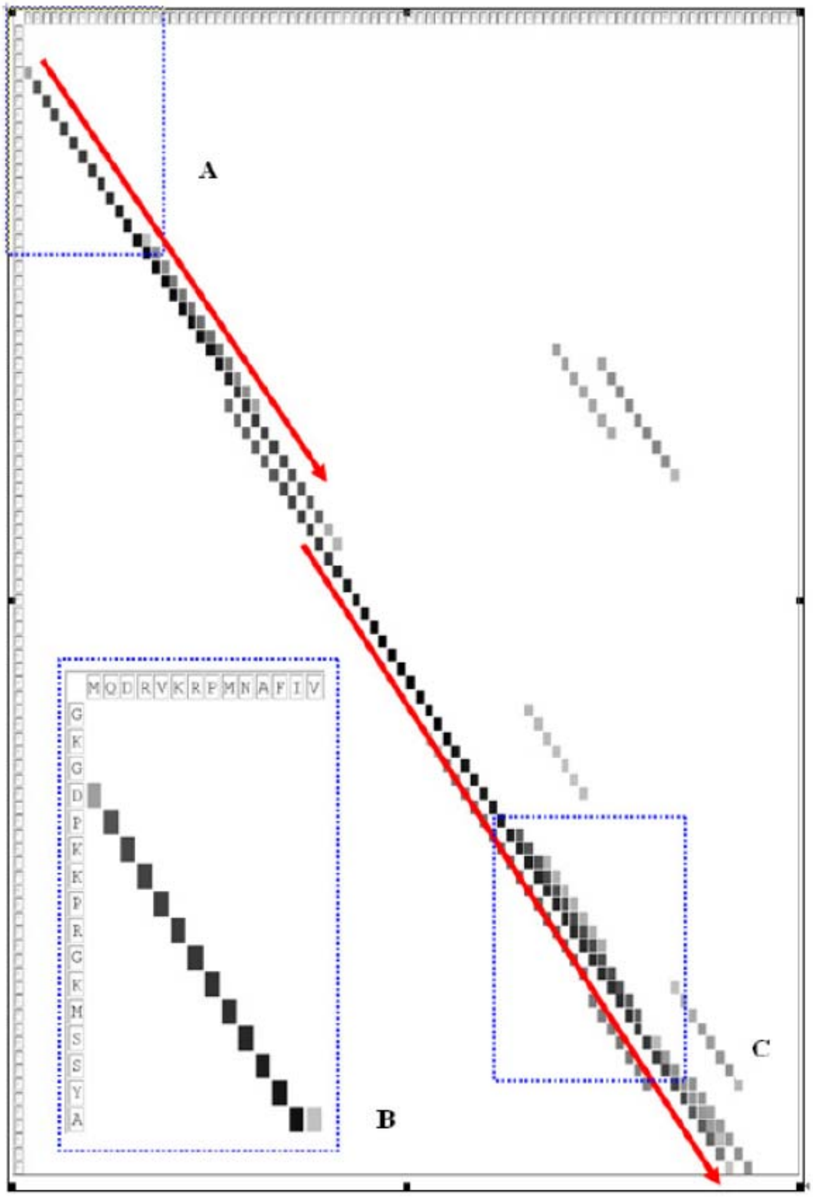


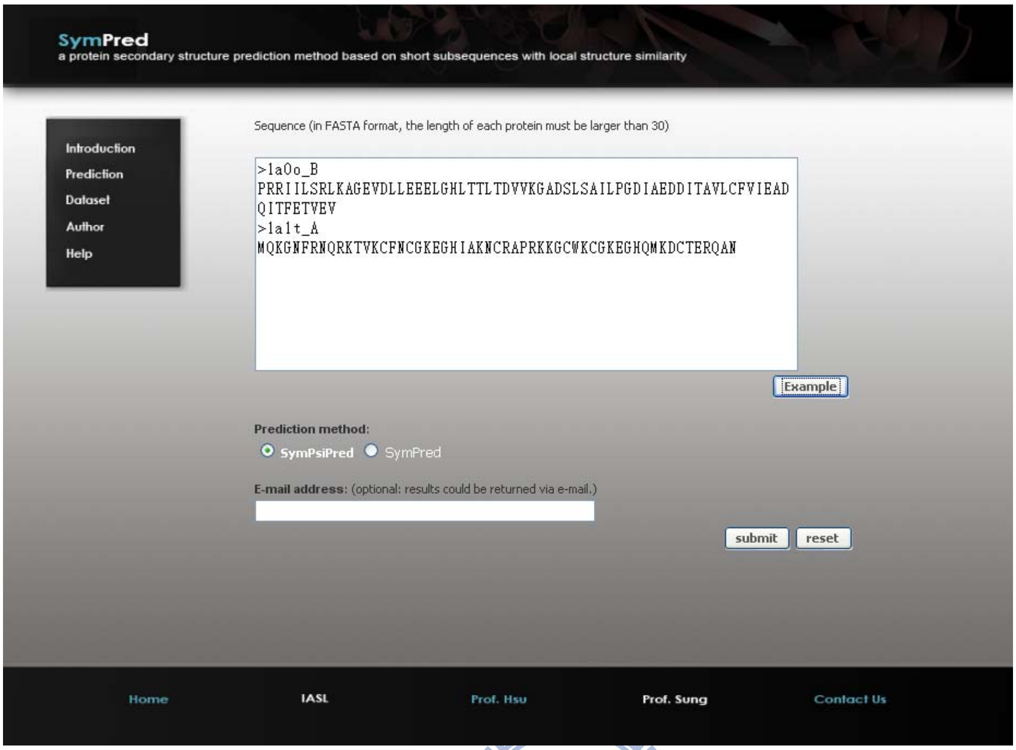
Figure 7 – The distribution of synonymous words shared by 1aab and 1j46_A. The x- and y-axes represent the sequence of 1j46_A and 1aab respectively. A grayscale pixel represents the number of shared synonymous words corresponding to a residue pair (x_i, y_j), where x_i and y_j denote a residue pair comprised of the *i*-th residue of 1j46_A and the *j*-th residue of 1aab respectively. Box B is a zoom-in of Box A. The red lines indicate the alignment based on the number of shared synonymous words, and the alignment is very close to that reported in Balibase for the two proteins. Notably, it can be observed that the path of the darker pixels is nearly perfectly matched the suggested alignment.

We align the two sequences based on the distribution of synonymous words shared by the two sequences. Instead of using a substitution matrix to calculate the score of an aligned residue pair, we use the number of shared synonymous words between a residue pair since the number of shared synonymous words can reflect both the sequence and the structure similarities of a residue pair. As a result, it generates an alignment indicated by the red lines shown in the figure, i.e., the fourth residue of 1j46_A is aligned with the first residue of 1aab, the fifth residue of 1j46_A with the second residue of 1aab, etc, and there are two gaps in the midst of the alignment. (The red lines are drawn shifted a little bit in order to avoid overlapping the dark pixels.) Notably, the resulting alignment is very close to the alignment reported in Balibase for the two proteins, matching 76 out of 78 correct residues pairs, i.e., 97% of alignment accuracy, while ClustalW aligns 64 out of 78 residue pairs (82.1% accuracy) correctly. More examples of highly accurate alignment by using synonymous words could be found in other protein pairs. Overall speaking, the distribution of shared synonymous words could indicate three-dimensional structural superpositions as well as the possible alignment of a protein sequence pair.

3.4 Availability

A major limitation of our synonymous dictionary based approach is that the storage of synonymous dictionary takes a lot of space. For the consideration of efficiency, we implement SymPred and SymPsiPred as parallel programs in a pc-cluster framework. To provide prediction service for the public domain, SymPred and SymPsiPred are also implemented as web servers. They accept either single sequence or multiple sequences and predict the secondary structure of the query protein(s). The web servers are available at http://bio-cluster.iis.sinica.edu.tw/prospref/. Figure 8 shows a screenshot of SymPred web server.

The sequence input should be in fasta format and the sequence length of each of query protein should be longer than 30 in order to have significant sequence alignment when performing a PSI-BLAST search. If an E-mail address is assigned, the prediction result of each query protein will be sent to the user immediately when the prediction is completed.



SymPred
a protein secondary structure prediction method based on short subsequences with local structure similarity

Sequence (in FASTA format, the length of each protein must be larger than 30)

```
>1a0o_B
PRRIILSRLKAGEVDLLEEELGHLTTLTDVVKGADSLSAILPGDIAEDDITAVLCFVIEAD
QITFETVEV
>1a1t_A
MQKGNFRNQRKTVKCFNCGKEGHIAKNCRAPRKKGCWKCGKEGHQMKDCTERQAN
```

Prediction method:
● SymPsiPred ○ SymPred

E-mail address: (optional: results could be returned via e-mail.)

Figure 8 – The SymPred and SymPsiPred web servers. We accept either single sequence or multiple sequences and predict the secondary structure of the protein(s).

3.5 Summaries

In this study, we have proposed an improved dictionary-based approach called SymPred for PSS prediction. We have also presented a meta-predictor called SymPsiPred, which combines a dictionary-based approach (SymPred) and a machine learning-based approach (PSIPRED). Tests on a proteome-scale dataset of 8297 protein chains show that the overall average Q_3 accuracy of SymPred and SymPsiPred is 81.0% and 83.9% respectively. Through the blind test on the two independent test sets, SymPred achieves the average Q_3 accuracies of 78.8% and 79.2% respectively, which are better than other state-of-the-art PSS predictors. SymPred can be regarded as a special case of a template-based approach because it predicts PSS by finding template sequences based on local similarities, i.e., synonymous words. However, the accuracy gap between the template-based methods and machine learning-based methods is approximately 10%. We show that SymPred can reduce that gap by using n-gram patterns.

From the analysis of two factors, we find that the prediction accuracy of SymPred can be gradually improved based on each factor's optimization. In particular, SymPred is very sensitive to the size of the template pool, as shown by the fact that its performance improves between 0.5% and 2.8% each time the number of template proteins is increased by 10%. Therefore, the performance accuracy will improve further as the number of known-structure proteins increases. Furthermore, from the analysis of the number of distinct synonymous words, we posit that, as the number of protein sequences of unknown structures increases in the NCBInr database, we will be able to discover more sequence variations and derive more synonymous words to improve SymPred's performance. The average Q_3 accuracy of SymPred is above 83% for proteins that have synonymous words satisfying $e \geqq 75$. Meanwhile, the Q_3 accuracy of SymPsiPred is above 85%, which is even closer to the estimated theoretical limit of PSS prediction accuracy. The results imply that SymPred and SymPsiPred have the potential to achieve higher accuracy as the number of protein sequences in the PDB database and the NCBInr database increases.

When SymPred is tested on proteins that have sequence similarities to the template proteins, the average Q_3 accuracy is approximately 88%. The result shows that SymPred can utilize the structural information in the template pool effectively. We also demonstrate the power of synonymous words in the sequence comparisons. The information about shared synonymous words can be used to infer three-dimensional structural superpositions. The experiments and the analysis results indicate that synonymous words are reliable short templates that can provide protein-related information.

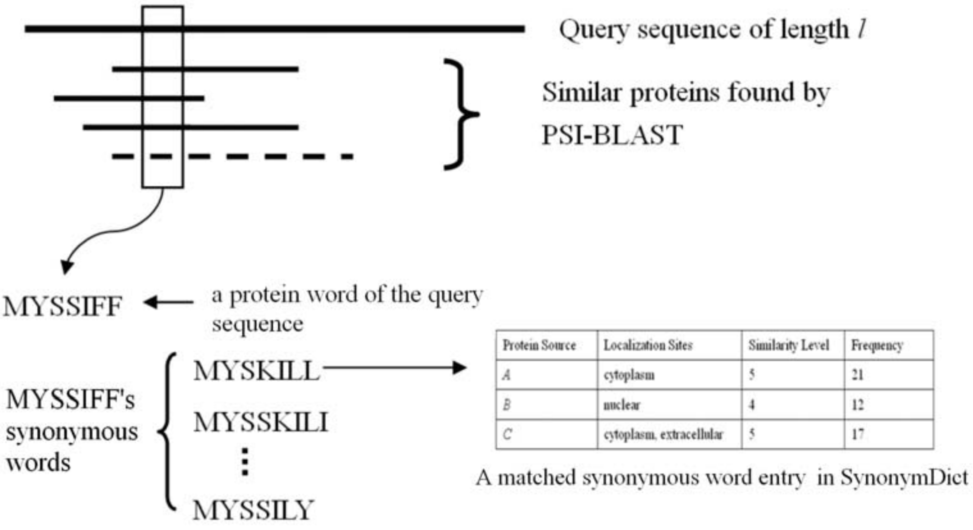
A major advantage of dictionary-based methods is that the prediction process is transparent and easy to understand. Unlike machine learning-based methods, which are computationally intractable, we can examine the prediction process to observe how SymPred generates predictions, including the synonymous words it matches against the dictionary and the template proteins involved in the prediction process. To differentiate the prediction model from machine learning-based methods, it is often referred to as a black box model. Another major advantage of dictionary-based methods is that adding more proteins with known structures is much easier than under machine learning-based methods. Unlike most machine learning-based methods, which need to retrain the prediction models, the proposed dictionary-based method can be expanded incrementally by simply adding new synonymous words or by updating existing entries with new protein sources and the associated structural information.

Chapter 4 Protein Subcellular Localization Prediction

4.1 Methods

4.1.1 KnowPred$_{site}$: a localization prediction method based on SynonymDict

The main idea of KnowPred$_{site}$ is illustrated in Figure 9. Given a target protein *t*, whose localization annotation is unknown and to be predicted, we perform PSI-BLAST search and compile a word set of *t*. Each word *sw* is then matched against words in *SynonymDict*, and the synonymous word entry with index *sw* is called a *hit*.


Query sequence of length *l*
Similar proteins found by PSI-BLAST
MYSSIFF
a protein word of the query sequence
MYSSIFF's synonymous words
MYSKILL
MYSSKILI
⋮
MYSSILY
A matched synonymous word entry in SynonymDict

Protein Source	Localization Sites	Similarity Level	Frequency
A	cytoplasm	5	21
B	nuclear	4	12
C	cytoplasm, extracellular	5	17

Figure 9 – The prediction procedure of KnowPred$_{site}$.

For each hit, we calculate two types of scores associated with each localization site *i*: the voting score s_i and the confidence score *CS*(*i*). The calculation of the voting score s_i is as follows: Let *f* denote the frequency of *sw* found in all *t*'s high-scoring segment pairs (HSPs). For each synonymous word entry in *SynonymDict*, we calculate the score loc_i associated with each localization site by summing up the frequencies of the synonymous words that contain the specific site. For example, for the peptide record MYSKILL shown in Table 2, the score of cytoplasm is 38 (21+17; since protein source *A* and *C* are both localized into cytoplasm), and those of nuclear and extracellular are 12 and 17, respectively. Then the voting score s_i is defined as *f* multiplied by (loc_i / total frequencies in that record). For example, if MYSKILL is a synonymous word of *t* and its frequency is 10 in *t's* HSPs, then the voting scores of cytoplasm, nuclear, and extracellular are 7.6 (=10×38/50), 2.4 (=10×12/50), and 3.4 (=10×17/50), respectively, while those of other localization sites are all 0.

The localization site prediction of the protein *t* is determined by the confidence score *CS*(*i*), which is the total voting score aggregated from all hit records. Finally, each *CS*(*i*) is divided by the summation of all frequencies *f* of all *t'*s hits and then multiplied by 100 to normalize the confidence score in the range of 0 and 100. $\text{KnowPred}_{\text{site}}$ predicts *t* being localized into the site with the highest confidence score for single-localized proteins or into the sites with the two highest confidence scores for multi-localized proteins (All multi-localized proteins in ngLOC dataset have two localization sites).

To differentiate single-localized proteins from those that are multi-localized, we followed King and Guda's method [54] to calculate the multi-localized confidence score (*MLCS*) associated with a protein *t*, which gives a relative measure of the likelihood that the protein *t* is multi-localized. It is derived from the two highest confidence scores (denoted as CS_1 and CS_2) and is defined as follows.

$$MLCS(t) = (CS_1 + CS_2) - \frac{(CS_1^2 - CS_2^2)}{100.0},$$

and *MLCS*(*t*) is bounded by 100, i.e., when the calculated *MLCS*(*t*) is over 100, it is assigned 100.

4.1.2 Best BLAST prediction method

Since BLAST is the most popular method for sequence comparison, we implemented a simple prediction method based on the BLAST search result. Given a dataset of proteins with known localization site(s), to predict the localization site(s) of a test protein *t* we first perform the BLAST search against the dataset and then assign the localization annotations of the best BLAST hit to the protein *t*. If there is no hit at the e-value cutoff 0.001, no annotation will be assigned to the protein *t*. As reported by Jones and Swindells, the e-value of 0.001 generally produces a safe searching [99]. The performance of BLAST-based prediction method is usually treated as the baseline to compare with those of other methods [115].

4.1.3 Evaluation measure

The performance is estimated using the following measurements. To assess the performance in each localization site, precision, accuracy and Matthew's correlation coefficient (*MCC*) are calculated by Equations (2) and (3), respectively. The overall accuracy is defined in Equation (4).

$$Precision = \frac{TP_i}{TP_i + FP_i} \times 100\% \tag{1}$$

$$Accuracy_i = \frac{TP_i}{N_i} \times 100\% \tag{2}$$

$$MCC_i = \frac{TP_i \times TN_i - FP_i \times FN_i}{\sqrt{(TP_i + FP_i)(TP_i + FN_i)(TN_i + FP_i)(TN_i + FN_i)}} \tag{3}$$

$$Accuracy = \left(\sum_{i=1}^{10} TP_i \Big/ \sum_{i=1}^{10} N_i \right) \times 100\% \tag{4}$$

where *TPi*, *TNi*, *FPi*, *FNi*, and *Ni* are, respectively, the number of true positives, true negatives, false positives, false negatives, and proteins in localization site i. *MCC*, which considers both under- and over-predictions, provides a complementary measure of the predictive performance, where *MCC* = 1 indicates a perfect prediction, *MCC* = 0 indicates a completely random assignment, and *MCC* = -1 indicates a perfectly reverse correlation.

4.2 Results

KnowPred$_{\text{site}}$ was implemented as a parallel program under the Linux environment. It was implemented using C++ and MPICH library. We used the ngLOC dataset [54] to compile the synonymous dictionary and test the performance of KnowPred$_{site}$. The dataset is compiled from 1923 different species and contains 28056 protein sequences, including 25887 single localized proteins and 2169 multi-localized proteins. There are ten different subcellular locations among these proteins, which are Cytoplasm (CYT), Cytoskeleton (CSK), Endoplasmic Reticulum (END), Extracellular (EXC), Golgi Apparatus (GOL), Lysosome (LYS), Mitochondria (MIT), Nuclear (NUC), Plasma Membrane (PLA), and Perixosome (POX).

We conducted two types of experiment on the dataset. First, in order to take advantages of local similarities from as many proteins as possible, we conducted the leave-one-out cross validation experiment to determine the parameters and to evaluate the performance of KnowPred$_{site}$. In this experiment, each protein was in turn used as the test protein and the remaining 28055 proteins were used to compile the synonymous dictionary. Second, we compared the performance of KnowPred$_{site}$ with existing methods. Since the dataset is from ngLOC and ngLOC has been shown to be better than PSORT [116], pTARGET [117] and PLOC [118] using the same dataset, we directly compare KnowPred$_{site}$ against ngLOC using ten-fold cross validation. In this experiment, all proteins are partitioned into 10 subsets, and each subset was in turn used as the test set and the remaining nine subsets were used to compile the synonymous dictionary.

4.2.1 Determining window size *w* and similarity threshold *k* for KnowPred$_{\text{site}}$

KnowPred$_{\text{site}}$ aims to utilize the localization annotations of synonymous words. The determination of semantic relations, which depends on the window size *w* and the threshold of similarity level *k*, can affect the performance of KnowPred$_{\text{site}}$. Using a smaller *w*, synonymous words have a higher probability to be hit against words in the synonymous dictionary; however, shorter synonymous words are likely to appear in many unrelated proteins. Given a fixed *w*, there is also a trade-off in choosing the threshold of similarity level *k*. A smaller *k* produces looser semantic relations, which leads to extracting more, but less reliable, synonymous words. To make an appropriate selection of *w* and *k*, we conducted a leave-one-out cross validation experiments on only the single-localized proteins in the ngLOC dataset for *w* ranging from 3 to 11 and *k* ranging from 0 to *w*.

Figure 10 shows the overall accuracies of KnowPred$_{\text{site}}$ using different window size *w* with fixed similarity threshold ($k = 0$). It shows that the appropriate window size is 7 or 8. Then we further investigate the performance using different thresholds of similarity levels.

Table 10 shows the overall accuracies ranging from 90.9% to 92.0% for all combinations of window sizes (*w* = 7, 8) and similarity thresholds. According to the experiment results, we chose the combination of *w* = 7 and *k* = 6 for the following experiments since they provided the best accuracy 92.0%.

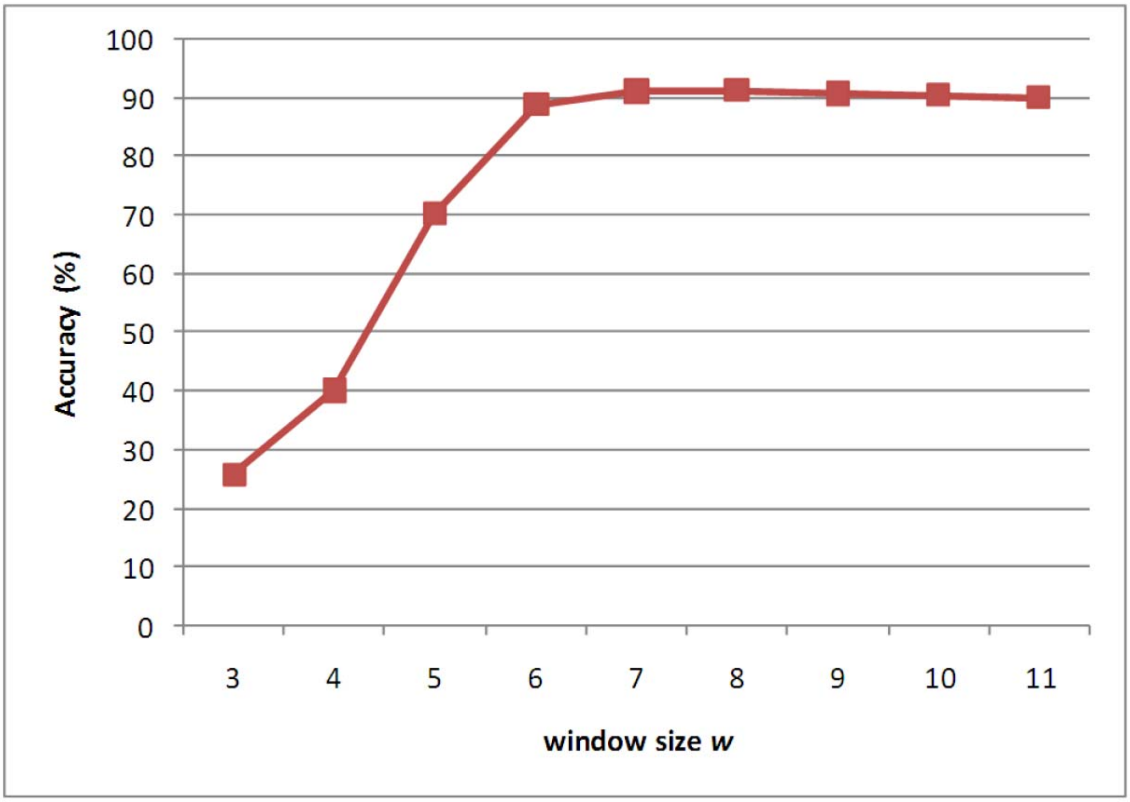


Figure 10 – The overall accuracies of KnowPred$_{site}$ using different size of word length.

Table 10 – The overall accuracies using different thresholds of similarity levels for window size 7 and 8. The combination of w = 7 and k = 6 provides the best accuracy. Some results are shown to have identical overall accuracies due to the rounding off to the first decimal place.

Similarity Level Threshold *k*	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Overall Accuracy (%) *w* = 7	91.2	91.2	91.3	91.4	91.5	91.8	92.0	91.6	—
Overall Accuracy (%) *w* = 8	91.4	91.4	91.4	91.4	91.4	91.5	91.6	91.7	90.9

4.2.2 Prediction performance of KnowPred$_{\text{site}}$

After the best parameters have been determined, we conducted a ten-fold cross validation experiment on the entire dataset to compare KnowPred$_{\text{site}}$ with ngLOC and Blast-hit prediction. We used the top *N* accuracy for evaluation, where *N* ranges from 1 to 4. A protein is considered to be correctly predicted when the real localization site(s) rank among the top *N* of the predicted sites. (Top 1 accuracy is simply the *Accuracy* defined in Equation (4).) Notably, for multi-localized proteins, the accuracy is measured in two ways: first, at least one site correctly predicted and second, both sites correctly predicted. Using the first measurement, a true positive is a multi-localized protein with at least one localization site correctly predicted; whereas a true positive using the second measurement is a multi-localized protein with both sites correctly predicted.

The prediction performance of KnowPred$_{\text{site}}$, ngLOC, and Blast-hit is summarized in

Table 11, in which KnowPred$_{site}$ performance is reported with ten-fold cross validation and leave-one-out cross validation as denoted by $^{\#}$KnowPred$_{site}$ and *KnowPred$_{site}$, respectively. It is observed that KnowPred$_{site}$ outperforms ngLOC and Blast-hit.

For single-localized proteins, the overall accuracies of KnowPred$_{site}$ are from 91.7 to 98.1 when the correct prediction is considered within the top 1 to top 4 most probable sites. Those of ngLOC are from 88.8% to 96.3%. The accuracy of Blast-hit is 86.0%, which means 86.0% of single-localized proteins could be correctly predicted by BLAST searches. It is noteworthy that 2114 sequences among all single-localized proteins failed to find significant similar proteins by Blast-hit method; however, 58.8% of them were correctly predicted by KnowPred$_{site}$. It shows that the local similarity helps identify related sequences for subcellular localization prediction.

Table 11 – Prediction performance of $\text{KnowPred}_{\text{site}}$, ngLOC, and Blast-hit. $^{*}\text{KnowPred}_{\text{site}}$ represents the experiment result using leave-one-out cross validation; $^{\#}\text{KnowPred}_{\text{site}}$ represents the experiment result using 10-fold cross validation.

***Overall Accuracy* (%)**	**Methods**	**Top 1**	**Top 2**	**Top 3**	**Top 4**
Single-localized	$^{*}\text{KnowPred}_{\text{site}}$	92.0	95.7	96.8	98.1
	$^{\#}\text{KnowPred}_{\text{site}}$	91.7	95.4	96.6	97.9
	ngLOC	88.8	92.2	94.5	96.3
	Blast-hit	86.0	—	—	—
Multi-localized (at least 1 correct)	$^{*}\text{KnowPred}_{\text{site}}$	90.8	96.4	98.2	98.9
	$^{\#}\text{KnowPred}_{\text{site}}$	90.1	96.1	98.1	98.9
	ngLOC	81.9	92.0	96.1	97.4
	Blast-hit	78.8	—	—	—
Multi-localized (both correct)	$^{*}\text{KnowPred}_{\text{site}}$		74.3	83.3	88.7
	$^{\#}\text{KnowPred}_{\text{site}}$		72.1	82.2	87.5
	ngLOC		59.7	73.8	83.2
	Blast-hit		45.7	—	—

The experiment result shows that $\text{KnowPred}_{\text{site}}$ has much higher accuracy on multi-localized proteins than the other methods. Using the first accuracy measurement, i.e., at least one site correctly predicted, $\text{KnowPred}_{\text{site}}$ achieves more than 90% of the top 1 accuracy, which is higher than ngLOC by 8.2%. Using the tighter second accuracy measurement, $\text{KnowPred}_{\text{site}}$ achieves 72.1% of the top 2 accuracy, which is higher than ngLOC by 12.4%. Further observing the top N accuracy, we find that $\text{KnowPred}_{\text{site}}$ is more able to narrow down the number of false positives than ngLOC.

The top 1 and top 2 accuracies of the Blast-hit method are 78.8% and 45.7% for the two accuracy measurements. Notably, 318 proteins among all multi-localized proteins failed to find any significant Blast hit; however, 73.3% and 49.7% of them were correctly predicted by KnowPred$_{site}$ using the two accuracy measurements, respectively.

4.2.3 Site-specific prediction performance

In contrast to the overall accuracy of the dataset reported in Table 11, we further analyze the prediction performance on each of the 10 distinct localization sites. The results are summarized in Table 12. Among the 10 localization sites, the precision ranges from 75.7% to 98.5% and the *Accuracy$_i$* ranges from 52.0% to 96.4%. It is observed that higher occurrence of the localization site, e.g., EXC (29.1%) and PLA (25.2%), leads to better prediction, e.g., the precision and accuracy on EXC are 98.5% and 93.9%, respectively. Low occurrence of the localization site could deteriorate prediction, for example, CSK (1%) and GOL (1.1%) have *MCCi* of 0.645 and 0.746, respectively. However, if the synonymous words of a site have higher specificity, prediction performance could be good despite low occurrence. For example, the precision and accuracy on LYS (0.6%) and POX (0.8%) are 87.2% and 81.9%, and 87.3% and 85.1%, respectively. Furthermore, it is noteworthy that although CYT represents 11.1% of the dataset, its *MCCi* is 0.774, much lower than other highly occurring sites. Its low *MCCi* is due to low precision since KnowPred$_{site}$ yields more false positives for CYT. High false positives usually occur when the synonymous word entries of a site have lower specificity and higher diversity. As a result, proteins of other localization sites are misclassified as CYT.

Table 12 – Prediction performance of KnowPred$_{site}$ for each site using precision, accuracy, and MCC.

Site *i*	**Occurrence in the dataset (%)**	***Precision* (%)**	***Accuracy$_i$* (%)**	***MCC$_i$***
CYT	11.1	75.7	84.4	0.774
CSK	1.0	81.1	52.0	0.645
END	3.6	92.9	84.1	0.88
EXC	29.1	98.5	93.9	0.946
GOL	1.1	79.1	70.9	0.746
LYS	0.6	87.2	81.9	0.844
MIT	9.4	96.7	86.9	0.907
NUC	18.0	87.3	93.8	0.884
PLA	25.2	94.4	96.4	0.938
POX	0.8	87.3	85.1	0.861

Figure 11 shows the site-specific comparison between KnowPred$_{site}$ and ngLOC in terms of accuracy and *MCC*. KnowPred$_{site}$ outperforms ngLOC in eight localization sites (CSK, END, EXC, GOL, MIT, NUC, PLA, POX) in terms of *MCC*. The two sites where ngLOC performs better are CYT (0.777 for ngLOC and 0.774 for KnowPred$_{site}$) and LYS (0.902 for ngLOC and 0.844 for KnowPred$_{site}$). In terms of accuracy, KnowPred$_{site}$ outperforms ngLOC in all sites except for LYS (represents around 0.6% of the whole dataset), where ngLOC and KnowPred$_{site}$ yields 85.5% and 81.9% of accuracy, respectively.

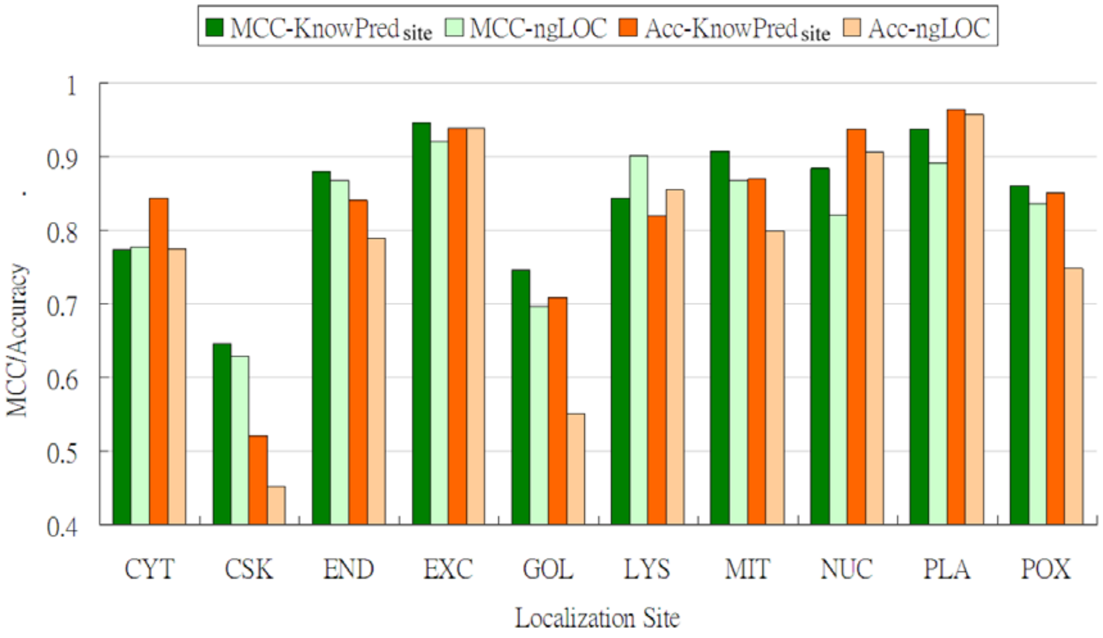


Figure 11 – Matthew's correlation coefficient (*MCC*) and accuracy comparison between KnowPred$_{site}$ and ngLOC.

4.2.4 Evaluation of the multi-localized confidence score (MLCS)

A significant number of eukaryotic proteins are known to be localized into multiple subcellular organelles; therefore, it is important to differentiate single-localized proteins from multi-localized proteins. We used the entire ngLOC dataset to compare different MLCS thresholds on the correct distinction between single-localized and multi-localized proteins. Specifically, we used the portions of true positives in the multi-localized proteins and true negatives in the single-localized proteins as the performance measures. A true positive represents a multi-localized protein whose MLCS is above the threshold and a true negative represents a single-localized protein whose MLCS is below the threshold.

We illustrate the cumulative percentages of true positive and true negative versus the MLCS threshold in Figure 12, which shows that the true negative curve is increasing along the MLCS axis whereas the true positive curve is decreasing. If the MLCS threshold is set to be 40, 60.7% of multi-localized proteins are true positives and 96.5% of single-localized proteins are true negatives. It shows that 60.7% of multi-localized proteins obtained MLCS of 40 or better, whereas only 3.5% of single-localized proteins within this range. If the MLCS threshold is set to be 20, 86.3% of multi-localized proteins are true positives and 82.8% of single-localized proteins are true negatives. In ngLOC, the best result shows that 76% of multi-localized proteins belong to true positives and 81% of single-localized proteins belong to true negatives when 40 of MLCS threshold is applied. The result shows that KnowPred$_{site}$ better differentiate multi-localized proteins from those that are single-localized.

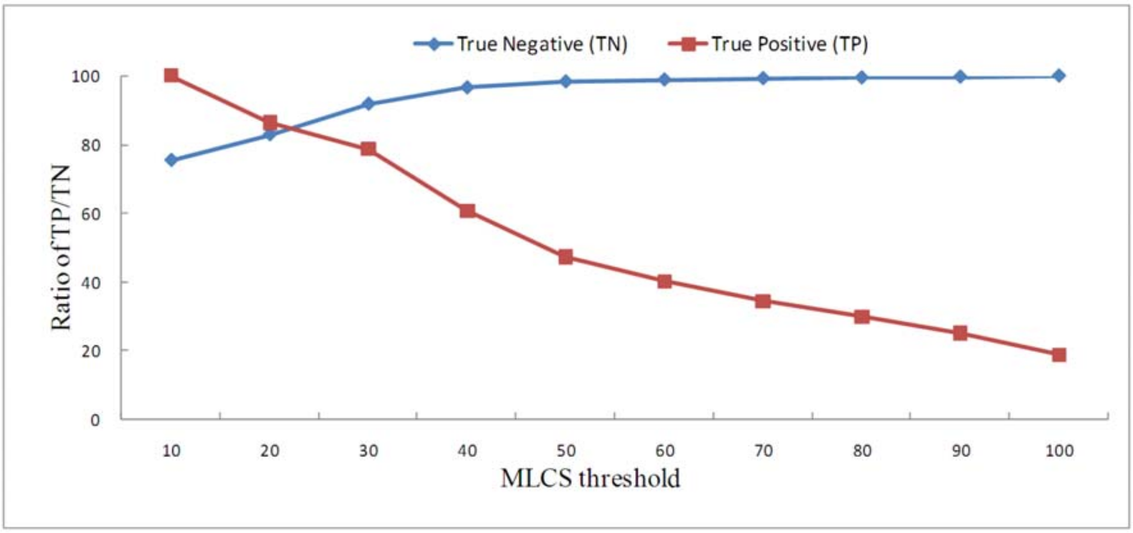


Figure 12 – MLCS analysis. A true positive represents a multi-localized protein whose MLCS is above the threshold and a true negative represents a single-localized protein whose MLCS is below the threshold. We compare the ratio of true positives/true negatives to the total number of multi-/single-localized proteins.

4.3 Discussions

Unlike most machine learning methods that the parameters of the prediction models are not biologically explainable, the prediction result of KnowPred$_{site}$ is explainable and the prediction process is transparent and traceable. To predict the localization sites of a protein, KnowPred$_{site}$ can show the template sequences and their associated contributive confidence scores for a query protein. Such information is useful for interpretation of the prediction results. In this section, we select the four sequences EF1A2_RABIT, RASH_HUMAN, MCA3_MOUSE, and CFDP2_BOVIN from the ngLOC dataset, to demonstrate the interpretation of KnowPred$_{site}$ prediction results.

The prediction result of each of the first three proteins and its template sequences extracted from the synonymous dictionary used for prediction are shown in Table 13 to Table 15, respectively. In each table, the prediction result shows the MLCS and the confidence score of each localization site that the query protein would be localized into. Moreover, the template proteins which are used to vote for the localization sites are shown in each table. We only list the top eight template proteins which contribute most to the confidence scores of the query sequence. For each template sequence, its contribution to confidence score of each localization site and the sequence identity to the query protein calculated by ClustalW (denoted by SI) are shown.

In the example of EF1A2_RABIT shown in Table 13, KnowPred$_{site}$ predicts it being single-localized at cytoplasm (CYT) since MLCS is very low (7.40) and CYT has the highest confidence score. However, the localization site of EF1A2_RABIT reported in the ngLOC dataset is nuclear (NUC). Examining the eight template proteins, we find that they all have high sequence identities with EF1A2_RABIT and most of them are localized into CYT except EF1A2_RAT localized into NUC. According to the Gene Ontology annotation, it is localized into CYT and NUC, which are the two sites with the highest confidence scores in KnowPred$_{\text{site}}$’s prediction.

Table 13 – Prediction result of EF1A2_RABIT.

Query	**CYT**	**CSK**	**END**	**EXC**	**GOL**	**LYS**	**MIT**	**NUC***	**PLA**	**POX**	**MLCS**
EF1A2_RABIT	95.45	0	0	1.45	0	0	0.04	2.97	0.05	0	7.40
Template	**CYT**	**CSK**	**END**	**EXC**	**GOL**	**LYS**	**MIT**	**NUC**	**PLA**	**POX**	**SI**
EF1A2_RAT	0	0	0	0	0	0	0	2.94	0	0	99.78
EF1A_CHICK	2.77	0	0	0	0	0	0	0	0	0	92.22
EF1A1_HUMAN	2.75	0	0	0	0	0	0	0	0	0	92.22
EF1A1_RAT	2.75	0	0	0	0	0	0	0	0	0	92.22
EF1A0_XENLA	2.69	0	0	0	0	0	0	0	0	0	90.06
EF1A_BRARE	2.64	0	0	0	0	0	0	0	0	0	90.06
EF1A2_XENLA	2.64	0	0	0	0	0	0	0	0	0	88.79
EF1A3_XENLA	2.60	0	0	0	0	0	0	0	0	0	88.55

*: correct answer; SI: sequence identity.

In the example of RASH_HUMAN shown in Table 14, $\text{KnowPred}_{\text{site}}$ predicts RASH_HUMAN being localized into plasma membrane (PLA) and cytoplasm (CYT). However, the correct localization site is cytoplasm and Golgi apparatus (GOL). Referring to the prediction result, the confidence score of PLA is much higher than those of CYT and GOL. It is also observed that most of the template proteins are localized into PLA. According to the annotation in Gene Ontology and SwissProt, RASH_HUMAN is localized into PLA and GOL, and the template protein, RASN_HUMAN, is also localized into PLA and GOL. If applying the new annotation data, $\text{KnowPred}_{\text{site}}$ can predict RASH_HUMAN correctly.

Table 14 – Prediction result of RASH_HUMAN.

Query	**CYT***	**CSK**	**END**	**EXC**	**GOL***	**LYS**	**MIT**	**NUC**	**PLA**	**POX**	**MLCS**
RASH_HUMAN	18.95	0.06	0.09	0.09	13.74	0.04	0.24	0.25	83.61	0	36.24

Template	**CYT**	**CSK**	**END**	**EXC**	**GOL**	**LYS**	**MIT**	**NUC**	**PLA**	**POX**	**SI**
RASK_HUMAN	0	0	0	0	0	0	0	0	13.88	0	86.32
RASK_MOUSE	0	0	0	0	0	0	0	0	13.81	0	86.32
RASN_HUMAN	13.19	0	0	0	13.19	0	0	0	0	0	85.19
LET60_CAEEL	0	0	0	0	0	0	0	0	10.55	0	74.07
RAS3_RHIRA	0	0	0	0	0	0	0	0	5.05	0	57.07
RAS1_RHIRA	0	0	0	0	0	0	0	0	4.88	0	58.62
RAS2_RHIRA	0	0	0	0	0	0	0	0	4.33	0	35.20
RAS_LIMLI	0	0	0	0	0	0	0	0	4.15	0	46.03

*: correct answer; SI: sequence identity.

As for MCAS_MOUSE shown in Table 15, KnowPred$_{site}$ predicts its MLCS 100 and it being localized into cytoplasm (CYT) and nuclear (NUC) correctly. Examining the template proteins, we observe that KnowPred$_{site}$ identifies some related proteins, i.e., which have the same localization with the query protein. EF1G1_YEAST and NU155_RAT, even though they share very low sequence identity 8.67% and 3.17%, respectively, with the query protein. Notably, the two template proteins rank second and seventh, respectively, among all template proteins. Furthermore, though GSTA_PLEPL has higher sequence identity (15.86%) with the query protein than EF1G1_YEAST, the confidence score contributed by EF1G1_YEAST is much higher than that by GSTA_PLEPL (2.74 vs. 0.35). It shows that the contributive confidence score is not necessary to be positively correlated with the sequence identity when template sequences are dissimilar with the query sequence. In this example, EF1G1_YEAST shares more local similarities (peptide fragments) with the query protein than GSTA_PLEPL does. If MCA3_HUMAN, the one that shares 88.51% sequence identity with the query protein, is taken out from the template pool, KnowPred$_{site}$ can still predict correctly for protein MCA3_MOUSE.

Table 15 – Prediction result of MCA3_MOUSE. Templates marked with ‘+’ are those that have the same localization annotation with the query protein.

Query	CYT*	CSK	END	EXC	GOL	LYS	MIT	NUC*	PLA	POX	MLCS
MCA3_MOUSE	95.46	0.3	0.27	0.36	0.2	0.01	1.13	93.59	1.82	0.22	100

Template	CYT	CSK	END	EXC	GOL	LYS	MIT	NUC	PLA	POX	SI
MCA3_HUMAN^{+}	89.16	0	0	0	0	0	0	89.16	0	0	88.51
EF1G1_YEAST^{+}	2.74	0	0	0	0	0	0	2.47	0	0	8.67
EF1G2_YEAST	0.49	0	0	0	0	0	0.49	0	0	0	8.50
GSTA_PLEPL	0.35	0	0	0	0	0	0	0	0	0	15.86
SYEC_YEAST	0.16	0	0	0	0	0	0	0	0	0	3.86
CCNA1_MOUSE	0	0.15	0	0	0	0	0	0	0	0	7.36
NU155_RAT^{+}	0.14	0	0	0	0	0	0	0.14	0	0	3.17
GCYB2_HUMAN	0.14	0	0	0	0	0	0	0	0	0	4.86

*: correct answer; SI: sequence identity.

For the multi-localized proteins, there are 318 proteins unable to find similar sequences by the Blast-hit method. However, the localization sites of around half of them can be correctly predicted by KnowPred$_{site}$. We randomly choose an example, CFDP2_BOVIN, to demonstrate the KnowPred$_{site}$’s capability of identifying related sequences from the template pool. The two highest confidence scores of CFDP2_BOVIN are 32.07 (CYT) and 41.18 (NUC). Among the top 100 templates (ranked by the contribution to the confidence scores), 12 of them are localized into CYT and NUC, 18 are localized into

CYT only, and 32 are localized into NUC only. Their sequence identities against CFDP2_BOVIN are very low, ranging from 3.47% to 13.8%. The result suggests that local similarity captured by our method is beneficial for PSL prediction when global sequence similarity is very low.

Another example comes form a user's query. We also implement KnowPred$_{site}$ as a web server to provide prediction service for the public domain. This example also demonstrates the local similarities among proteins with low sequence identities.

Table 16 shows the prediction result of the query protein sent by a user. The query protein, X1005941 should be the protein of the first template since the two share 100% of sequence identity. Therefore, its correct localization site should be the nuclear. In addition to the 100% identical sequence, we also identify more other sequences localized into the same site. However, their sequence identities are very low with the query protein, which range from 7.67% to 15.82%. According to the prediction result, we can still correctly predict the query protein without referring to the first template sequence. It shows that proteins with low sequence similarities actually not only share synonymous words but also move to the same localization site.

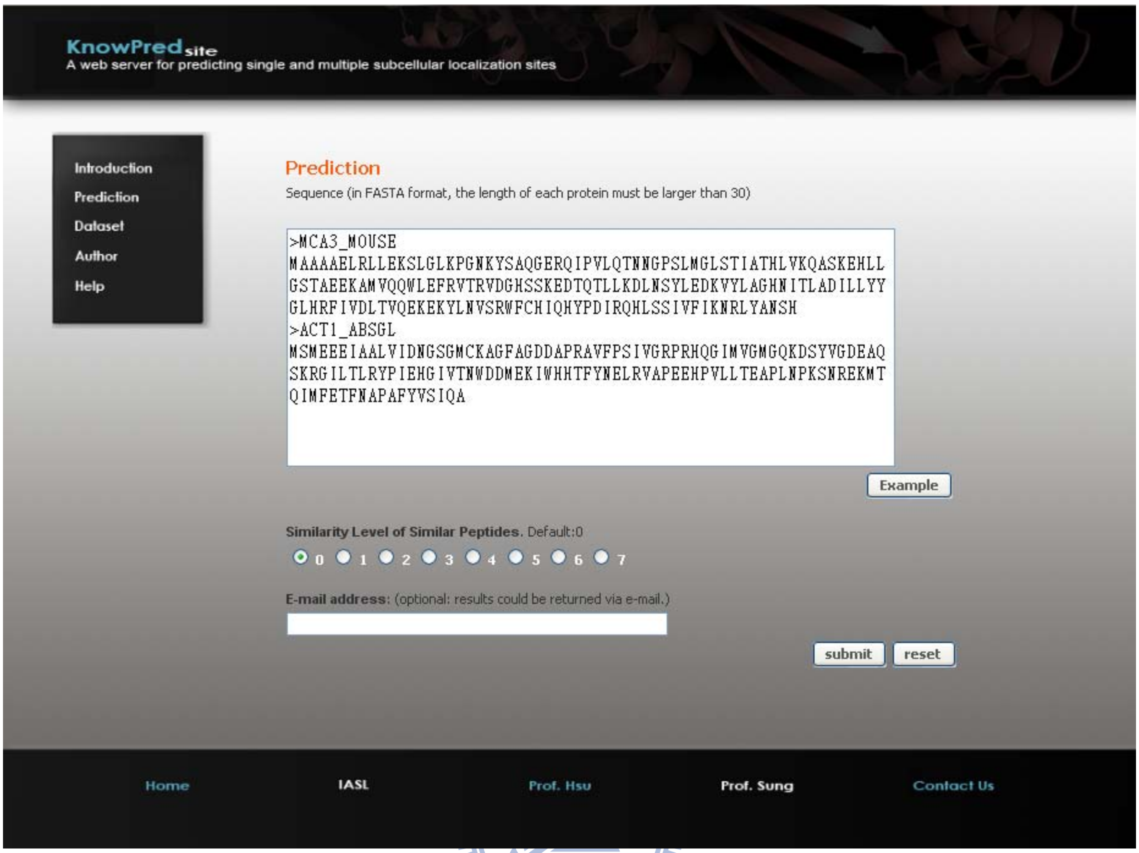
Table 16 – An example from user’s query.

Query	**CYT**	**CSK**	**END**	**EXC**	**GOL**	**LYS**	**MIT**	**NUC**	**PLA**	**POX**	**MLCS**
X1005941	0.83	0.08	0.1	0.32	0.11	0	0.16	98.18	0.5	0.01	2.62

Template	**CYT**	**CSK**	**END**	**EXC**	**GOL**	**LYS**	**MIT**	**NUC**	**PLA**	**POX**	**SI**
PBX1_MOUSE	0	0	0	0	0	0	0	90.25	0	0	100
MEIS1_MOUSE	0	0	0	0	0	0	0	1.15	0	0	12.09
MEIS1_XENLA	0	0	0	0	0	0	0	1.14	0	0	12.79
PKNX2_HUMAN	0	0	0	0	0	0	0	0.87	0	0	15.82
TGIF_HUMAN	0	0	0	0	0	0	0	0.53	0	0	11.34
B3_USTMA	0	0	0	0	0	0	0	0.47	0	0	10.71
TGIF2_HUMAN	0	0	0	0	0	0	0	0.36	0	0	7.67
TGIF_MOUSE	0	0	0	0	0	0	0	0.3	0	0	10.47

4.4 Availability

The KnowPred$_{\text{site}}$ web server as well as the ngLOC dataset is available at http://bio-cluster.iis.sinica.edu.tw/kbloc/. Figure 13 shows a screenshot of KnowPred$_{\text{site}}$ web server. Like SymPred and SymPsiPred web servers, KnowPred$_{\text{site}}$ takes either single sequence or multiple sequences and predict the localization sites of the protein(s). The sequence input should be in fasta format and the sequence length of each of query protein should be longer than 30 in order to have significant sequence alignment when performing a PSI-BLAST search. If an E-mail address is assigned, the prediction result of each query protein will be sent to the user immediately when the prediction is completed. Moreover, users can set the threshold of similarity level freely before the prediction. The prediction result is an html file showing the prediction scores and the template proteins we used. We list template proteins and their sequence identities with the query protein to show how we make the prediction.



Prediction

Sequence (in FASTA format, the length of each protein must be larger than 30)

```
>MCA3_MOUSE
MAAAAELRLLEKSLGLKPGNKYSAQGERQIPVLQTNNGPSLMGLSTIATHLVKQASKEHLL
GSTAEEKAMVQQWLEFRVTRVDGHSSKEDTQTLLKDLNSYLEDKVYLAGHNITLADILLYY
GLHRFIVDLTVQEKEKYLNVSRWFCHIQHYPDIRQHLSSIVFIKNRLYANSH
>ACT1_ABSGL
MSMEEEIAALVIDNGSGMCKAGFAGDDAPRAVFPSIVGRPRHQGIMVGMGQKDSYVGDEAQ
SKRGILTLRYPIEHGIVTNWDDMEKIWHHTFYNELRVAPEEHPVLLTEAPLNPKSNREKMT
QIMFETFNAPAFYVSIQA
```

Similarity Level of Similar Peptides. Default:0

● 0 ○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7

E-mail address: (optional: results could be returned via e-mail.)

Figure 13 – The KnowPred$_{site}$ web server.

4.5 Summaries

In this study, we propose a highly accurate subcellular localization prediction method for single- and multi-localized proteins, called KnowPred$_{\text{site}}$, which is based on a synonymous dictionary instead of frequently used machine learning approaches. The synonymous dictionary, called *SynonymDict*, is compiled from a given dataset of proteins with known localization site annotation to capture local similarity between proteins so that related proteins with the same localization can be identified. Using these related proteins obtained from the synonymous dictionary, the localization site of a query protein can be better predicted.

We used the ngLOC dataset to evaluate the performance of KnowPred$_{\text{site}}$. The dataset consists of 25887 single-localized proteins and 2169 multi-localized proteins of ten subcellular proteomes from 1923 species. In order to compare KnowPred$_{\text{site}}$ with ngLOC and the baseline Blast-hit method, we performed ten-fold cross validation on the dataset. The experiment results show that KnowPred$_{\text{site}}$ achieves higher prediction accuracy than ngLOC and Blast-hit. Particularly, on multi-localized sequences KnowPred$_{\text{site}}$ outperformed ngLOC by 8.2% in accuracy when a protein is correctly predicted if at least one site is correctly identified and by 12.4% in accuracy when a protein is correctly predicted if both sites are correctly identified.

A major advantage of dictionary based approaches is that the prediction process is transparent and explainable. We can examine the prediction process to see how KnowPred$_{\text{site}}$ generates the prediction. Furthermore, with close observation from the prediction results in our experiments, we find that KnowPred$_{\text{site}}$ can efficiently use local similarity to identify related sequences even when their sequence identity is low so as to predict localization site with high accuracy.

When more proteins have known localization sites, most machine learning based methods need to retrain the prediction models, In contrast, KnowPred$_{\text{site}}$ can be easily improved by incrementally expanding the synonymous dictionary, i.e., adding new synonymous word entries or updating existing entries with new protein sources and their localization site information. This feature indicates the expansibility and efficiency in maintaining the KnowPred$_{\text{site}}$ prediction system.

Chapter 5 Remote Homology Detection

5.1 Methods

Since remote homologs share low sequence identity, it is hard to use a traditional homology tool to search a novel protein against a large-scale annotated database to infer their relationship. Therefore, we use a traditional homology tool, e.g., PSI-BLAST, to search a protein against a protein sequence database, e.g., NCBInr, and extract short conserved peptides from high-scoring segment pairs of the protein’s PSI-BLAST results to define its synonymous words that represent the sequence conservation and variation information.

In this study, we propose a two-stage framework to detect remotely homologous proteins. Our proposed framework can be exemplified by the book classification scenario. For example, we have four books at hand, entitled *Introduction to Algorithms*, *Introduction to Psychology*, *The Art of Computer Programming*, and *Interpretation of Dreams*. To group them by relatedness, one may consider using book titles or keywords for similarity comparison. Using titles, the first two books would be grouped together; however, they belong to different disciplines. Using keywords (keywords of these books could be found in Amazon), the first and the third books would be grouped together since they share the following keywords: "Algorithms", "Data structures", and "Languages and Programming". Similarly, though the second book and the fourth book look dissimilar in their titles, but they share keywords of "Psychology", "Health, Mind and Body", "Philosophy & Social Sciences", and "Behavioral Sciences", and thus can be grouped together.

In sequence analysis, we face similar problems to book classification. A protein sequence is like a book. Likewise, when sequence similarity is insufficient to reveal protein homology relationship, we try to define "keywords", later referred to as protein synonymous words, to represent a protein sequence. The critical issue is how to determine corresponding keywords for a protein sequence. Clearly, subsequences as features for a protein are insufficient. We thus consider using available sequence comparison results of a target protein, e.g., PSI-BLAST output, to select similar proteins of the target protein and determine its synonymous words accordingly.

The proposed method, called SymDetector, employs a two-stage mechanism to detect remotely homologous protein sequences. Figure 14 shows the idea of SymDetector. In the figure, we are given 5 protein sequences whose mutual sequence identities are all below 25%. For example, protein *A* and protein *B* share a sequence identity of 17%, and protein *A* and protein *D* share a sequence identity of 22%. Based on their low sequence similarities, it is difficult to distinguish homologous proteins from non-homologous proteins. It is not reliable to determine the homologous relations by setting a sequence similarity threshold among those protein sequences. SymDetector predicts the SCOP classifications of these protein sequences using their synonymous words and a reference of synonymous dictionary. We label those sequences as SCOP classifications and then infer the homologous relations according to the prediction results. For example, SymDetector predicts both protein *A* and protein *B* as type 1, and protein *C* and protein *E* as type 2, and protein *D* as type 3. Therefore, we could divided the five protein sequences into three groups and infer their homologous relations.

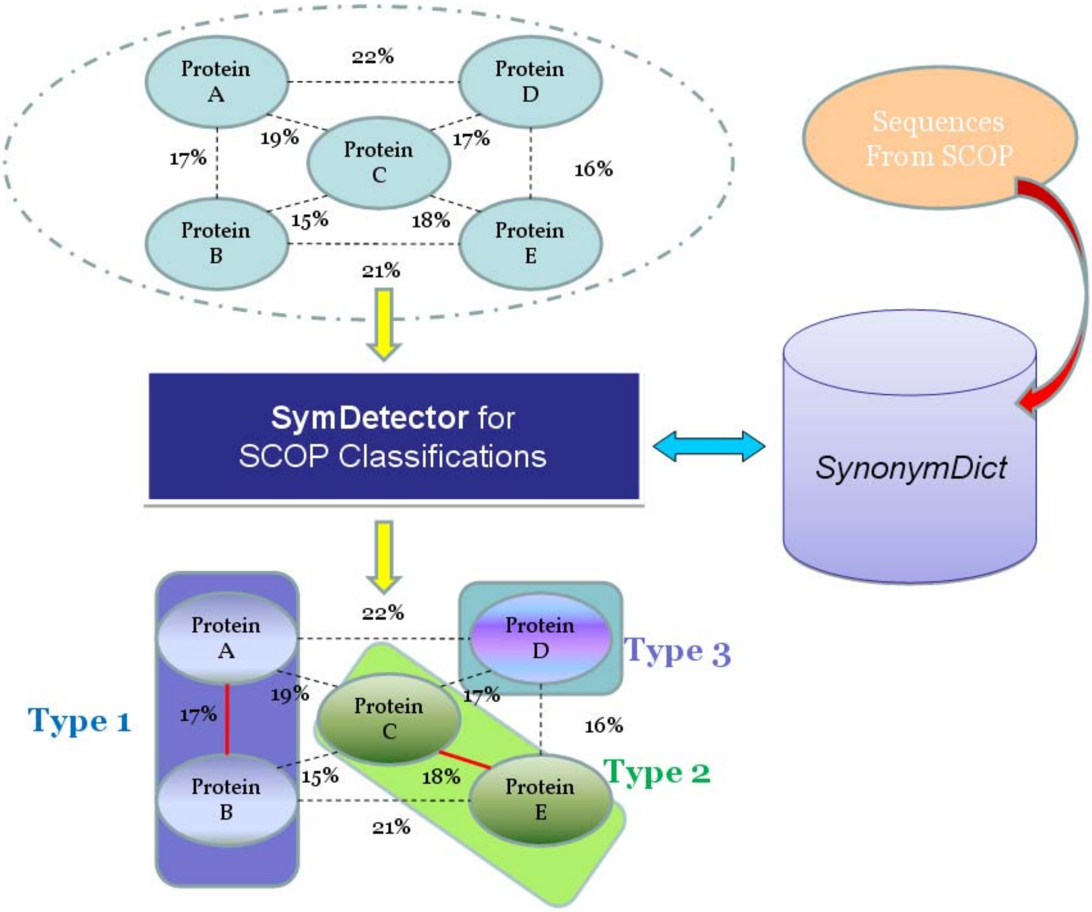


Figure 14 – The main idea of SymDetector.

The First stage of SymDetector: prediction of SCOP classification

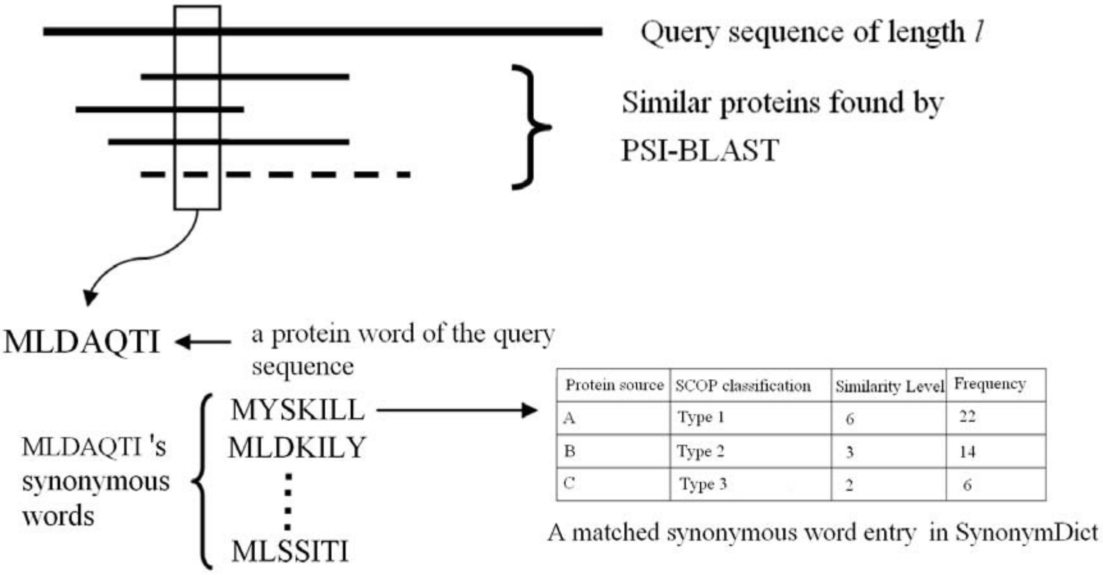
The prediction procedure of SymDetector is shown in Figure 15. Given a query protein *t*, we perform a PSI-BLAST search on *t* to compile a word set containing its original protein words and synonymous words. Like SymPred and KnowPred$_{\text{site}}$, we also calculate the frequency and similarity level of each word in the word set.

- The Scoring Function

The scoring function of SymDetector is like that of SymPred. To define the scoring function, we consider the similarity level and the frequency of the word w in the word set of t, denoted by Sim_t and $freq_t$ respectively, as well as those of a protein source i in its matched entry, denoted by Sim_i and $freq_i$ respectively. Sim_t and $freq_t$ are obtained in the preprocessing stage. To measure the effectiveness of the SCOP classification of the protein source i, we define the voting score s_i as $min(freq_t, freq_i)\times(1+min(Sim_t, Sim_i))$. We choose the minimum value in our formula here to avoid biases derived from those regions of a large amount of HSPs. Although this formula can be refined further, we intend to show that such a simple mechanism already performs well in predicting SCOP classification. The annotation information provided by protein source i will be highly effective if: 1) w is very similar to the corresponding words of t and i; and 2) w is well conserved among the similar proteins of t and i.

Take the synonymous word MYSKILL in Figure 15 as an example. In the figure, MYSKILL is a synonymous word of MLDAQTI which is the original word of the query protein. Assume $freq_t$ and Sim_t of MYSKILL for the query protein are 10 and 2 respectively. We match a synonymous word entry in SynonymDict. The voting score of protein source A is $min(10, 22)\times(1+min(2, 6)) = 10\times(1+2) = 30$. Similarly, the voting score of protein source B is $min(10, 14)\times(1+min(2, 3)) = 10\times(1+2) = 30$, and the voting score of protein source C is $min(10, 6)\times(1+min(2, 2)) = 6\times(1+2) = 18$. In this example, protein sources A and B contribute equal voting scores to the query protein.

The final prediction SCOP classification for the query protein is determined by summing up the voting scores of all the protein sources in the matched entries. The query protein is predicted as the SCOP class with the highest voting score. The score is then used as a confidence score indicating the amount of confidence we make the prediction.


Query sequence of length *l*
Similar proteins found by PSI-BLAST
MLDAQTI
a protein word of the query sequence
MLDAQTI 's synonymous words
MYSKILL
MLDKILY
MLSSITI
A matched synonymous word entry in SynonymDict

Protein source	SCOP classification	Similarity Level	Frequency
A	Type 1	6	22
B	Type 2	3	14
C	Type 3	2	6

Figure 15 – The prediction procedure of SymDetector. An HSP represents a high-scoring segment pair which is a significant sequence alignment reported by PSI-BLAST.

The Second stage of SymDetector: pairing of protein sequences with the same SCOP prediction

- SCOP classification

We use the Structural Classification of Protein (SCOP) database as our standards for determining protein homology relations, and focus on detecting distantly related protein pairs based on their SCOP-Superfamily or SCOP-Fold annotations. SCOP classifies proteins into a four-level hierarchy: Class, Fold, Superfamily, and Family. Currently, the entire protein domains in SCOP are partitioned into 11 Classes, while sequences in each Class would be further classified into different Folds by their secondary structures. According to functions and structural information, homologous sequences in a Fold are further clustered into different Superfamilies, in which highly similar sequences would then be assigned to the same Family.

Remote homology detection targets at any pair of sequences with low sequence identity to determine whether they are homologous. In terms of SCOP classifications, remote homology detection is conventionally referred to determining whether two sequences in the twilight zone (sequence identities between 25% and 40%) or midnight zone (sequence identities below 25%) are from the same Superfamily. Specifically, a sequence pair is regarded as a true positive (TP) of remote homology if they are in the same Superfamily, but not in the same Family, since sequence pairs from the same Family often have sequence identity over 30% and most homology tools can perform well.

In this application, we study not only the conventional remote homology detection but also detection of remote homology with structure similarity, which will be referred to as structurally remote homology detection, in which a pair of sequences share even lower sequence identity than that in the conventional case. A sequence pair is regarded as a true positive of structurally remote homology if the two sequences are in the same Fold, but in different Families.

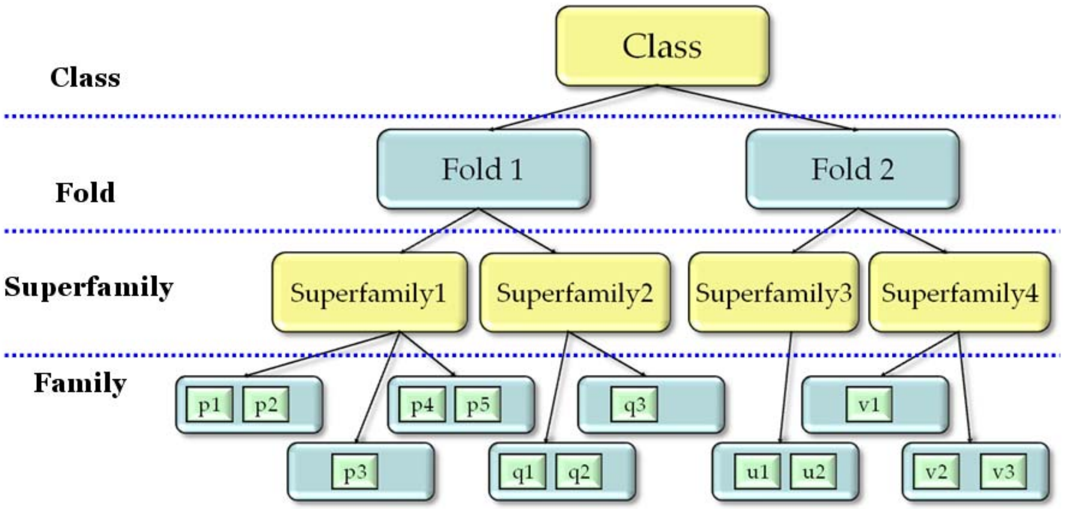


Figure 16 – Remote Homology Detection and SCOP Classifications: The major four-level hierarchy of SCOP classifications.

Figure 16 shows the four-level hierarchy of SCOP classification. In remote homology detection, sequence pairs from the same Superfamilies but different Families are treated as true positives (TPs). For example, the pairs of (p1, p5), (q2, q3), and (v1, v2) are true positives. Those pairs such as (p1, p2) and (p1, q1) would be ignored in this metric. Sequence pairs from different Folds, such as (p1, u1), would then be considered as false positives (FPs). In structurally remote homology detection, the definition of FPs is identical to that in remote homology detection. The definition of TPs is relaxed such that pairs in the same Fold but different Families are counted. The major difference is that, pairs in different Superfamilies, such as (p_i, q_j) and (u_k,v_l), are defined as TPs here, but are ignored in the traditional remote homology detection problem.

- Pairing

In the first stage of SymDetector, we predict each protein sequence a SCOP classification as well as a voting score indicating the reliability of our prediction. In the second stage, we pair two protein sequences with the same SCOP classification as a putative true positive and assign a confidence score showing the reliability of begin a homologous protein pair of the two sequences. The confidence score is given by the smaller of voting scores of the two proteins. For example, if protein *A* is predicted as SCOP Superfamily of Globin-like with the voting score of 5820, and protein *B* is predicted as the same SCOP Superfamily with the voting score of 4175, then the confidence score of pairing protein *A* and *B* as a homologous pair is *min*(5820, 4175) = 4175.

5.2 Results

5.2.1 Datasets and evaluations

Remote homology detection methods are often evaluated by qualities of detected sequence pairs from a set of non-redundant SCOP sequences. We adopt the dataset of 2,476 non-redundant SCOP sequences used in Przybylski and Rost's ConSequenceS [119] (https://rostlab.org/owiki/index.php/ConSequenceS) as a benchmark dataset, which is called the *PR* dataset. In short, they selected sequences from SCOP 1.69 (The latest version is SCOP 1.75) such that, while searching against UniProt, none of sequence pairs could be aligned by BLAST with e-value better than 0.001.

Performances of our approach would benefit from the synonymous dictionary constructed based on a reference SCOP set. To obtain the reference dataset, we use the PSI-CD-HIT to select sequences from SCOP such that the selected sequences would share no more than 25% of sequence identities to each sequence in the *PR* dataset. The resultant reference set consists of 8,442 SCOP sequences sharing low identities to any of the 2,476 benchmark sequences.

Among millions of all possible sequence pairs generated from the *PR* dataset, 52,620 and 18,780 order pairs of sequences belong to identical Folds and Superfamilies, respectively. According to ClustalW, these two sets of sequence pairs have average sequence identities of 11.63% (pairs in identical Folds) and 12.02% (pairs in identical Superfamilies), while average identities of all possible pairs being 9.70%. The average sequence identity about the benchmark dataset will be discussed more detailed in section 5.3.3. By ordered pairs, we indicate that for sequences *A* and *B* in benchmark set, pairs (*A*, *B*) and (*B*, *A*) would be treated as different cases in evaluations. The notion of ordered pairs reflects that, in most homology search tools, relatedness between *A* and *B* would be assigned with different significances due to different query sequences. While detection results of our framework are symmetric, in which scores of pairs (*A*, *B*) and (*B*, *A*) are both equal to the minimum of their voting scores of the predicted SCOP classifications, we still provide evaluations based on ordered pairs for convenient comparisons.

To evaluate the performance of SymDetector, we count the cumulative number of true positive pairs given a number of cumulative false positive pairs. This evaluation serves as the standard measurement of remote homology detection. Protein sequence pairs are sorted by their confidence scores and regarded into true positive pairs and false positive pairs by the real SCOP classification. Two proteins in a pair classified into the same Superfamily or Fold but not the same Family are regarded as a true positive pair. On the contrary, two proteins in a pair classified into different Folds are regarded as a false positive pair.

5.2.2 Experiment result on Remote Homology Detection

Figure 17 shows the experiment results of SymDetector on remote homology detection. We evaluate the performance of SymDetector using Superfamily prediction and Fold prediction respectively in the first stage. We can see that before the first false positive pair appears, SymDetector can identify 5,294 true positive pairs and 186 true positive pairs respectively, and before the 100th false positive pair appears, SymDetector can identify

6,892 and 4,368 true positive pairs respectively. The ROC curves in Figure 17 become stable when the cumulative numbers of false positives are larger than 300. It shows that most true positive pairs identified by SymDetector have higher confidence scores than false positive pairs. Therefore our confidence scores are good indicators showing the reliability of being homologous protein pairs.

In this experiment, we find that the performance of SymDetector with Superfamily prediction is better than that with Fold prediction since in this problem we define a true positive pair consisting of two proteins with the same Superfamily. Therefore, SymDetector perform better with Superfamily prediction than with Fold prediction in the first stage of our method.

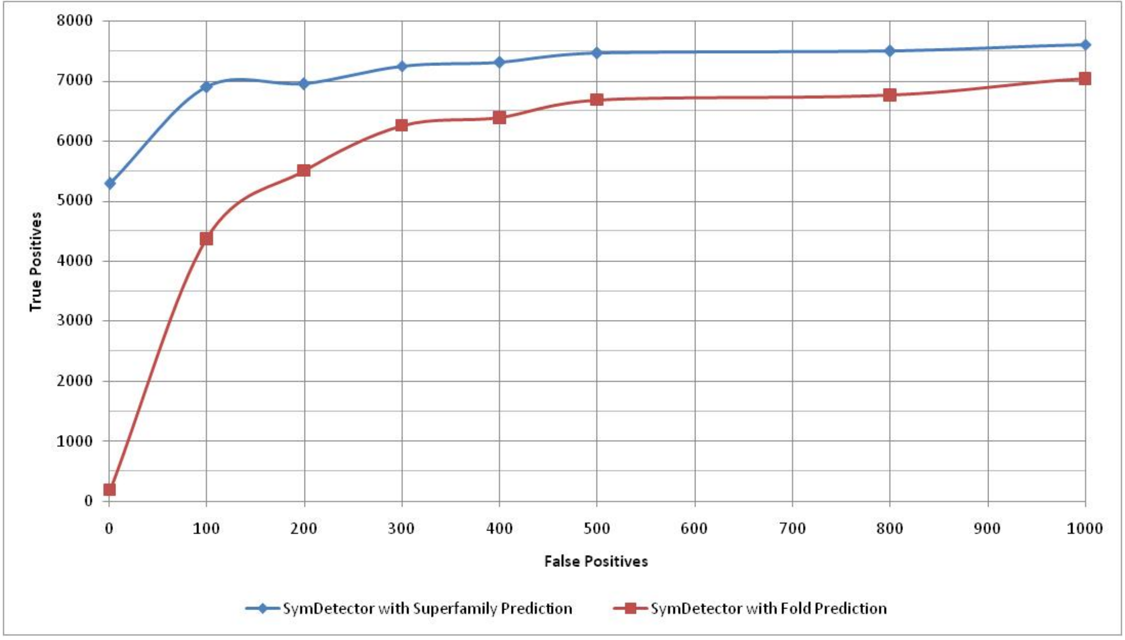


Figure 17 – Performances of our framework on remote homology detection.

5.2.3 Experiment result on Structurally Remote Homology Detection

Figure 18 shows the experiment results of SymDetector on structurally remote homology detection. In this problem, we also evaluate the performance of SymDetector using Superfamily prediction and Fold prediction respectively in the first stage and compare with ConSequenceS and PSI-BLAST.

We can see that before the first false positive pair appears, SymDetector can identify 5,308 true positive pairs and 772 true positive pairs respectively, and before the 100th false positive pair appears, SymDetector can identify 6,906 and 12,805 true positive pairs respectively. It can be observed that SymDetector could identify more true positive pairs given a specific number of false positive pairs than ConSequenceS and PSI-BLAST. For example, ConSequenceS identified around 2,100 true positive pairs before the 100th false positive pair appears and PSI-BLAST identified around 1,400 true positive pairs at the same cutoff.

Both ConSequenceS and PSI-BLAST to identify remote homology sequences are mainly based on sequence similarities (sequence alignments). However, it is rather difficult to distinguish homologous protein sequences from non-homologous protein sequences when the sequences are in the midnight zone. Therefore, SymDetector identifies homologous proteins by transforming protein sequences into SCOP classifications. We avoid direct sequence comparison and transform the sequences into other annotations to find some relations with other sequences. We show that our method is more efficient than sequence alignment based approaches. Therefore, given a query protein sequence, SymDetector could find all possible related sequences by predicting its SCOP classification no matter how similar or dissimilar those protein sequences are.

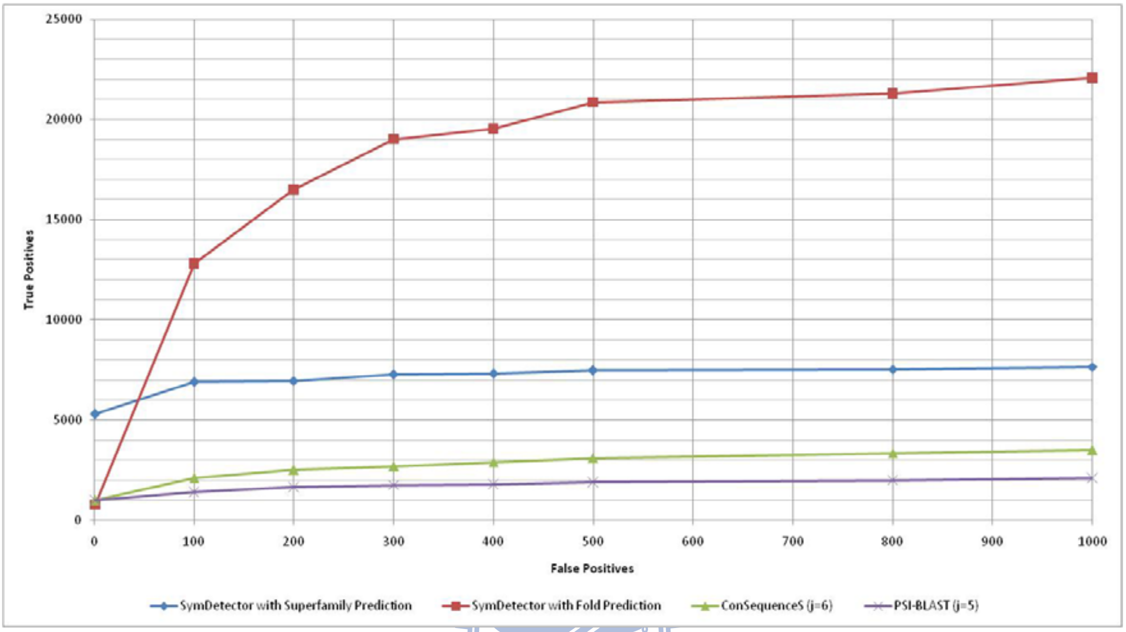


Figure 18 – Performances of SymDetector on structurally remote homology detection and Comparison with ConSequenceS and PSI-BLAST.

5.2.4 Prediction performance of SymDetector on *PR* dataset

Below we provide the basic statistics about SCOP annotations of 2,476 sequences in the benchmark dataset. Statistics of 8,442 sequences in the reference dataset which are used to compile the SynonymDict would also be shown. There are 607 Folds and 969 Superfamilies in the benchmark dataset, while reference dataset contains 975 Folds and 1,609 Superfamilies. Among these annotations, the two sets share 500 Folds and 763 Superfamilies. It implies that not all sequences in *PR* dataset have sequence templates with the same Fold or Superfamily annotations in the reference dataset. Therefore, our prediction performance is limited to the number of sequences with the same annotations.

We measure the prediction accuracy based on sequence level. In other words, we evaluate the number of sequences that share their Folds or Superfamilies with at least one of 8,442 reference sequences. There are 2,352 sequences and 2,234 sequences respectively permitting the constraint above. Therefore, these ratios could be treated as the theoretical upper bounds for annotation prediction accuracy for the benchmark dataset. Since SymDetector only assigns query sequences annotations from SynonymDict, the annotation assignment accuracy should be therefore adjusted accordingly. After all, for the remaining 124 (or 242) sequences whose Fold (or Superfamily) annotations are not in SynonymDict, it would be impossible for SymDetector to assign them with correct annotations.

Table 17 shows the prediction accuracies of SymDetector. It can be observed that there are 2,352 protein sequences in the *PR* dataset which share the same Fold with protein sequences in the reference dataset. Therefore, the theoretical upper bound of prediction accuracy is about 95.0%. Among those protein sequences, 1,759 proteins are correctly predicted, therefore, the prediction accuracy of SymDetector for Fold classification is about 74.8%. Likewise, there are 2,234 protein sequences in the *PR* dataset which share the same Superfamily with proteins in the reference dataset. The theoretical upper bound is 90.2% and the prediction accuracy for Superfamily classification is about 78.0%.

Table 17 – The prediction accuracy of SymDetector.

Evaluation Types	Number of proteins in *PR* dataset (*A*)	Upper Bounds for Prediction Accuracy	Number of proteins with correct predictions (*B*)	Prediction Accuracy (*B*/*2476*)	Adjusted Prediction Accuracy (*B*/*A*)
Sequences belong to 500 shared Fold	2352	95.0%	1759	71.0%	74.8%
Sequences belong to 763 shared Superfamily	2234	90.2%	1742	70.4%	78.0%

5.3 Discussions

5.3.1 Sequence Classification: Different Annotations Capture Different Relations

The efficacy of SymDetector relies on the integrating information from SynonymDict to infer relations among query proteins. Because SymDetector is adaptive to different types of sequence annotations, the sequence relations would be affected by different sequence annotations. Although we use the identical SynonymDict to analyze the benchmark dataset, detection results based on Superfamily classification and Fold classification are different.

In Figure 19, we adopt two different evaluations to assess the detection results only based on Superfamily classification. It shows that, even though the evaluation for structurally remote homology allows sequence pairs in the same Fold to be true positives, the detection result does not benefit to capture such sequence pairs when we perform Superfamily prediction in the first stage. On the other hand, most of reported pairs based on Fold classification belong to those sequence pairs in the same Fold but different Superfamilies. Therefore, the detection result based on Fold classification could achieve a remarkable improvement under structurally remote homology detection evaluation.

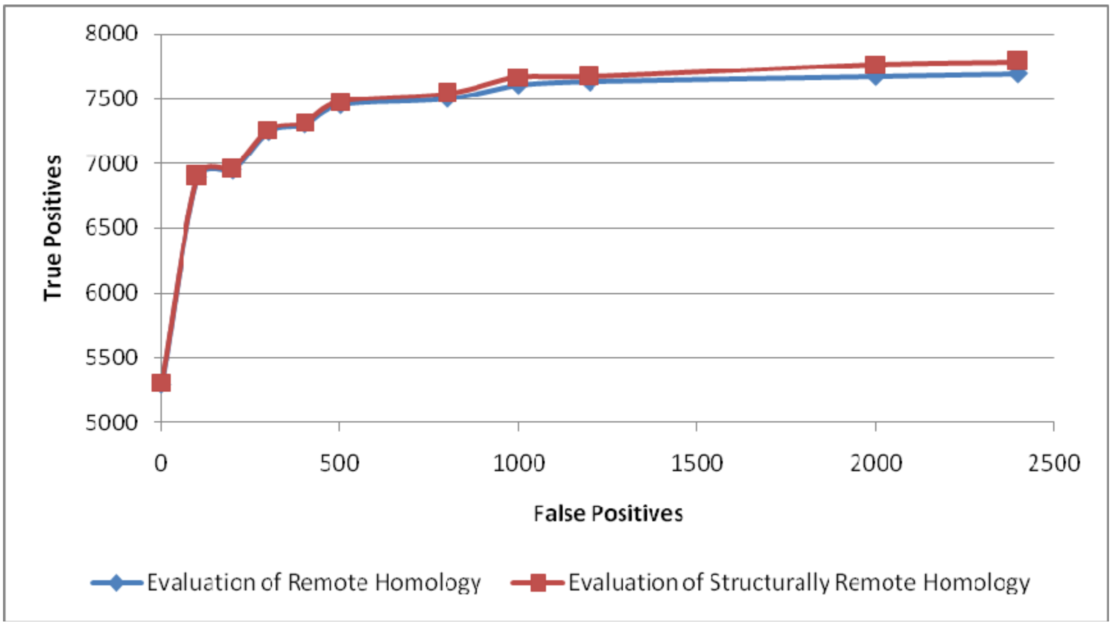


Figure 19 – Performance of Classification by Superfamily under two metrics: We evaluate the same ranked list by two different metrics: remote homology detection and structural remote homology detection. The performances are similar, and indicate that such classification strategy mainly capture sequence relations in the same superfamily.

5.3.2 Remote homology detection in the real world

In the previous experiment results, we infer the homologous relations among proteins in the benchmark dataset. That is, we focus on the identification of homologous relations among a group of unknown proteins. However, in the real world we are often given an unknown protein and asked to identify other proteins of known annotations that are homologous to the query protein. By referring to those protein sequences, we could transfer the structure or function of the query sequence. Therefore, we here analyze the detection performance of SymDetector under this situation.

Given an unknown protein sequence, SymDetector will predict its Superfamily classification and identify protein sequences which have been annotated with the same Superfamily classification. For example, given a query sequence *A*, if its Superfamily prediction is *S1* with a voting score of 3,500, then we pair protein *A* and all protein sequences, say protein *B*, *C*, and *D*, of real Superfamily *S1* in the benchmark dataset. In this example, we can have the pairs of (*A*, *B*), (*A*, *C*), and (*A*, *D*) all with the confidence score 3,500.

Figure 20 shows results of such evaluations for remote homology detection. We first predict a sequence to some specific Superfamily or Fold classification, and examine the relations between this sequence and all protein sequences truly of this classification. Given 1, 100, and 1000 false positives, the result based on Superfamily prediction can report 9083, 9867, and 10168 homologous pairs. On the other hand, the result based on Fold prediction only reports 9095, 9450, and 9856 homologous pairs.

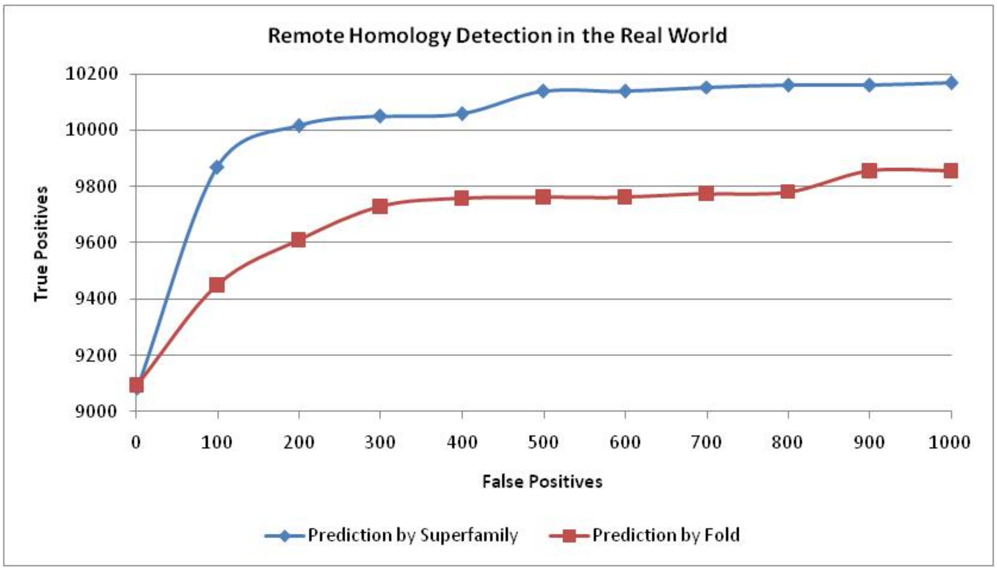


Figure 20 – The experiment result of remote homology detection in the real world.

On structurally remote homology detection, we apply the same rules to evaluate the performance. The difference is that, pairs in the same Fold but different Family are now considered as true positives. Figure 21 shows that, once we classify query sequence based on Fold, reliability of structurally homology detection based on Fold prediction would be higher than that based on Superfamily prediction.

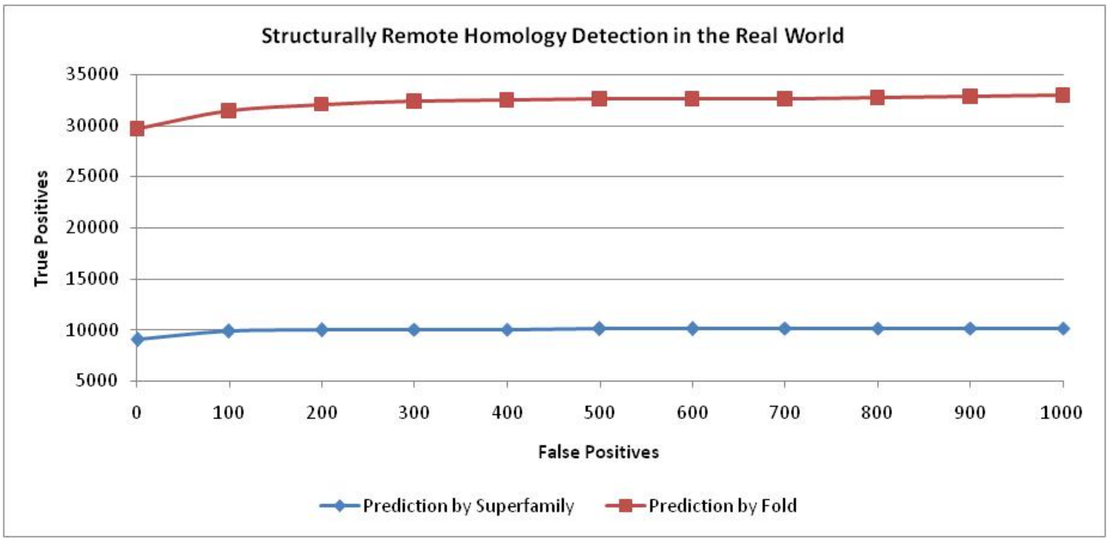


Figure 21 – The experiment result of structurally remote homology detection in the real world.

5.3.3 SymDetector Assists to Overcome Difficulties Due to Low Sequence Identities

SymDetector identifies homologous protein pairs with confidence scores showing the reliability of the identifications. In this subsection, we study the relationship between sequence identities and confidence scores of correctly identified homologous protein pairs. For 2,476 sequences in the benchmark dataset, we consider all 9,218 correctly detected homologous pairs based on Superfamily classifications. We calculate their sequence identities using ClustalW, and get the following regression line (in Figure 22) between the sequence identities and the confidence scores reported by SymDetector. The correlation coefficient between the two is -0.017. Apparently, the confidence scores in SymDetector are irrelevant to the sequence identities. The behavior of regression line is similar for all 31,670 detected structurally remote homologous pairs (in Figure 23). The correlation coefficient in this case is 0.002. It implies that SymDetector could identify remotely homologous protein pairs without considering their sequence identities.

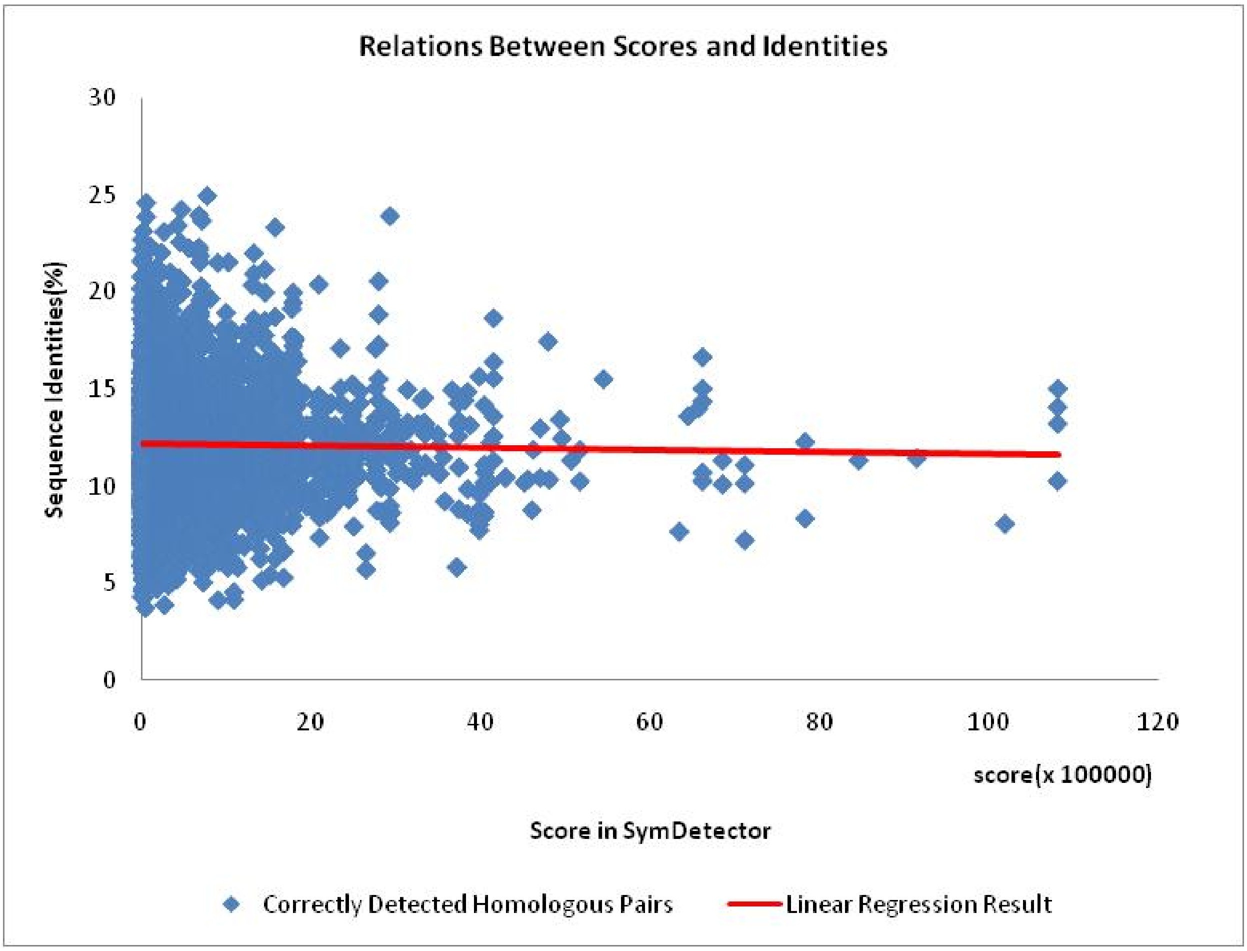


Figure 22 –The relationship between sequence identities and confidence scores reported by SymDetector for the problem of remote homology detection.

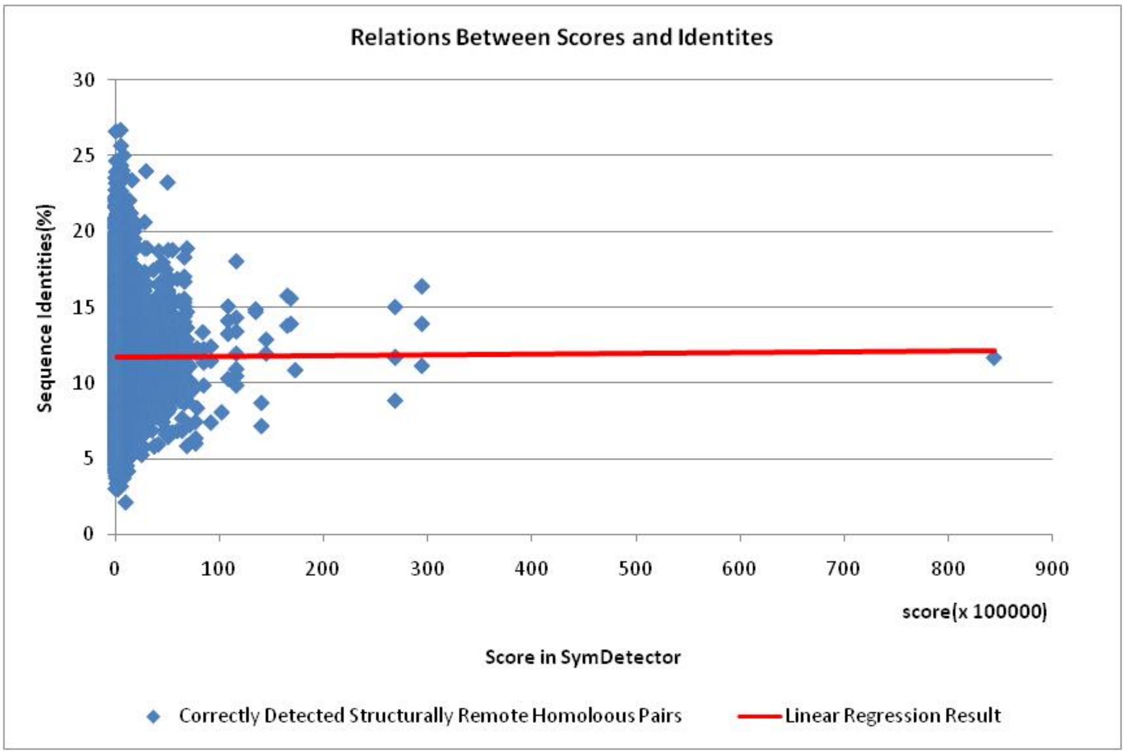


Figure 23 – The relationship between sequence identities and confidence scores reported by SymDetector for the problem of structurally remote homology detection.

In Table 18 we shows the average sequence identities between sequences in different categories. Among all 3,064,050 possible pairs generated from 2,476 sequences, the average sequence identity is about 9.70%. For sequences in the same Fold, the Superfamily, and same Family, their average identities are 11.63%, 12.02%, and 14.68%, respectively. All the average seqeunce identities in different catories are much lower than 25%, which shows the benchmark dataset is a very challenging one for remote homology detection. The identification of homologous protein pairs based on sequence alignment approaches is very difficult by only thresholding a single cut-off value of sequence identity. Therefore SymDetector adopts the two-stage framework to identify the homologous relations between proteins in the midnight zone.

Table 18 – The average sequence identities of protein sequences in different categories.

Category	Type	Number of Sequence Pairs	Average Identities
All Seuqence Pairs		3064050	9.70%
Structurally Remote Homology Detection	True Positives	24035	11.63%
	True negatives	3037693	9.68%
Remote Homology Detection	True Positives	7066	12.02%
	True negatives	3037693	9.68%
Sequences in the same Family	True Positives	2322	14.68%
	True negatives	3061798	9.69%

5.4 Summaries

Based on the concepts of the synonymous words described above, we extend it to design a two-stage framework for analyzing homology-based inference problems, especially for those in twilight zone and midnight zone. We achieve this goal by using synonymous words as intermediates so that information from other annotated sequences could be applied to boost detections of relatedness on the unknown sequence set. Conceptually, the analysis framework contains three steps: 1) the construction of synonymous dictionary from a set of reference sequences; 2) the extraction of synonymous words from query sequences; 3) and relation detections by SCOP classification based on the synonymous dictionary.

Since the first stage of SymDetector is independent of any type of annotations, this framework allows for great flexibility to solving different kinds of problems. The integration of synonymous words and information from dictionary provides a different point of view for evaluating relatedness between sequences. As a result, while the pairwise similarities between homologous and non-homologous sequences are of the same level, our framework can boost detection results from PSI-BLAST search results. Moreover, based on the design of this framework, it can be easily to be applied for improving results from other search and alignment tools, such as CSI-BLAST, HHSearch, COMPASS, and so on.

Chapter 6 Concluding remarks and outlook

The N-gram models (protein words) have been used in protein sequence analysis since 1970s. BLAST extended the idea of N-gram models and devised similar words for identifying more similar proteins while performing sequence searches. BLAST used similar words to recover the sensitivity lost by only matching identical words. However, the generation of similar words is from a substitution matrix and there is no guarantee of structure similarity between similar words. Based on the observation that protein structures are more conserved than protein sequences, we treat two protein sequences which form a significant alignment as two paragraphs which have similar meanings in terms of structure. We define synonymous relations between two words that are aligned together in a significant sequence alignment.

In this study, we proposed synonymous words as protein sequence features to study some problems in Bioinformatics. We devised a synonymous dictionary based approach to study those problems. We demonstrated that our approach could deal with protein secondary structure prediction, protein subcellular localization prediction, remote homology detection, and protein sequence alignments.

Using a set of protein sequences with structural or functional annotations, we performed PSI-BLAST searches and used the reported sequence alignments to extract synonymous words and then compiled a synonymous dictionary. By looking up the dictionary, we treated protein prediction or classification problems as translation problems. According to the experiment results, we show that synonymous words would tend to express similar structures or have similar functions. In the application of protein secondary structure prediction, we show that SymPred achieves around 81% of Q_3 accuracy and outperforms existing PSS predictors. In the application of protein subcellular localization prediction, we show that KnowPred$_{\text{site}}$ can predict both single-localized and multi-localized proteins at high accuracy. We demonstrated that KnowPred$_{\text{site}}$ could identify related protein sequences (with the same localization sites) using synonymous words. In the application of remote homology detection, we suggest that a two-stage mechanism seems more efficient than traditional sequence comparison methods. And in the application of protein sequence alignment, we demonstrated that synonymous words could be used to measure the alignment scores between amino acid pairs.

From the experiment results of four different applications, we find that synonymous words could represent the local sequence similarities among protein sequences and they tended to express similar structures and functions. We find that even if the sequence identity between two homologous (related) proteins is low, they might share a number of synonymous words. Moreover, we also show that our synonymous dictionary based approach is sensitive to the size of template pool and the number of sequence variations in protein evolution. With the increasing number of protein sequences and structures, our method could improve further in the future.

References

1. Fischer, D., et al., *CAFASP2: The second critical assessment of fully automated structure prediction methods.* Proteins-Structure Function and Genetics, 2001: p. 171-183.
2. Gong, H.P. and G.D. Rose, *Does secondary structure determine tertiary structure in proteins?* Proteins-Structure Function and Bioinformatics, 2005. **61**(2): p. 338-343.
3. Meiler, J. and D. Baker, *Coupled prediction of protein secondary and tertiary structure.* Proceedings of the National Academy of Sciences of the United States of America, 2003. **100**(21): p. 12105-12110.
4. Rost, B., *Review: Protein secondary structure prediction continues to rise.* Journal of Structural Biology, 2001. **134**(2-3): p. 204-218.
5. Aydin, Z., Y. Altunbasak, and M. Borodovsky, *Protein secondary structure prediction for a single-sequence using hidden semi-Markov models.* Bmc Bioinformatics, 2006. **7**: p. -.
6. Eisner, R., et al. *Improving Protein Function Prediction using the Hierarchical Structure of the Gene Ontology*. in *Computational Intelligence in Bioinformatics and Computational Biology, 2005. CIBCB '05. Proceedings of the 2005 IEEE Symposium on*. 2005.
7. Ferre, S. and R.D. King, *Finding motifs in protein secondary structure for use in function prediction.* Journal of Computational Biology, 2006. **13**(3): p. 719-731.
8. Lisewski, A.M. and O. Lichtarge, *Rapid detection of similarity in protein structure and function through contact metric distances.* Nucleic Acids Research, 2006. **34**(22): p. -.
9. Nair, R. and B. Rost, *Mimicking cellular sorting improves prediction of subcellular localization.* Journal of Molecular Biology, 2005. **348**(1): p. 85-100.
10. Lobley, A., et al., *Inferring function using patterns of native disorder in proteins.* Plos Computational Biology, 2007. **3**(8): p. 1567-1579.
11. Przytycka, T., R. Aurora, and G.D. Rose, *A protein taxonomy based on secondary structure.* Nature Structural Biology, 1999. **6**(7): p. 672-682.
12. Bondugula, R. and D. Xu, *MUPRED: A tool for bridging the gap between template based methods and sequence profile based methods for protein secondary structure prediction.* Proteins-Structure Function and Bioinformatics, 2007. **66**(3): p. 664-670.
13. Ceroni, A., et al., *A combination of support vector machines and bidirectional recurrent neural networks for protein secondary structure prediction.* Ai(Asterisk)Ia 2003: Advances in Artificial Intelligence, Proceedings, 2003. **2829**: p. 142-153.
14. Cheng, H.T., et al., *Prediction of protein secondary structure by mining structural fragment database.* Polymer, 2005. **46**(12): p. 4314-4321.

15. Jones, D.T., *Protein secondary structure prediction based on position-specific scoring matrices.* Journal of Molecular Biology, 1999. **292**(2): p. 195-202.

16. Karplus, K., C. Barrett, and R. Hughey, *Hidden Markov models for detecting remote protein homologies.* Bioinformatics, 1998. **14**(10): p. 846-856.

17. Kim, H. and H. Park, *Protein secondary structure prediction based on an improved support vector machines approach.* Protein Engineering, 2003. **16**(8): p. 553-560.

18. Rost, B. and C. Sander, *Third generation prediction of secondary structure*, in *Protein Structure Prediction: Methods and Protocols*. 2000, Humana Press. p. 71-95.

19. Ward, J.J., et al., *Secondary structure prediction with support vector machines.* Bioinformatics, 2003. **19**(13): p. 1650-1655.

20. Rost, B., C. Sander, and R. Schneider, *Redefining the goals of protein secondary structure prediction.* J Mol Biol, 1994. **235**(1): p. 13-26.

21. Zemla, A., et al., *A modified definition of Sov, a segment-based measure for protein secondary structure prediction assessment.* Proteins-Structure Function and Genetics, 1999. **34**(2): p. 220-223.

22. Rost, B., *Rising accuracy of protein secondary structure prediction*, in *Protein Structure Determination, Analysis, and Modeling for Drug Discovery*, D.I. Chasman., Editor. 2003, Marcel Dekker: New York. p. 207–249.

23. Przybylski, D. and B. Rost, *Alignments grow, secondary structure prediction improves.* Proteins-Structure Function and Genetics, 2002. **46**(2): p. 197-205.

24. Pollastri, G. and A. McLysaght, *Porter: a new, accurate server for protein secondary structure prediction.* Bioinformatics, 2005. **21**(8): p. 1719-1720.

25. Dor, O. and Y.Q. Zhou, *Achieving 80% ten-fold cross-validated accuracy for secondary structure prediction by large-scale training.* Proteins-Structure Function and Bioinformatics, 2007. **66**(4): p. 838-845.

26. Salamov, A.A. and V.V. Solovyev, *Prediction of Protein Secondary Structure by Combining Nearest-neighbor Algorithms and Multiple Sequence Alignments.* Journal of Molecular Biology, 1995. **247**(1): p. 11-15.

27. Salamov, A.A. and V.V. Solovyev, *Protein secondary structure prediction using local alignments.* Journal of Molecular Biology, 1997. **268**(1): p. 31-36.

28. Dmitrij Frishman, P.A., *Seventy-five percent accuracy in protein secondary structure prediction.* Proteins: Structure, Function, and Genetics, 1997. **27**(3): p. 329-335.

29. Wu, K.P., et al., *HYPROSP: a hybrid protein secondary structure prediction algorithm--a knowledge-based approach.* Nucleic Acids Res, 2004. **32**(17): p. 5059-65.

30. Kursun, O., et al., *ANSWER: Approximate name search with errors in large databases by a novel approach based on prefix-dictionary.* International Journal on Artificial Intelligence Tools, 2006. **15**(5): p. 839-848.

31. Kursun, O., et al., *A dictionary-based approach to fast and accurate name matching in large law enforcement databases.* Intelligence and Security Informatics, Proceedings, 2006. **3975**: p. 72-82.

32. Egorov, S.R., A. Yuryev, and N. Daraselia, *A simple and practical dictionary-based approach for identification of proteins in medline abstracts.* Journal of the American Medical Informatics Association, 2004. **11**(3): p. 174-178.

33. Nair, R. and B. Rost, *Better prediction of sub-cellular localization by combining evolutionary and structural information.* Proteins-Structure Function and Genetics, 2003. **53**(4): p. 917-930.

34. Gardy, J.L., et al., *PSORTb v.2.0: expanded prediction of bacterial protein subcellular localization and insights gained from comparative proteome analysis.* Bioinformatics, 2005. **21**(5): p. 617-23.

35. Chang, J.M., et al., *PSLDoc: Protein subcellular localization prediction based on gapped-dipeptides and probabilistic latent semantic analysis.* Proteins, 2008. **72**(2): p. 693-710.

36. Hoglund, A., et al., *MultiLoc: prediction of protein subcellular localization using N-terminal targeting sequences, sequence motifs and amino acid composition.* Bioinformatics, 2006. **22**(10): p. 1158-65.

37. Wang, J.R., et al., *Protein subcellular localization prediction for Gram-negative bacteria using amino acid subalphabets and a combination of multiple support vector machines.* Bmc Bioinformatics, 2005. **6**: p. -.

38. Yu, C.S., et al., *Prediction of protein subcellular localization.* Proteins, 2006. **64**(3): p. 643-51.

39. Yu, C.S., C.J. Lin, and J.K. Hwang, *Predicting subcellular localization of proteins for Gram-negative bacteria by support vector machines based on n-peptide compositions.* Protein Sci, 2004. **13**(5): p. 1402-6.

40. Bhasin, M., A. Garg, and G.P. Raghava, *PSLpred: prediction of subcellular localization of bacterial proteins.* Bioinformatics, 2005. **21**(10): p. 2522-4.

41. Chou, K.C. and Y.D. Cai, *Predicting protein localization in budding yeast.* Bioinformatics, 2005. **21**(7): p. 944-50.

42. Gardy, J.L., et al., *PSORT-B: Improving protein subcellular localization prediction for Gram-negative bacteria.* Nucleic Acids Res, 2003. **31**(13): p. 3613-7.

43. Lee, K., et al., *PLPD: reliable protein localization prediction from imbalanced and overlapped datasets.* Nucleic Acids Res, 2006. **34**(17): p. 4655-66.

44. Huang, W.L., et al., *ProLoc-GO: utilizing informative Gene Ontology terms for sequence-based prediction of protein subcellular localization.* BMC Bioinformatics, 2008. **9**: p. 80.

45. Marcotte, E.M., et al., *Localizing proteins in the cell from their phylogenetic profiles.* Proc Natl Acad Sci U S A, 2000. **97**(22): p. 12115-20.

46. Mott, R., et al., *Predicting protein cellular localization using a domain projection method.* Genome Res, 2002. **12**(8): p. 1168-74.

47. Su, E.C., et al., *Protein subcellular localization prediction based on compartment-specific features and structure conservation.* BMC Bioinformatics, 2007. **8**: p. 330.

48. Rychlewski, L., et al., *Comparison of sequence profiles. Strategies for structural predictions using sequence information.* Protein Science, 2000. **9**(2): p. 232-241.

49. Sadreyev, R. and N. Grishin, *COMPASS: A tool for comparison of multiple protein alignments with assessment of statistical significance.* Journal of Molecular Biology, 2003. **326**(1): p. 317-336.

50. Przybylski, D. and B. Rost, *Consensus sequences improve PSI-BLAST through mimicking profile-profile alignments.* Nucleic Acids Research, 2007. **35**(7): p. 2238-2246.

51. Pietrokovski, S., *Searching databases of conserved sequence regions by aligning protein multiple-alignments.* Nucleic Acids Research, 1996. **24**(19): p. 3836-3845.

52. Yona, G. and M. Levitt, *Within the twilight zone: A sensitive profile-profile comparison tool based on information theory.* Journal of Molecular Biology, 2002. **315**(5): p. 1257-1275.

53. Zhang, S., et al., *DBMLoc: a Database of proteins with multiple subcellular localizations.* BMC Bioinformatics, 2008. **9**: p. 127.

54. King, B.R. and C. Guda, *ngLOC: an n-gram-based Bayesian method for estimating the subcellular proteomes of eukaryotes.* Genome Biology, 2007. **8**(5): p. -.

55. Lin, H.N., et al., *HYPROSP II--a knowledge-based hybrid method for protein secondary structure prediction based on local prediction confidence.* Bioinformatics, 2005. **21**(15): p. 3227-33.

56. Chen, C.T., et al., *HYPLOSP: a knowledge-based approach to protein local structure prediction.* J Bioinform Comput Biol, 2006. **4**(6): p. 1287-307.

57. Koski, L.B.a.G., G. B., *The closest BLAST hit is often not the nearest neighbor.* Journal of Molecular Evolution, 2001. **52**(6): p. 3.

58. Rost, B., *Twilight zone of protein sequence alignments.* Protein Eng, 1999. **12**(2): p. 85-94.

59. Fariselli, P., et al., *The WWWH of remote homolog detection: the state of the art.* Brief Bioinform, 2007. **8**(2): p. 78-87.

60. Wan, X.F. and D. Xu, *Computational methods for remote homolog identification.* Curr Protein Pept Sci, 2005. **6**(6): p. 527-46.

61. Stormo, G.D.a.S., Thomas D. and Gold, Larry and Ehrenfeucht, Andrzej, *Use of the 'Perceptron' algorithm to distinguish translational initiation sites in E. coli.* Nucleic Acids Research, 1982. **10**(9): p. 15.

62. Gribskov, M.a.M., A. D. and Eisenberg, D., *Profile analysis: detection of distantly related proteins.* Proceedings of the National Academy of Sciences of the United States of America, 1987. **84**(13): p. 4.

63. Baldi, P., et al., *Hidden Markov models of biological primary sequence information.* Proc Natl Acad Sci U S A, 1994. **91**(3): p. 1059-63.

64. Henikoff, S. and J.G. Henikoff, *Amino acid substitution matrices from protein blocks.* Proc Natl Acad Sci U S A, 1992. **89**(22): p. 10915-9.

65. Krogh, A.a.B., M. and Mian, I. S. and Sj\"{o}lander, K. and Haussler, D, *Hidden Markov models in computational biology. Applications to protein modeling.* Journal of Molecular Biology, 1994. **235**(5): p. 31.

66. Johnson, L.S., S.R. Eddy, and E. Portugaly, *Hidden Markov model speed heuristic and iterative HMM search procedure.* BMC Bioinformatics, 2010. **11**: p. 431.
67. Sadreyev, R. and N. Grishin, *COMPASS: a tool for comparison of multiple protein alignments with assessment of statistical significance.* J Mol Biol, 2003. **326**(1): p. 317-36.
68. Sadreyev, R.I., et al., *COMPASS server for remote homology inference.* Nucleic Acids Res, 2007. **35**(Web Server issue): p. W653-8.
69. Sadreyev, R.I.a.G., Nick V., *Accurate statistical model of comparison between multiple sequence alignments.* Nucleic Acids Research, 2008. **36**(7): p. 9.
70. Edgar, R.C. and K. Sjolander, *A comparison of scoring functions for protein sequence profile alignment.* Bioinformatics, 2004. **20**(8): p. 1301-8.
71. Soding, J., *Protein homology detection by HMM-HMM comparison.* Bioinformatics, 2005. **21**(7): p. 951-60.
72. Madera, M., *Profile Comparer: a program for scoring and aligning profile hidden Markov models.* Bioinformatics, 2008. **24**(22): p. 2630-1.
73. Eddy, S.R., *A Probabilistic Model of Local Sequence Alignment That Simplifies Statistical Significance Estimation.* PLoS Computational Biology, 2008. **4**(5).
74. Pearson, W.R. and M.L. Sierk, *The limits of protein sequence comparison?* Curr Opin Struct Biol, 2005. **15**(3): p. 254-60.
75. Altschul, S.F., et al., *Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.* Nucleic Acids Res, 1997. **25**(17): p. 3389-402.
76. Biegert, A. and J. Soding, *Sequence context-specific profiles for homology searching.* Proc Natl Acad Sci U S A, 2009. **106**(10): p. 3770-5.
77. Johnson, S., *Remote homology protein detection using hidden markov models*, in *Division of biology and biomedical sciences*. 2006, Washington university: Saint Louis, Missouri.
78. Jaakkola, T.a.D., M. and Haussler, D. *Using the Fisher kernel method to detect remote protein homologies.* in *International Conference on Intelligent Systems for Molecular Biology ; ISMB. International Conference on Intelligent Systems for Molecular Biology*. . 1999.
79. Liao, L.a.N., W. *Combining pairwise sequence similarity and support vector machines for remote protein homology detection*. in *International Symposium on Molecular Biology*. 2002.
80. Leslie, C., E. Eskin, and W.S. Noble, *The spectrum kernel: a string kernel for SVM protein classification.* Pac Symp Biocomput, 2002: p. 564-75.
81. Leslie, C.S., et al., *Mismatch string kernels for discriminative protein classification.* Bioinformatics, 2004. **20**(4): p. 467-76.
82. Saigo, H.a.V., Jean-Philippe and Ueda, Nobuhisa and Akutsu, Tatsuya, *Protein homology detection using string alignment kernels.* Bioinformatics, 2004. **20**(11): p. 8.
83. Lingner, T. and P. Meinicke, *Word correlation matrices for protein sequence analysis and remote homology detection.* BMC Bioinformatics, 2008. **9**: p. 259.
84. Hou, Y., et al., *Efficient remote homology detection using local structure.* Bioinformatics, 2003. **19**(17): p. 2294-2301.

85. Ben-Hur, A. and D. Brutlag, *Remote homology detection: a motif based approach.* Bioinformatics, 2003. **19 Suppl 1**: p. i26-33.

86. Kuang, R., et al., *Profile-based string kernels for remote homology detection and motif extraction.* J Bioinform Comput Biol, 2005. **3**(3): p. 527-50.

87. Rangwala, H. and G. Karypis, *Profile-based direct kernels for remote homology detection and fold recognition.* Bioinformatics, 2005. **21**(23): p. 4239-47.

88. Comin, M. and D. Verzotto, *Classification of protein sequences by means of irredundant patterns.* BMC Bioinformatics, 2010. **11 Suppl 1**: p. S16.

89. Vert, J.-P., *Classification of Biological Sequences with Kernel Methods.* 2006.

90. Weston, J., et al., *Protein ranking: from local to global structure in the protein similarity network.* Proc Natl Acad Sci U S A, 2004. **101**(17): p. 6559-63.

91. Ku, C.J. and G. Yona, *The distance-profile representation and its application to detection of distantly related protein families.* BMC Bioinformatics, 2005. **6**: p. 282.

92. Park, J., et al., *Sequence comparisons using multiple sequences detect three times as many remote homologues as pairwise methods.* Journal of Molecular Biology, 1998. **284**(4): p. 1201-1210.

93. Park, J., et al., *Intermediate sequences increase the detection of homology between sequences.* Journal of Molecular Biology, 1997. **273**(1): p. 349-354.

94. Bateman, A. and R.D. Finn, *SCOOP: a simple method for identification of novel protein superfamily relationships.* Bioinformatics, 2007. **23**(7): p. 809-14.

95. Jung, I. and D. Kim, *SIMPRO: simple protein homology detection method by using indirect signals.* Bioinformatics, 2009. **25**(6): p. 729-735.

96. Merkeev, I.V. and A.A. Mironov, *PHOG-BLAST--a new generation tool for fast similarity search of protein families.* BMC Evol Biol, 2006. **6**: p. 51.

97. Przybylski, D.a.R., B., *Consensus sequences improve PSI-BLAST through mimicking profile-profile alignments.* Nucleic Acids Research
2007. **35**(7): p. 9.

98. Przybylski, D.a.R., Burkhard, *Powerful fusion: PSI-BLAST and consensus sequences.* Bioinformatics, 2008. **24**(18): p. 7.

99. Jones, D.T. and M.B. Swindells, *Getting the most from PSI-BLAST.* Trends in Biochemical Sciences, 2002. **27**(3): p. 161-164.

100. Jones, D.T., *Critically assessing the state-of-the-art in protein structure prediction.* Pharmacogenomics J, 2001. **1**(2): p. 126-34.

101. Cuff, J.A., et al., *JPred: a consensus secondary structure prediction server.* Bioinformatics, 1998. **14**(10): p. 892-893.

102. Kabsch, W. and C. Sander, *Dictionary of Protein Secondary Structure - Pattern-Recognition of Hydrogen-Bonded and Geometrical Features.* Biopolymers, 1983. **22**(12): p. 2577-2637.

103. Li, W.Z. and A. Godzik, *Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences.* Bioinformatics, 2006. **22**(13): p. 1658-1659.

104. Lohr, S.L. and J.N.K. Rao, *Jackknife estimation of mean squared error of small area predictors in nonlinear mixed models.* Biometrika, 2009. **96**(2): p. 457-468.

105. Cheng, J.L. and P. Baldi, *Three-stage prediction of protein beta-sheets by neural networks, alignments and graph algorithms.* Bioinformatics, 2005. **21**: p. I75-I84.

106. Zhou, X.H., et al., *An analysis of the helix-to-strand transition between peptides with identical sequence.* Proteins-Structure Function and Genetics, 2000. **41**(2): p. 248-256.

107. Montgomerie, S., et al., *Improving the accuracy of protein secondary structure prediction using structural alignment.* Bmc Bioinformatics, 2006. **7**: p. -.

108. Laskowski, R.A., J.D. Watson, and J.M. Thornton, *ProFunc: a server for predicting protein function from 3D structure.* Nucleic Acids Research, 2005. **33**: p. W89-W93.

109. Pandey, G., V. Kumar, and M. Steinbach, *Computational Approaches for Protein Function Prediction.* 2006, Department of Computer Science and Engineering, University of Minnesota, Twin Cities.

110. Frenkel-Morgenstern, M., H. Voet, and S. Pietrokovski, *Enhanced statistics for local alignment of multiple alignments improves prediction of protein function and structure.* Bioinformatics, 2005. **21**(13): p. 2950-2956.

111. Camon, E., et al., *The Gene Ontology Annotation (GOA) Database: sharing knowledge in Uniprot with Gene Ontology.* Nucleic Acids Research, 2004. **32**: p. D262-D266.

112. Borgwardt, K.M., et al., *Protein function prediction via graph kernels.* Bioinformatics, 2005. **21**: p. I47-I56.

113. Dobson, P.D. and A.J. Doig, *Distinguishing Enzyme Structures from Non-enzymes Without Alignments.* Journal of Molecular Biology, 2003. **330**(4): p. 771-783.

114. Thompson, J.D., et al., *BAliBASE 3.0: Latest developments of the multiple sequence alignment benchmark.* Proteins-Structure Function and Bioinformatics, 2005. **61**(1): p. 127-136.

115. Forslund, K. and E.L.L. Sonnhammer, *Predicting protein function from domain content.* Bioinformatics, 2008. **24**(15): p. 1681-1687.

116. Nakai, K. and P. Horton, *PSORT: a program for detecting sorting signals in proteins and predicting their subcellular localization.* Trends in Biochemical Sciences, 1999. **24**(1): p. 34-35.

117. Guda, C. and S. Subramaniam, *pTARGET: a new method for predicting protein subcellular localization in eukaryotes (vol 21, pg 3963, 2005).* Bioinformatics, 2005. **21**(24): p. 4434-4434.

118. Park, K.J. and M. Kanehisa, *Prediction of protein subcellular locations by support vector machines using compositions of amino acids and amino acid pairs.* Bioinformatics, 2003. **19**(13): p. 1656-1663.

119. Przybylski, D. and B. Rost, *Powerful fusion: PSI-BLAST and consensus sequences.* Bioinformatics, 2008. **24**(18): p. 1987-1993.