Evolution-aware Protein Structure Comparison and Applications in Protein-Protein Interaction Prediction

A Thesis

Submitted to the Faculty

of

Drexel University

by

Chunyu Zhao

in partial fulfillment of the

requirements for the degree

of

Doctor of Philosophy

June 2016

Acknowledgements

I would like to thank my advisor Dr. Ahmet Sacan for his continuous support and guidance in leading my research. His expertise and patience has shaped me into the scientist that I am today, for which I am most grateful!

In addition, I would like to thank my committee members: Dr. Lin Han, Dr. Andres Kriete, Dr. Santiago Ontanon, and Dr. Will Dampier for their support and guidance. They have provided invaluable feedback on my projects and steered me along the right direction.

I also would like to thank my colleagues at the Center for Integrated Bioinformatics: Rehman Qureshi and Ceylan Tanes for their valuable discussions. I want to thank my friends, Yizhou Zang, Prescial Paz, Joyanna Friedman and Xiaoli Song, for always being there for me through good times and bad, as well as their encouragement.

I also want to take this opportunity to express my gratitude to Natalia Broz, Danielle Crocker, Estella Macke, Lisa Williams and other wonderful Biomed staffs who have helped me along the way.

Special thanks to my parents for their unconditional love and support!

Table of Contents

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Evolution-aware Protein Structure Comparison and Applications in Protein-Protein Interaction Prediction

Chunyu Zhao

Ahmet Sacan Supervisor, Ph.D.

Comparison of protein structures provide insights into the function and interactions of proteins and enhance our understanding of biomolecular mechanisms driving life and disease. Available protein structure comparison methods are based solely on the 3D geometric similarity, limiting their ability to detect functionally relevant correspondences between the residues of the proteins, especially for distantly related homologous proteins. However, non-geometric features, carried in primary sequence and evolutionary history of proteins, contain valuable information that can enhance detection of such similarities.

In this study, we introduced a new method to incorporate additional biochemical and evolutionary features of the proteins being compared. We proposed UniScore as a new protein similarity score, which integrates geometric similarity, sequence similarity, conservation similarity, and evolutionary profiles of the proteins. We further developed a corresponding UniAlign algorithm for finding structural alignment of proteins with near-optimal UniScore. We evaluated UniAlign in terms of the consistency between the alignments it produces with human-curated alignments, calculated by the fraction of correctly aligned residues. Experimental results show that UniAlign outperforms other structural programs in aligning proteins from the NCBI’s human-curated Conserved Domain Database.

UniAlign's ability in detecting functionally important structural similarities is utilized in an application to discover interactions between HIV-1 ENV protein (gp41 and gp120) and human proteins. Structural compatibility of an HIV-human interaction pairs are evaluated via geometric, biochemical, and evolutionary features and a prediction model is developed using a Support Vector Machine. This provides the first model for prediction of interactions that can also generate a protein-protein 3D complex. The results of the HIV-human interaction study have discovered novel virus-host interactions as well as potential clinical targets for therapeutic intervention.

CHAPTER 1: BACKGROUND

1.1 Protein Structure Alignment

Structural alignment can reveal the distant evolutionary history of the protein, that sequence alignment alone is not capable to capture, and thus plays an essential role in understanding the function of the protein [1]. Protein structure alignment identifies functionally equivalent residues in proteins and is often used as the gold standard for improving multiple sequences alignment [2]. Structural alignment has also been widely used for improving organize and classify known structures [3], and infer the function of newly discovered proteins. To be specific, proteins can still share similar structures without any detectable sequence similarity, structure alignments have been used to improve and being gold standard for sequence alignment [4]. Also, assuming an evolutionary continuity of structure and function, identification of structure similarities could elucidate the possible function of newly discovered proteins. Despite the importance of the alignment problem and the recent advances in the field, there is yet no widely accepted method for structural alignment. Large scale comparisons of existing methods have concluded that there is no single best method that works well for all proteins [5].

Various pairwise protein structure alignment programs have been developed, differing in their representation of protein structure, the scoring function used to evaluate the “goodness” of an alignment, and the optimization algorithm used to search for the optimal alignment with respective to the scoring function [6-11]. For example, DALI [12] uses Monte-Carlo procedure after the initial structural alignment to minimize the intra-structural distance of aligned substructures. CE [7] also uses fragment assembly to build the initial set of equivalences similar with DALI, yet generates the final alignment by gradually adding new eight-residue fragments to the existing alignment. TM-align [8] extends the approaches of STRUCTAL [10] and SAL [11] by using TM-score rotation matrix instead of RMSD rotation matrix and extend the initial guess of equivalent residues by iterative residue-level dynamic programming. FATCAT[9] is a flexible alignment, adopting aligned fragment pairs (AFP) – based dynamic programming and allowing multiple rotational frames resulting from protein flexibility or evolutionary divergence.

The common goal of protein structure alignment methods is to identify a set of structurally-similar residue pairs from each protein. Different methods score the good alignment differently, which can impact the performance of that method. And generally there are two strategies to find good alignment: directly searching for the optimal alignment by piecing together small aligned substructures, and iteratively superposition and residue pair collection steps. TMalign belong to the first group, whereas Dali, CE, and Deepalign belong to the second group. Most methods also rely on a structure superposition procedure (such as that due to Kabsch), which finds the transformation (i.e. rotation and translation) to optimally match the aligned pairs, in terms of their RMSD.

Speaking of what constitute a good alignment, we need to consider what leads to protein structural similarity. Structural similarities between different proteins could either result from the evolution from a same ancestor (remote structural homologs) or from convergent evolution (structural analogs). During the evolution of the protein, both the functional sites on the surface of the protein and the hydrophobic core which is essential to maintain the structural integrity are in general remain relative conserved [13]. However, most of the available algorithms align protein structures solely based on 3D geometric similarity, and are limited in their ability to find functionally relevant correspondences between the residues of the proteins, especially for distantly related homologous residues.

The scoring function is used to evaluate how good an alignment is and recognize the optimal alignment among all the candidates. Common scoring functions utilize the root mean square distance (RMSD) between the aligned residues, taking into account the length of the proteins. Traditionally, protein structure alignment has been described as a geometric optimization problem, prudently optimizing the geometric superposition of proteins. This limits the usefulness of the resulting alignments, requiring researchers to seek additional validation from the amino acid types or catalytic activity of the aligned residues. This is evident from the presence of alternative alignments that are equally good in terms of the quality of their geometric superposition, but that vary widely in terms of their accuracy in identifying functionally and evolutionarily equivalent residues [14]. For example, it is possible to achieve a good geometric superposition of immunoglobulin, despite misaligning a conserved disulfide bridge (Gerstein and Levitt, 1998). Furthermore, pure geometric information based structure alignment programs are found highly sensitive to conformational changes [15].

Could extra information gained from the evolution history of natural protein improve the ability of structure alignment method to find homologous relationships? It has been suggested that structural alignment can benefit from the evolutionary information provided by the alignment of homologous proteins [1, 14]. This is not surprising since information extracted from homologous proteins represent general features of the protein family and allow the identification of similarity to a remote sequence or family, even when the similarity to each individual aligned sequence is not significant [16].

In recent years, researchers have begun to integrate sequence similarity information into the scoring function. For example, Formatt (Daniels, et al., 2012) incorporates amino acid substitution matrices derived from evolutionarily-related protein pairs when constructing the alignment. Deepalign (Wang, et al., 2013) incorporates the BLOSUM mutation matrix, a local substructure mutation matrix, and hydrogen-bonding similarity into its scoring function. While these methods utilize the amino acid similarity as described by the BLOSUM substitution matrix, they do not utilize evolutionary information available from the history of the proteins being aligned.

However, the so-called evolutionary distance in these methods is just a simple transformation of the BLOSUM mutation matrix. Considering that the BLOSUM matrix, which is derived from close homologs, is not very sensitive to distant homologs. On the other hand, sequence profiles, initially introduced for detecting distantly related proteins, are indeed more sensitive at detecting remote homologs [17].

Motivation

During the evolution, functional sites on the surface of the protein as well as the hydrophobic core maintaining the structural integrity are well-conserved. However, available protein structure alignment methods align protein structures based solely on the 3D geometric similarity, limiting their ability to detect functionally relevant correspondences between the residues of the proteins, especially for distantly related homologous proteins. At the same time, it is well established that evolutionary profiles have a dominant advantage over sequence-based alignments, and provide greater accuracy in fold recognition [18] and protein classification tasks [19].

Considering that protein structural alignment is widely used to identify homologous residues (encoded by the same codon in the genome of a common ancestor) of the proteins compared, structural similarity seems not enough to capture the similarity between amino acid residues.

Motivated by this, the goal of our proposed structure alignment model is to recognize the maximal number of evolutionarily important residues as being structurally equivalent with minimal spatial deviations after the optimal rotation and translation. In order to accomplish this, we first need a scoring function that can capture the optimal alignment among several alignments during the heuristic searching, and secondly we need to apply properly the new scoring function to the structural alignment.

Aim 1) UniScore: a Novel Protein Similarity Score

Even though we all know it when we see it, it is still a challenge to develop an objective scoring function to evaluate the similarity of protein structures, both geometrically and evolutionarily. In fact, determining the proper way to integrate evolutionary similarity metrics into the construction of the scoring function of the protein structure program has proved surprisingly difficult. RMSD, MaxSub, LG-score, TMscore and etc., are all widely used to in the protein structure alignment programs to evaluate the geometric similarity between two protein structures. Yet, many researchers have observed that structurally-equal good alterative alignments may be a significant contributor to the overall discrepancy of the whole structures. Thus, it is time to integrate other sources of protein similarity besides geometric similarity to the scoring function to better measure the similarity of two proteins.

In this study, we propose **UniScore**, a new scoring function, which integrates geometric similarity, sequence similarity, and evolutionary information of the proteins. While sequence similarity has previously been investigated, to the best of our knowledge, this is the first study to systematically utilize different conservation and evolutionary profiles in structure alignments.

Aim 2) UniAlign: Protein Structure Alignment Meets Evolution

Protein structures are usually modeled as rigid 3D coordinates of atoms. And the modeling of aligning two proteins structures can be stated in two ways:

1) The alignment of two protein structures can be modeled as an optimization problem to minimize the distance between two proteins structures after a specific rotation and translation.

2) The alignment of two protein structures is to find the optimal rotation and translation matrix which gives us the largest non-continuous fragments such that after rotation their distance is below a predefined threshold in 3D space [20].

Equally important to the scoring function is the heuristic iterations involving a rotation matrix calculation and the dynamic programming algorithm to maximize the similarity score function, subject to the constraints that the weighted distance deviation of two aligned structures minimal. Using this new similarity score UniScore, we implement **UniAlign**, a new protein pairwise structure alignment algorithm, which focuses on identifying not only structurally equivalent, but also evolutionarily favorable residue alignments. We demonstrate that compared to other methods, the alignments generated by UniAlign are in better agreement with hand-curated reference alignments. Furthermore, for difficult cases when UniAlign and other methods fail to generate good alignments, we propose family-specific alignment models to drastically improve the alignments.

1.2 Prediction of HIV-1, Human Protein-Protein Interactions

Human immunodeficiency virus type I (HIV-1) uses host surface proteins to gain entrance into the host cell. Interaction between HIV-encoded proteins and human proteins is important in the course of HIV-1 infection [21]. Thus, understanding the protein-protein interaction (PPI) between HIV-1 and human proteins provides critical insights into how the pathogen manipulates the biological pathways and processes of the host and subsequently assists designing new therapeutic approaches. Computational approaches for protein interaction in the pathogen-host context are of significant value as large-scale experimental characterization of these interactions is expensive in terms of time and money [22].

Several computational PPI methods have been applied for HIV-1 - human interactions. Tastan et al. integrated multiple features information including Gene Ontology (GO), properties of human interactome and sequence motifs, and employed random forest method to predict protein-protein interactions [23]. Evans et al. predicted possible interactions using the presence of conserved sequence motifs and counter domain in both HIV-1 and human proteins [24]. The rapid progress in structure determination technologies gave rise to the establishment and deposition of large-scale protein structure in Protein Data Bank (PDB), with over 80,000 protein structures currently deposited [25]. The central assumption in predicting in pathogen-host interaction prediction based on structural similarity is that, for those defined structures and associated interactions, proteins with similar structures or substructures might share same interaction partners. Doolittle et al has already applied structure similarity based method to predict interactions between HIV-1 and human proteins, using the Dali Database for structure comparisons [26]. Zhang et al proved that three-dimensional structural information predict PPIs with superior accuracy and coverage compared to predictions based on non-structural evidence, for both close and remote proteins [27]. Based on the hypothesis that proteins with similar structures share similar interacting partners, our previous study used a novel evolution-aware structure alignment method (UniAlign) to predict the interaction map between HIV-1 protein gp41 protein and all the human proteins [28].

Motivation

With the increasing amount of protein structural data, we gain more knowledge about protein-protein interactions. For example, there are localized regions on the protein surfaces that are conserved among structural neighbors that participate in PPIs [29]. In other words, protein interacts with its partners through the interface region on its surface. The protein-protein interface is defined as the contact region between two interacting proteins or two complementary chains [30]. And the properties of the protein – protein interface have been deeply studied. For example, compared with non-interacting residues, interacting residues are evolutionarily more conserved than the other surface regions [31]. Also, amino acid propensities vary significantly between interface region and other surface residues [32]. From an energetic perspective, the residues in the protein interface regions contribute unequally to binding, among which some of these residues, called 'hot spots', play exceptional roles [33]. PRISM is the first algorithm that uses structure and sequence conservation in protein interfaces for protein-protein interaction prediction [30].

The benefit of studying interface scaffold is that regardless of dissimilar global sequence or structure folds, proteins can still interact through similar interface scaffold [34]. Therefore, interface architectures, rather than global sequence or structure similarity, is used in our study to model protein complexes.

Aim 3) Prediction of PPIs between HIV-1 and Human

Computational approaches for protein interaction in the pathogen-host context are of significant value as large-scale experimental characterization of these interactions is expensive in terms of time and money. For the third specific aim, we first tried to predict the interaction between HIV-1 proteins and host proteins based on the structural similarity. The central assumption in predicting pathogen-host interaction based on structural similarity is that, for those defined structures and associated interactions, proteins with similar structures or substructures might share same interaction partners.

We present an application of evolution-aware structural alignment and supper vector machine (SVM) for predicting physical interactions between HIV-1 and human protein, based on the hypothesis that proteins with similar interface scaffolds share similar interaction partners. The benefit of using the interface architecture similarity over structural similarity is that, proteins can still interact through similar surface regions despite the dissimilar global structural folds. Using a support vector machine with a Gaussian kernel, we explored 18 features including geometric similarity, phylogenetic profile similarity, conservation similarity, contacting residues pair number and etc. After the feature selection, we achieved the best 10-fold cross-validation performance with a combination of 12 features. We used the trained and tuned SVM classifier to discover potential novel HIV-1 interacting partners for human proteins. Many predicted interactions had significant literature support, and we modeled the novel 3D interacting complex for HIV-1 envelope gp120 and gp41 proteins.

Our method does not count on other functional genomic information, such as co-expression or cellular localization, and may be served as an addition contribution into an integrative computational framework for predicting novel PPIs based on information from multiple sources.

CHAPTER 2: UniScore A NOVEL PROTEIN SIMILARITY SCORE

An objective scoring function of a protein structure alignment program should reflect how likely two residues shall be aligned such that the program will be able to align those functionally important residue pairs more accurately. Based on the observation that important residues are likely to result in a loss of function were they to mutate into other residues during the evolution, in this chapter, we recover the functional importance of residues by analyzing residue conservation among homologous proteins. While sequence similarity has previously been investigated, to the best of our knowledge, this is the first study to systematically utilize evolutionary information, including conservation score and sequence profiles, in structure alignments.

2.1 Introduction

Background

It is still a challenge to develop an objective scoring function to evaluate the similarity of proteins, both geometrically and evolutionarily, even though we all know it when we see it. A good alignment of two proteins not only reveals the structural conservation, which can be captured by pure geometric criteria, but also recognizes the evolutionary conservation during the evolutionary history of the proteins.

Speaking of measuring the geometric similarity of two protein structures, the most widely used metric is the root mean square deviation (RMSD) between two protein structures. Yet the RMSD value itself can be a very misleading indicator due to two reasons: no account of alignment coverage and bias the result towards flexible regions. Since RMSD doesn't consider the alignment length or coverage, apparently, a smaller RMSD would be obtained if only a small residues pairs were aligned. The relative RMSD with an alignment length as a constraint is a simple solution to this problem. Other alternative RMSD-based scoring matric includes: global distance test score (GDT-TS score), utilizing several different distance thresholds to identify multiple maximum substructures (average coverage of the target protein) [35]; LG-score, summing aligned pairs with a gradually decayed weights from 1 (distance=0A) to 0 (distance is at infinity) [10]; MaxSub, setting the $d0$ to 3.5Å [36]. However, these RMSD-based scores show a power-law dependence on the protein size. For example, an absolute value of GDT=0.4, may indicate a significant similarity for proteins with 400 residues, yet it is close to a random selection for proteins with 40 residues. These scoring functions' dependence on the protein size make their absolute value meaningless. Then TM-score was proposed to remove the similarity score's dependence on the protein size and radius of gyration by using an empirical size-dependent $d0$ instead of a fixed cutoff distance [37]. The assumptions that this size-dependent distance cutoff $d0$ build on is: the aligned proteins are globular proteins and more importantly aligned in a predetermined sized L (either shorter length of the two proteins or the average size). However, in reality, the length of the alignment does not necessarily relate to the size of proteins aligned, especially for multi-domain proteins. And the SP-score was to remove the size dependency by utilizing a normalization pre-factor and the idea of effective alignment length [38]. Another significant contribution that TM-score made is to consider all the residues when evaluating the structural similarity of two proteins, while the other scoring methods mentioned before only consider those residues within the distance cutoff and ignoring the spatial information of those residues outside. The second problem inherent in RMSD is that it is very sensitive to those badly aligned or flexible regions or local structural changes, since all residues in the structure are equally weighted [39]. Consequently, the position-weighted RMSD, have been proposed, which assigns weights directly based on the distance between two superimposed atoms, in form of Gaussian function [40].

All the geometric similarity scores mentioned before have been applied in their own protein structure alignment programs. Apparently, only geometric information is included, while protein sequence and evolutionary information are omitted, in the scoring function of most protein structural alignment programs. Many researchers have observed the probability of finding equally good alternate alignments for many structure pairs, and further pointed out that those alterative alignments may be a significant contributor to the overall discrepancy of the whole structures [14]. For example, some shifted alignments are as good as the reference alignments based on the measurement of RMSD and the alignment length, yet are clearly wrong since the conserved residues are not correctly aligned (refer to the “hard to align” pairs of immunoglobulin) [16]. What’s more, another study reported that structure alignment methods based on pure structural similarity sometimes suffer from conformational changes [15]. A large-scale comparative analysis of protein structure alignments reported that current structure alignment methods still misalign 11-19% of the conserved core regions indicated in CDD [14]. Thus, it is time to integrate other sources of protein similarity besides geometric similarity to the scoring function to better measure the similarity of two proteins. And it is also clear that even protein structure alignment can benefit from the evolutionary information between protein sequences, gained from the multiple sequences alignments.

As a matter of fact, three research groups have tried to improve the quality of the structural alignments by integrating the sequence similarity and evolutionary similarity to the scoring function, and each of them will be briefly introduced in the following.

Formatt [41] incorporates the amino acid substitution matrices derived from evolutionarily-related protein pairs when constructing the multiple structural alignments. They utilized the same structure-sequence conservation score introduced in Staccato [42] as an objective criterion. For each column c in the multiple sequence alignment (MSA), the sequence-structure conservation score is a combination of structural and sequence scores:

$$Cons(c) = w \times Cons_{seq}(c) + (1 - w) \times Con_{str} \tag{1}$$

where w is set to 0.5. Formatt chose the optimal alignment with lower (better) conservation score. However, this $Cons(c)$ of this position in the MSA is not length-invariant,

DeepAlign [43] takes into account BLOSUM mutation matrix, local substructure mutation matrices and hydrogen-bonding similarity to the protein similarity score, in addition to the TM-score-based structural similarity. In particular, the equivalence of residue i and residue j is evaluated by:

$$\mathrm{DEEPSCORE}(i,j) = \left(\max\left(0, \mathrm{BLOSUM}(i,j)\right) + \mathrm{CLESUM}(i,j)\right) \times v(i,j) \times d(i,j) \tag{2}$$

where BLOSUM and CLESUM measure the evolutionary distance of two proteins at the sequence and local substructure levels, respectively, $v(i,j)$ measures the hydrogen-bonding similarity and $d(i,j)$ is the geometric similarity score (TM-score).

To the best of our knowledge, all the existing protein structural alignment programs that go beyond the spatial proximity only added the sequence similarity information gained from the sum-of-pairs of BLOSUM mutation matrix to their scoring function. In this study, we proposed UniScore, a protein similarity score, which systematically integrates residue conservation information as well as evolutionary profiles of the proteins besides the geometric similarity.

UniScore Overview

UniScore is a protein similarity score which incorporate various similarity scores from different biological sources. Based on the observation that important residues are likely to result in a loss of function were they to mutate into other residues during the evolution, we recover the functional importance of residues by analyzing residue conservation among homologous proteins. While sequence similarity has previously been investigated, to the best of our knowledge, this is the first study to systematically utilize evolutionary information, including conservation score and sequence profiles, in structure alignments.

While we utilize evolutionary information in structure alignment, we do not abandon other types of information that can help determine residue equivalences. We define UniScore as the weighted average of various sources of protein similarity measures. Specifically, for a residue pair i and j from two proteins being aligned, UniScore is defined as:

$$UniScore(i,j) = w_{\mathrm{geo}} \times Uni_{\mathrm{geo}}(i,j) + w_{\mathrm{pro}} \times Uni_{\mathrm{pro}}(i,j) + w_{\mathrm{con}} \times Uni_{\mathrm{con}}(i,j) + w_{\mathrm{seq}} \times Uni_{\mathrm{seq}}(i,j) + w_{\mathrm{sse}} \times Uni_{\mathrm{sse}}(i,j) \tag{3}$$

where Uni_{xxx} represents different types similarity measures, including geometric, profile, conservation, sequence, and secondary structure similarity. The UniScore of an alignment is the sum of the UniScores of the aligned residue pairs. The weights w_{xxx} adjust the contributions of different similarity measures and are normalized to a sum of one. For the sequence similarity component Uni_{seq}, we use the BLOSUM62 amino acid substitution matrix. We describe each of the other similarity measures in more detail below.

2.2 Geometric Similarity Score

We adopt the TMscore as the geometric component of UniScore, as TMscore has been proven to perform excellently as a geometric similarity measure [37].

$$\mathrm{TMscore} = \mathrm{MAX}\left[\frac{1}{L_N}\sum_{i=1}^{L_T}\frac{1}{1+(\frac{d_{i,j}}{d_0(L_{min})})^2}\right] \tag{4}$$

where L_N is the length of the target protein structure; L_T is the length of the alignment, $d_{i,j}$ is the distance between the ith and the jth aligned residue pairs, and d_0 is a scale to normalize the match different. It is worth noting that behind the 'MAX' is the heuristic search to find the optimally superposition with the maximum TM-score; and the summation includes all the aligned residues. TM-score ranges from 0 to 1. TM-score used a size-dependent cutoff distance to remove its dependence on the protein size,

$$d_0(L_N) = 1.24\sqrt[3]{L_N - 15} - 1.8 \tag{5}$$

thus TMscore is independent of protein size for a random structure pairs. As we have introduced before, despite the popularity of TM-score, there is still room for improvement, and that is the way TMscore used to remove size dependency effect.

The actual size of the proteins being aligned does not necessarily decide the actual alignment length. For example, for multi-domain proteins, it is possible that only one portion of the protein is being aligned [38]. SP-score introduce the definition of effective alignment length (L_e) to calculate a normalization prefactor and a fixed cutoff distance 4Å.

$$\text{SPscore} = \frac{1}{3\times L_e^{1-\alpha}} \sum_{d_{i,j}\le d_0} \left(\frac{1}{1+(\frac{d_{i,j}}{d_0})^2} - 0.2\right) \qquad (6)$$

where $d_{i,j}$ is the distance of two aligned residues, d_0 is chosen to be 4Å, α is optimized to be 0.3, and L_e is the so-called effective alignment length. In SP-score, only the meaningfully aligned pairs contribute to the final score, thus there is no apparent correlation between the score and the quality of the full-length alignment quality.

In our study, we also explored the idea of effective alignment length (L_e). Instead of calculating a pre-normalization factor like what SP-score did, we used the effective alignment length to calculate the size-dependent threshold $d_0(L_e)$, compared to the size of the smaller protein $d_0(L_{min})$ used in TM-score.

$$d_0(L_e) = 1.24\sqrt[3]{L_e - 15} - 1.8 \qquad (7)$$

The way to calculate the L_e is same with SPscore [38]: L_e is the total number of core aligned residues ($d_{i,j} \le 2\times 4$Å) plus the average number of surrounding residues in two proteins that are within $3d_0$ from any core residues. Unlike SP-score, we summed up all the aligned residues to get the Uni_{geo}, so that the full-length of the aligned can be reflected in this score.

$$Uni_{\text{geo}} = \text{MAX}\left[\frac{1}{L_T}\sum_{i=1}^{L_T}\frac{1}{1+(\frac{d_i}{d_0(L_e)})^2}\right] \tag{8}$$

where L_T is the length of the alignment, $d_{i,j}$ is the distance between the ith and the jth aligned residue pairs, and d_0 is the scale to normalize the match difference depending on L_e.

2.3 Evolutionary Information

The variability pattern of an amino acid observed in each column of the multiple sequences alignment (MSA) tells the story of evolutionary pressure, mutation, recombination, and genetic drift during the evolution of that protein. According to the neural model of molecular evolution, most substitutions observed are neutral; rather than representing improvements in a protein, they actually indicate how tolerant that protein is to change at that position. In other words, the substitution rate of a protein is reversely correlated with the functional constraints acting on that protein [44]. Thus, estimating the conservation value of every amino acid of a protein is of tremendously importance.

Conservation score constitutes the second part of the UniScore (Uni_{con}), and is aimed to reflect the structural and functional importance of each amino acid of a protein. Uni_{con} is inferred from how conserved a residue appears in a multiple sequence alignment of a set of evolutionarily related proteins. Usually, a set of similar protein sequences can be characterized by a multiple sequence alignment (MSA) within common sequence domains (in the case of protein families) or just a small sequence region (in the case of protein motif). So the MSA is used in our study extract both sequence profiles and conservation scoring matrices [45]. In fact, we constructed the position specific score matrix (PSSM) or conservation score vector directly from the PSI-BLAST output. As has been mentioned in PSI-BLAST, the construction of PSSM is a multi-stage process, and at each stage a choice must be made among a number of alternatives choices [46]. We will introduce what we did at each stage in the following.

Multiple Sequences Alignment Construction

Given a protein, there are five steps to construct a multiple sequence alignment (MSA) and prepare it to be ready for further calculation: 1) identification of the entire homologous family of the interested protein; 2) pruning the raw homologs list to remove false positive; 3) construction of the multiple sequence alignment; 4) calculation of the sequence weights to remove redundancy; 5) estimation of the target frequencies in each position, as shown in Figure 1.

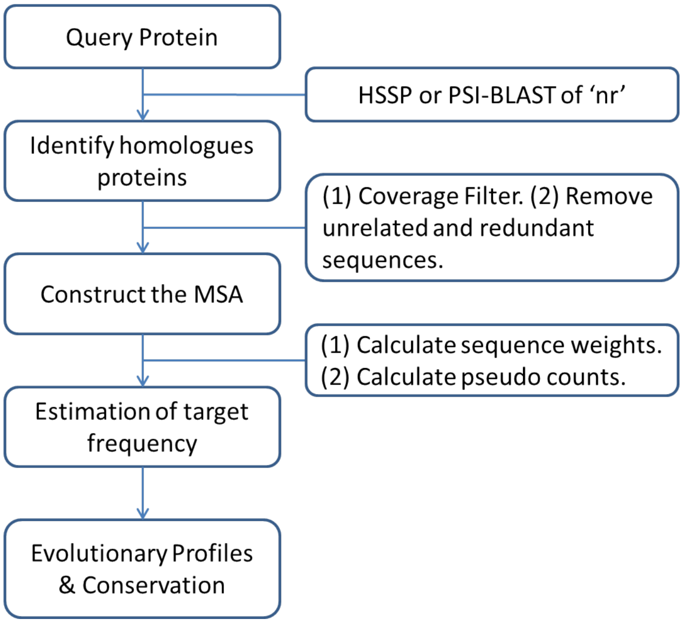

Query Protein
HSSP or PSI-BLAST of 'nr'
Identify homologues proteins
(1) Coverage Filter. (2) Remove unrelated and redundant sequences.
Construct the MSA
(1) Calculate sequence weights. (2) Calculate pseudo counts.
Estimation of target frequency
Evolutionary Profiles & Conservation

Figure 1 Multiple Sequences Alignment Construction

(1) Identifying Homologous Proteins Set

For the purpose of finding homologous proteins, local sequence alignment is better than global sequence alignment, since many proteins share only a portion of the complete sequence or a domain [47]. To better estimate the sequence identity at longer evolutionary distances, we preferred profile-sequence alignment rather than sequence-sequence alignment.

In order to create the set of sequence candidates homologous to the protein of interest, we first tried the Homology-derived Secondary Structure of Proteins (HSSP) database to extract the homologous protein list. If no result is found, then we ran the PSI-BLAST [46] against the NCBI Entrez non-redundant protein sequence database (as of 09/16/2013). Up to 1000 sequences were retained in the MSA with e-value cutoff of 0.005 and up to 3 iterations. This is our raw list of homologs candidates.

(2) Pruning the Raw Homologs List (Coverage Filter)

Similarity searching is effective because proteins sharing statistically significant sequence similarity can be inferred to be homologous and homologous proteins may share similar structures and functions. However, PSI-BLAST may also overestimate sequence identity for distantly related proteins by aligning only the most conserved regions of the sequence pairs [48]. Generally speaking, PSI-BLAST makes two types of errors: alignments to non-homologous regions and HOE alignments that start in a homologous region but extend into neighboring sequence regions. Thus we need to prune the raw candidate list so that we can further generate a more accurate alignment. To be more specific, we first enriched the raw blast result with the query sequence coverage and hit sequence (the sequence found by PSI-BLAST) coverage. The sequence identity reported by PSI-BLAST is the sequence identity of the aligned region, while what we need is the percentage of aligned region over the full length of the query (hit) sequence. Namely, query coverage equals to the percentage of aligned region length over the full length of query sequence; so is the hit coverage. After that, we further set up a minimum query/hit coverage threshold to 20% and filter the raw homolog candidates list [49]. Basically, in order for a hit sequence enter the multiple sequence alignments, at least 20% of the target residues must be aligned with residues from the hit sequence; so is the hit sequence.

(3) Construction of the MSA (Similarity Filter)

After filtering the homologs set, we used the query sequence as the master and constructed the multiple sequence alignment, with or without using an external multiple sequence alignment program MUSCLE[50]. If MUSCLE is not used, we used gap character for those un-aligned regions in all the hit sequences. Several filters were used to make sure only true homologous would enter the profile construction.

In order to remove those too distantly related sequences, we applied the size-dependent similarity threshold introduced by HSSP[51]. They determined the homology threshold empirically (given sequence length L):

$$t(L) = \begin{cases} - & L < 10 \\ 290.15 \times L^{-0.562} & 10 \leq L \leq 80 \\ 24.8 & L > 80 \end{cases} \quad (9)$$

For a sequence with length L, Sequence similarity equal to or above $t(L)$ infers structural homology, and can further stay in the multiple sequence alignment. For $L < 10$, any value of sequence similarity is consistent with any degree of structure similarity. The sequence similarity refers to the percent identity of amino acids. It is worth noting that the alignment length referred to the number of aligned residue pairs excluding gaps, rather than the length of the query protein; otherwise, it would underestimate the alignment quality of the local alignment.

In order to remove the redundancy of the multiple sequence alignment, the sequences are further purged leaving sequences with mutual identity lower than 98%. There is no gap in the master sequence, and any columns that involve gap inserted into the master protein are simple ignored. We have not further prune the M into a simpler ‘reduced’ one, as PSI-BLAST.

When the query protein is just a specific range of a whole chain, we used the whole chain sequence to generate the MSA and further extracted the corresponding region of it.

(4) Calculation of Sequence Weights

The generated MSA will be used to not both calculating conservation score vector but also building sequence profiles. As a result, normalizing against redundancy and compensating for over-representation among MSA is a must. For example, by down weighting the contribution of redundant sequences to a position specific score matrix (PSSM), it could be more sensitive to distant relationships [52]. It is mentioned by PSI-BLAST it is a mistake to give the same weights to all the sequences of the alignment that hen constructing a score matrix from a multiple sequence alignment. In our study, two different sequence weights methods are included: the first one used in HSSP and the second type used in PSI-BLAST.

The first type of sequence weight relates to the sequence's average genetic distance to all the other sequences [44, 53]. The more close neighbors a sequence have, the smaller the sequence weights is. For the ith sequence in a multiple sequence alignment S, the sequence weight is:

$$w_i = \frac{1}{N-1}\sum_{j \neq i}^{N} d(S_i, S_j) \tag{10}$$

where N is the number of homologous proteins; $d(S_i, S_j)$ is the distance between the ith sequence and jth sequence (measured in percentage identity):

$$d\left(S_i, S_j\right) = 1 - ident(S_i, S_j) \tag{11}$$

The second type of sequence weight relates to the amino acid type diversity observed at each aligned position. Position-based sequence weights assigned each different residue an equal weight at a position, and then divided that weight equally among the sequences sharing that same residue [44, 52]. The weight of the ith sequence at position x is:

$$w_{i_x} = \frac{1}{k_x \times n_{i_x}} \tag{12}$$

where k_x is the number of amino acid types presented in column x, and n_{i_x} is the frequency of the ith sequence's amino acid at position x. For the ith sequence, the contributions from every position are summed up to give the sequence weights:

$$w_i = \frac{1}{N} \sum_{x=1}^{N} w_{i_x} \tag{13}$$

Some changes were made in our implementation: gaps were treated as the 21st distinct character; both sequence weights were normalized to sum equals to one. From now on, in speaking of the observed residue frequencies of a column, we shall mean its weighted frequencies instead of raw frequencies.

(5) Estimation of Target Frequency

Contrary to the redundancy issues that sequence weights tried to solve above, incomplete sample set of the homologs may also bias the observed residue frequencies and fail to build a statistically robust profile solely from the occurrence in the MSA [54-56]. Thus for further estimation of the target frequency, we employed the pseudo-counts method proposed by Henikoff [54], which used the prior knowledge of the probabilities of residue occurrences and residue-residue substitutions to generate the residue pseudo-counts g_i.For a given column C in the multiple sequences alignment, the pseudo-counts frequencies g_i is constructed using the following formula:

$$g_i = \sum_{j=1}^{N} \frac{f_j}{P_j} q_{ij} \tag{14}$$

where, f_j is the observed residue frequency; q_{ij} are the target frequencies calculated from BLOSUM62 matrix:

$$q_{ij} = P_i \cdot P_j \cdot e^{\lambda M_{ij}} \tag{15}$$

where P_i is the background probabilities gained from [57]; M_{ij} is the BLOSUM62 probabilities [58]; λ is the scaling factor, for BLOSUM62, $\lambda = 0.5\times \log(2)$.

For a given column C in the multiple sequences alignment, the expected frequencies Q_i was estimated as the mixture of (weighted) observed frequency f_i and pseudo-count frequency g_i:

$$Q_i = \frac{\alpha f_i + \beta g_i}{\alpha + \beta} \tag{16}$$

where α and β are the relative weights reflecting how strongly observed and pseudo-count residue frequencies contributes respectively. α is the total number of counts in column C. And the total number of pseudo-counts at position i is calculated using the following equation:

$$\beta = m \times R_C \tag{17}$$

where R_C is the effective number of independent observations of each column(including gap characters), and equals to the number of different amino acid types at column C. Since our pseudo-counts number β takes into account the residue diversity of various columns, thus fewer pseudo-counts for conserved column. And parameter m is set to the optimal value of 5 [54].

Sequence Profile

Sequence profile of a given multiple sequence alignment specifies a preference for each of 20 standard amino types at each position. Information extracted from homologous proteins may represent general features of the family, and allow the prediction of similarity to a remote sequence or family, even when the similarity to each individual aligned sequence is not significant [56]. More importantly, different positions in the same protein may under different functional constraint, and thus some positions may tolerate some substitutions better than the others. In our study, we generated our own sequence profile (Position Specific Score Matrix, PSSM) rather than using native PSSM or HMM (Hidden Markov Models) profile produced by other programs, using all the previous introduced techniques, such as sequence weights, pseudo-counts, and estimated target frequency.

Conservation Score

Whereas the sequence profile describes the amino acid composition at each position of the protein, the conservation score describes the variability at each position. With conservation, we aim to capture equivalent residues that share a similar conservation level, but that may or may not share a similar amino acid composition. Generally speaking, the conservation value should be able to normalize against redundancy and bias in the MSA without loss of evolutionary information [44]. Three types of conservation scores are studied in UniScore (Uni_{con}): symbol entropy score, sequence variability and mutation data score. And the default calculation option of the conservation scores of a single protein from MSA is the sequence-weighted sum-of-pairs scheme [42].

The first type of conservation is based on the concept of entropy [59]. Shannon's entropy is a widely used term to measure diversity. For a column C in a given multiple sequences alignment S, the entropy is calculated by the estimated frequency of amino acid type R:

$$entropy(C) = -\sum_{R=1}^{20} f_{R_C} \cdot \log(f_{R_C}) \tag{18}$$

If all 20 amino acids are equally distributed in a column, then the entropy value of that column is $\log(20)$. Thus our entropy is normalized to [0,1] by dividing $\log(20)$. The conservation score equals to one minus the entropy, since small $entropy(C)$ means small variability and thus strong conservation. However, one of the biggest inherent problems with entropy-based conservation score is that gaps was not accounted. Also if we simply consider gap character as the 21st amino acid, then the gap-dominated positions would also be considered as evolutionarily conserved.

The other two types of conservation utilized the mutation data from a modified substitution matrix (the diagonal is changed) to quantify stereochemical variability in an aligned column [60].

The second type of conservation score is similar to the sequence variability in HSSP database, except that we used the BLOSUM62 matrix rather than the Dayhoff exchange matrix. Specifically, for a position C in a given multiple sequences alignment S, the conservation is defined as the weighted summation of residue similarities over all sequence pairs $(i \neq j)$:

$$cons(C) = \frac{\sum_{i,j}^{N} w_{ij} * M(S_{i_C}, S_{j_C})}{\sum_{i,j}^{N} w_{ij}} \tag{19}$$

where N is the number of sequences; w_{ij} is the weights for sequence pairs ith and jth sequence, which was introduced in HSSP to correct the uneven representation of sequences in the database, and related to each sequence pair's mutual distance in the sequence space. w_{ij} was defined as the fractions of amino acid mismatches over the alignment length L:

$$w_{ij} = 1 - \frac{1}{L} \sum_{i=1}^{L} \delta(S_{i_C}, S_{j_C}) \tag{20}$$

The third type of conservation score is a weighted sum-of-pairs measure [41, 42]:

$$cons(C) = \frac{\sum_{i=1}^{N} \sum_{j>i}^{N} w_i * w_j * M(S_{i_C}, S_{j_C})}{\sum_{i=1}^{N} \sum_{j>i}^{N} w_i * w_j} \tag{21}$$

where w_i is the previously mentioned sequence weight; S_{i_C} is the amino acid type at of ith sequence at position C; M is a modified BLOSUM62 matrix, by setting the diagonal elements to a constant (equal to the rounded average of original diagonal elements).

Evolutionary information derived from the larger number of available homologous proteins sequences could also powerfully guide analysis and prediction of protein-protein interfaces. Analysis of conservation patterns in binding sites has been explored by several groups, and the entropy based conservation developed by me is also used in some other programs developed by the other members in the lab, including PDBCIRCLEPLOT and SURFOLD.

2.4 Profile - Profile Substitution Score

The story of mutation, substitution, and genetic drift of one protein can be told from the amino acid patterns observed at each position of the multiple sequence alignments (MSA) of homologous proteins. A sequence profile represents the propensity of each amino acid to occur at each position of that protein. For each protein being aligned, we construct the MSA from the HSSP database of curated homologous proteins (Sander and Schneider, 1991). When HSSP does not contain an entry for a protein, we collect homologous proteins from a PSI-BLAST query [46] against NCBI's non-redundant sequence database (with E-value cutoff=0.005, 3 iterations, and a maximum of 1000 search results).

From the constructed MSA of homologous proteins, we generate the position specific score matrix (PSSM), following sequence weight and pseudo-count calculations as used in [46]. When comparing residues i and j, we calculate the score of aligning their PSSM columns as the sum-of-pairs of substitutions looked up from the BLOSUM matrix and weighted by the amino acid frequencies in the PSSM. Uni_{pro} of an alignment is then the sum of these scores for all aligned residue pairs. Another option of the similarity function is Person correlation coefficient, which was used in the earlier version of UniScore. Although there are other methods for comparing two profiles, there is no statistically significant difference between these methods [61].

2.5 Conservation Similarity Score Table

While the conservation score is routinely used to evaluate functional importance of residues, we are not aware of any study that aligns proteins based on conservation levels of the residues. While it is expected that highly conserved residues are more likely to align, no quantification of conservation-based similarity is available. Here, in order to systematically quantify likelihood of aligning residues with different conservation values, we generate a conservation similarity score table as illustrated in Figure 2, similar to the generation of the BLOSUM substitution matrix [58]. We generate the conservation values of the aligned residues for all proteins in the CDD reference alignment database and discretize these conservation values into 20 conservation categories by equal frequency binning. This essentially gives us a set of alignments where each residue is now encoded by a discrete conservation level. A conservation similarity scoring table tabulating the log-odds ratios of observing the alignment of any two conservation levels is calculated from these reference alignments. Uni_{con} between two residues can then be looked up from this substitution table. Notice in the heatmap shown in Figure 2 that alignment of not only the highly conserved residues, but also of highly variable residues is favorable. As a matter of fact, a protein-protein interactions study found dispersing hot spot within a large contact area rather than compactly clustering the conserved residues and suggested that maybe this is a strategy to sustain essential key interactions while still allowing certain protein flexibility at the interface. Thus surrounding residues form a flexible cushion for those conserved residues on the binding interfaces [62].

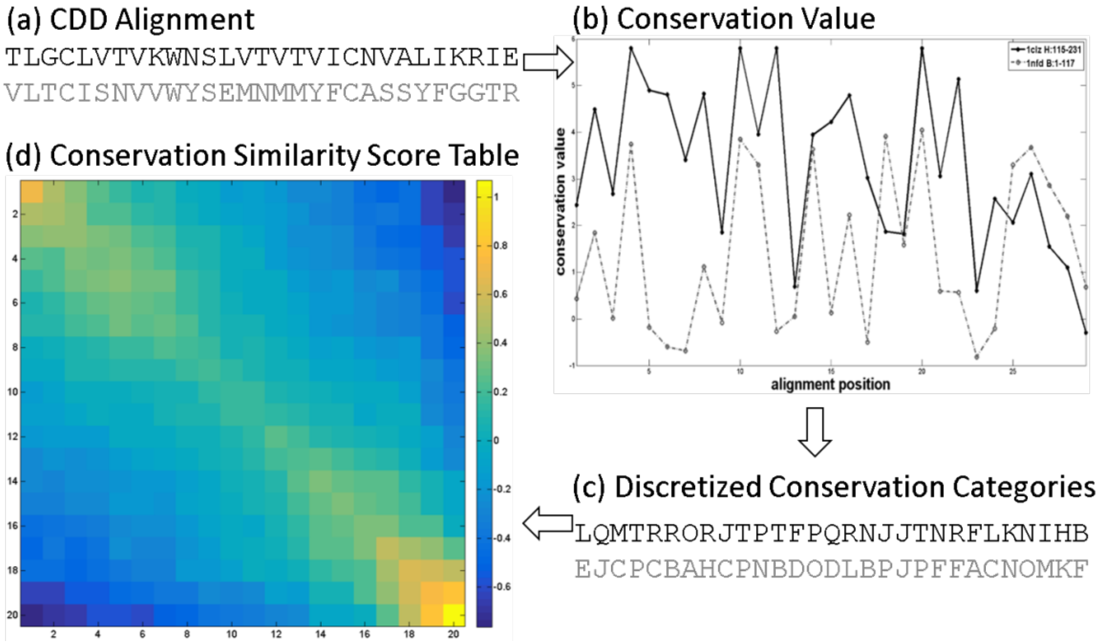

(a) CDD Alignment
TLGCLVTVKWNSLVTVTVICNVALIKRIE
VLTCISNVVWYSEMNMMYFCASSYFGGTR
(b) Conservation Value
1clz H:115-231
1nfd B:1-117
conservation value
alignment position
-1
0
1
2
3
4
5
6
5
10
15
20
25
(c) Discretized Conservation Categories
LQMTRRORJTPTFPQRNJJTNRFLKNIHB
EJCPCBAHCPNBDODLBPJPFFACNOMKF
(d) Conservation Similarity Score Table
2
4
6
8
10
12
14
16
18
20
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6

Figure 2 Generating the CDD Conservation score table. For each protein in the (a) CDD reference alignments, (b) the conservation values of the residues are calculated and (c) converted into one of 20 discrete conservation levels, encoded here by letters A (least conserved) through T (most conserved). (d) A conservation similarity score table shown as a heatmap here, is calculated using log-odds ratios of observing different conservation levels aligned in the encoded alignments.

Conversion of Sequence Conservation Values into Discrete Conservation Levels

We calculate the conservation values for all the residues of each protein appearing in the CDD database. These conservation values were then pooled together. The histogram in Figure 3 depicts the distribution of conservation values for all residues in all CDD proteins:

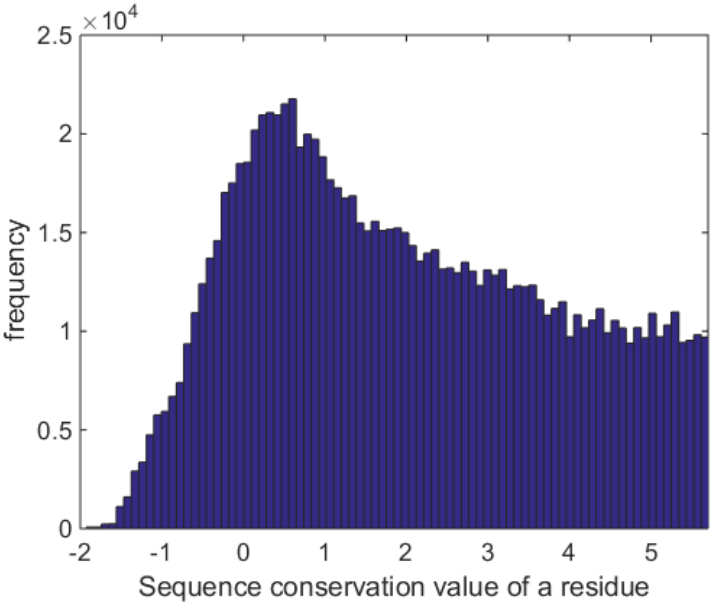


Figure 3 Distribution of conservation values for all residues in CDD

We discretize these conservation values into 20 categories by equal frequency binning. Selection of an "ideal" number of bins is not explored in this study. The number 20 is chosen for convenience, to obtain a matrix of similar size to the standard amino acid substitution matrices. The boundary values of these bins are as follows: -3.3988, -0.6083, -0.2473, 0.0258, 0.2694, 0.4990, 0.7323, 0.9803, 1.2601, 1.5656, 1.8831, 2.2196, 2.5741, 2.9530, 303311, 3.7395, 4.1919, 4.6559, 5.1426, 5.6313, 5.8000.

Calculation of the Conservation Similarity Score Table

Each residue in CDD is assigned into one of the discrete conservation levels. The pairwise CDD alignments are then encoded in alignment of these conservation categories. The log likelihood ratio of observing an alignment of two conservation categories is calculated from these reference alignments by:

$$ConsSimTable(a,b) = \ln \frac{f_{a,b}}{f_a f_b} \qquad (22)$$

where $f_{a,b}$ is the frequency of observing the conservation categories a and b aligned in the reference alignments and f_a, f_b are the background frequencies of observing these categories individually. Note that since we used an equal-frequency binning for defining the conservation categories, each of these background categories are roughly equal to 1/20.

Table 1 shows the generated Conservation Similarity Score Table, where the rows and columns represent each of the 20 conservation categories, ordered from least conserved to most conserved category (as defined by the bins described in the previous section).

Table 1 Conservation Similarity Score Table

0.723																			
0.474	0.456																		
0.360	0.399	0.402																	
0.254	0.238	0.337	0.321																
0.168	0.225	0.307	0.339	0.376															
0.074	0.142	0.235	0.277	0.322	0.290														
0.124	0.121	0.182	0.215	0.301	0.247	0.284													
-0.004	0.017	0.082	0.122	0.179	0.181	0.216	0.210												
-0.042	-0.019	0.070	0.078	0.116	0.109	0.111	0.181	0.193											
-0.154	-0.079	-0.012	-0.010	0.009	0.034	0.067	0.064	0.140	0.181										
-0.121	-0.112	-0.052	0.007	-0.025	-0.039	0.040	0.070	0.094	0.159	0.160									
-0.171	-0.162	-0.160	-0.147	-0.029	-0.038	-0.019	0.016	0.035	0.084	0.146	0.260								
-0.259	-0.202	-0.222	-0.219	-0.152	-0.089	-0.062	-0.045	0.014	0.061	0.100	0.190	0.235							
-0.264	-0.297	-0.247	-0.237	-0.202	-0.183	-0.178	-0.114	-0.071	0.011	0.060	0.119	0.240	0.334						
-0.317	-0.308	-0.251	-0.264	-0.189	-0.184	-0.210	-0.083	-0.052	0.004	0.062	0.093	0.191	0.274	0.340					
-0.428	-0.404	-0.367	-0.356	-0.266	-0.226	-0.237	-0.188	-0.094	-0.033	-0.052	0.063	0.134	0.243	0.273	0.333				
-0.478	-0.373	-0.466	-0.392	-0.238	-0.294	-0.301	-0.196	-0.137	-0.039	-0.004	0.016	0.115	0.239	0.261	0.338	0.532			
-0.425	-0.423	-0.496	-0.437	-0.369	-0.328	-0.320	-0.190	-0.176	-0.125	0.003	-0.036	0.034	0.142	0.188	0.236	0.447	0.628		
-0.628	-0.636	-0.557	-0.554	-0.476	-0.421	-0.378	-0.375	-0.271	-0.280	-0.271	-0.130	-0.135	0.061	0.047	0.219	0.405	0.585	0.729	
-0.764	-0.691	-0.677	-0.585	-0.585	-0.610	-0.514	-0.488	-0.414	-0.391	-0.352	-0.286	-0.252	-0.065	-0.056	-0.011	0.261	0.529	0.791	1.070

2.6 Secondary Structure Similarity

We obtain the secondary structure assignments from DSSP [63], or calculate it from the alpha carbon distances [8] when DSSP entry is not available. We then calculate a secondary structure scoring table (available in the supplementary documents) from the CDD reference alignments as log-odds ratio of observed substitutions. The resulting secondary structure substitution scoring table is shown in Table 2 (H: Helices, E: Strand, C: Coil).

Table 2 Secondary Structure Similarity Score Table

	H	E	C
H	0.171		
E	-0.299	0.367	
C	-0.065	0.055	0.028

2.7 Experimental Analysis

Heuristic Search Engine

We also implemented an iterative search algorithm similar to what was used in TM-score to find the spatially optimal superposition of the aligned structures with maximum TM-score [36, 37]. Given an aligned fragment that consists of L_{int} neighboring residues, they superposed the fragments according to Kabsch's rotation matrix. The correspondence was then updated by collecting all the residue distance after aligned less than d_0. The newly collected correspondence was then again superposed again and this procedure repeated until the optimal rotation matrix was converged. Due to the optimal superposition's sensitivity to the selection of the initial aligned fragment L_{int}, UniScore ran an iterative selection with different initial aligned fragment: $L_{int} = [L_T, L_{\frac{T}{2}}, L_{\frac{T}{4}}, L, 4]$, where L_T is the length of the alignment. When $L_{int} < L_T$, we shifted the fragments continuously from the N- to the C-terminus. The UniScore-rotation matrix was returned.

Results

In the first experiment, we focused on a simpler version of UniScore as the weighted arithmetic average of three sources similarity. Specifically, for an aligned residue pairs i and j UniScore is defined as:

$$Uniscore(i,j) = \frac{w_{\text{geo}} \times \text{geo}(i,j) + w_{\text{pro}} \times \text{pro}(i,j) + w_{\text{con}} \times \text{con}(i,j)}{(w_{\text{geo}} + w_{\text{pro}} + w_{\text{con}})} \tag{23}$$

where $geo(i,j)$ is the geometric similarity score of two aligned residues, denoted by a modified TM-score, using the idea of effective alignment length to remove size dependency; $pro(i,j)$ represents the evolutionary profiles score, denoted by the correlation coefficients between two position specific scoring matrixes (PSSMs); $con(i,j)$ refers to the residue conservation score, represented by entropy, sequence variability or the sum-of-pairs score from the conservation table we generated; Three weights ($w_{\text{geo}}, w_{\text{pro}}, w_{\text{con}}$), ranging from 0 and 1, are assigned to each score separately to modify their contributions to the overall UniScore.

There are two aims for validating the idea of UniScore: 1) to calculate a more accurate and more meaningful structural alignment given a protein pair; 2) to be used as a structural similarity measurement.

For Goal One, we tried three different combinations of scoring function and superimposing method:

(a) TMscore + standard RMSD

(b) UniScore + standard RMSD

(c) UniScore + weighted RMSD

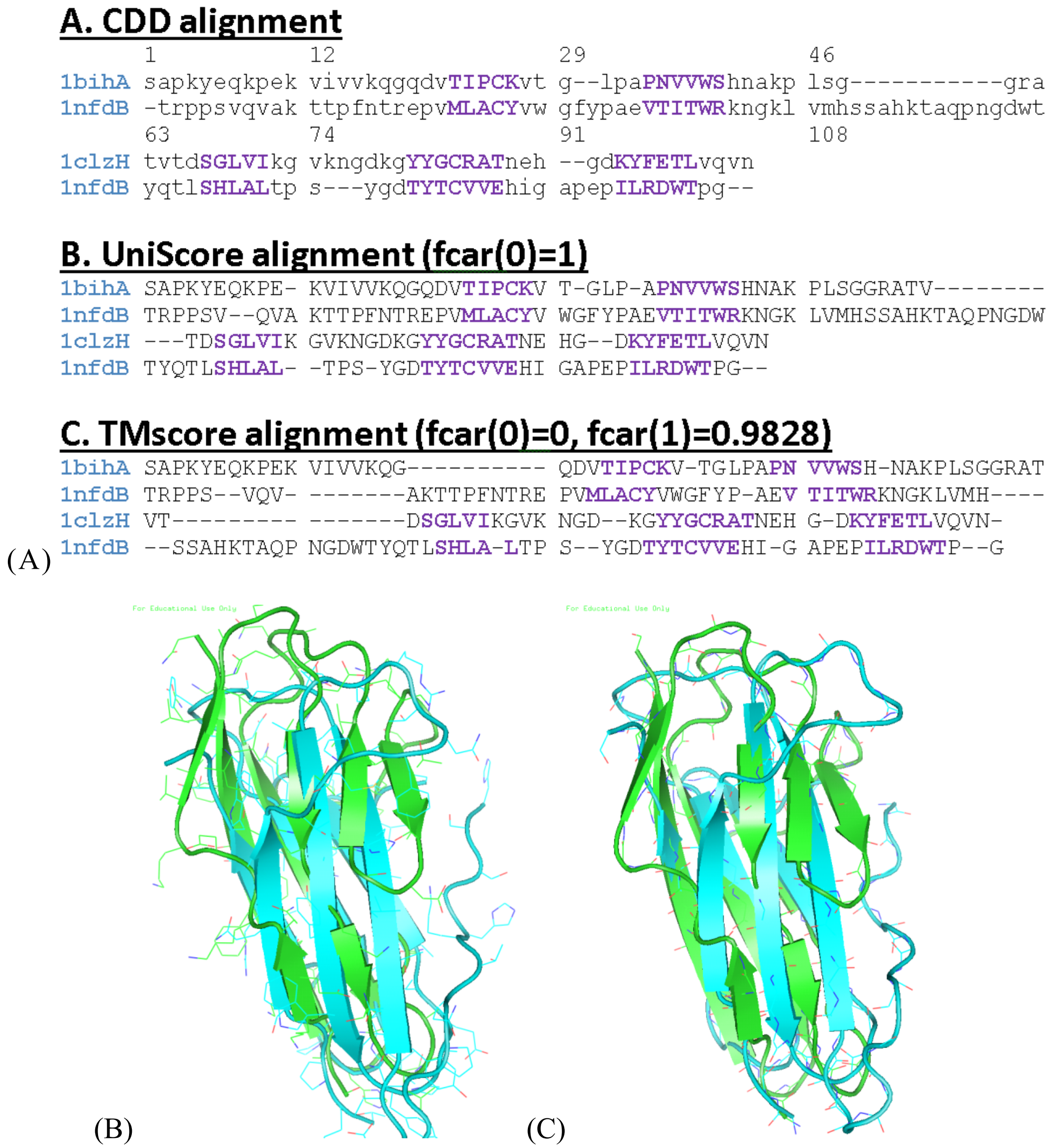
For the simplicity, we only employed the gapless threading as the initial alignment, and for comparison, only the geometric similarity scoring were used in the dynamic programming to collect new pairs. To be specific, for each fragment-extended pairs, these three different scoring schemes were used. The aim of this experiment is to see whether UniScore could better capture the evolutionarily structural equivalence and also whether weighted-UniScore RMSD would help to find the optimal alignment. 3642 homologous protein pairs in the CDD benchmark were examined. Fractions of correctly aligned residues were used to evaluate the quality of the sequence alignments generated by the structural alignment, results shown in Table 3 (details about the evaluation method will be introduced in next chapter).

There are 125 cases where using pure geometric score completely failed while UniScore could excellently generate biologically meaningfully alignment judged by manual alignment. Except in terms of RMSD, UniScore + weighted RMSD combination obtained the best performance in generating both structurally similar pairs and biologically meaningful pairs. One reason of this success is due to that UniScore can recognize the good alignment during the gapless searching, while pure geometric score might just miss it. One case study of 1bih A: 307-395 and 1HDM b: 88-185 will be shown in Figure 4.

Table 3 Comparison of Pure Geometric Score and UniScore

Method	fcar(0)	fcar(8)	TMscore	UniScore	RMSD
TMscore+ standard RMSD	0.8135	0.9575	0.6447	0.5565	2.0372
UniScore + standard RMSD	0.8475	0.9612	0.6435	0.5575	**2.0068**
UniScore + weighted-UniScore RMSD	**0.8723**	**0.9817**	**0.6720**	**0.5833**	3.4848

For Goal Two, we utilized a small-size benchmark dataset used by [64] to detect proteins in the Globins family from other family proteins. The dataset contained 200 proteins selected from representative ASTRAL database with less than 40% sequence homology. Among these 200 proteins, 20 were randomly selected from two distinct families: 10 proteins from Globins family (a.1.1.2 in SCOP) and 10 proteins from Serine/Threonine Kinases family (d.144.1.1 in SCOP). The remaining 180 proteins were randomly selected from four major SCOP classes of the same representative ASTRAL database. It is noteworthy that the 3D structures stored in ASTRAL are not actually whole proteins, but they are domains within the proteins according to SCOP domain definitions. We analyzed the ability of UniScore as a comparison metric to verify the structures that are classified from the same family more similar, whereas all other structures pairs are not, by rescoring the TM-align result for UniScore and TMscore. We selected the first protein in the a.1.1.2 Globins family (1a6m__) and ran TM-align one-verses-all the other eight a.1.1.* Globin-like protein and 200 random proteins. In Figure 5, we plotted three components of our UniScore separately of the aligned pairs.



A. CDD alignment

```
      1           12                   29                   46
1bihA sapkyeqkpek vivvkqgqdvTIPCKvt g--lpaPNVVWShnakp lsg-----------gra
1nfdB -trppsvqvak ttpfntrepvMLACYvw gfypaeVTITWRkngkl vmhssahktaqpngdwt
      63          74                   91                   108
1clzH tvtdSGLVIkg vkngdkgYYGCRATneh --gdKYFETLvqvn
1nfdB yqtlSHLALtp s---ygdTYTCVVEhig apepILRDWTpg--
```

B. UniScore alignment (fcar(0)=1)

```
1bihA SAPKYEQKPE- KVIVVKQGQDVTIPCKV T-GLP-APNVVWSHNAK PLSGGRATV--------
1nfdB TRPPSV--QVA KTTPFNTREPVMLACYV WGFYPAEVTITWRKNGK LVMHSSAHKTAQPNGDW
1clzH ---TDSGLVIK GVKNGDKGYYGCRATNE HG--DKYFETLVQVN
1nfdB TYQTLSHLAL- -TPS-YGDTYTCVVEHI GAPEPILRDWTPG--
```

C. TMscore alignment (fcar(0)=0, fcar(1)=0.9828)

```
1bihA SAPKYEQKPEK VIVVKQG---------- QDVTIPCKV-TGLPAPN VVWSH-NAKPLSGGRAT
1nfdB TRPPS--VQV- -------AKTTPFNTRE PVMLACYVWGFYP-AEV TITWRKNGKLVMH----
1clzH VT--------- -------DSGLVIKGVK NGD--KGYYGCRATNEH G-DKYFETLVQVN-
1nfdB --SSAHKTAQP NGDWTYQTLSHLA-LTP S--YGDTYTCVVEHI-G APEPILRDWTP--G
```

Figure 4 Special case: comparison of pure geometric score with UniScore for proteins 1bih A: 307-395 and 1HDM b:88-185. The residues aligned in the reference CDD alignment are marked with purple. (B) and (C) shows the alignment using UniScore and geometric score.

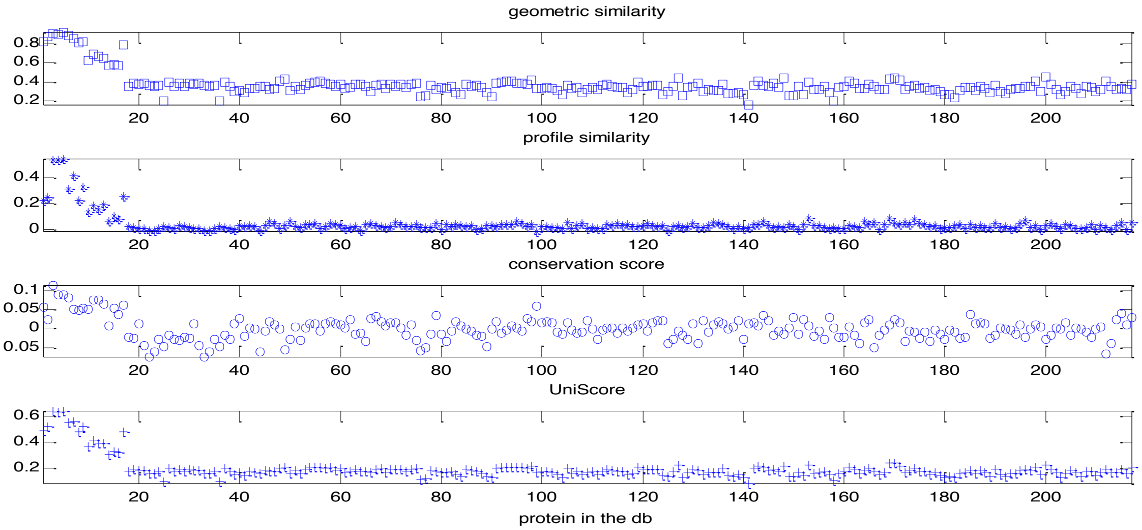


Figure 5 Three components of UniScore of the aligned pairs

Both geometric similarity and profile similarity are able to differentiate the globins family and globin-like proteins from the random proteins; however one case (17th globin-like protein) within the same fold of the query protein is not obvious. We sort the pairwise structural comparisons based on the score of interest, including TMscore and Uniscore, from the best to the worst. The 17th globin-like protein 1b8dA was mistakenly considered as the globin family, no matter what score metric were used, and this actually indicates the possibility of the misalign of TMalign and further indicate the importance to incorporate the evolutionary information into the protein structure alignment. On the other hand, conservation scores is not indicative of whether two proteins are from the same family or not, and this is due to the fact that the conservation score is on the residue pairs level; it prefers two consistent conservation pattern residue being aligned.

2.8 Conclusion

To sum up, while we utilize evolutionary information in structure alignment, we do not abandon other types of information that can help determine residue equivalences. We define **UniScore** as the weighted average of various sources of protein similarity measures, and it can be used in any structural alignment method. The reports of alternatively alignments generated by structural alignment and bad performance resulting from conformational changes, pure geometric similarity is no longer enough to evaluate the similarity between two proteins in the structural alignment. The focus of our study is to generate the most “accurate” structure alignments for a given pair of proteins. We do not claim UniScore to be a uniformly scaled metric with respect to the alignment accuracy. From the sensitivity analysis for each of the UniScore components using a subset of the CDD by selecting a protein pair from each of the 91 CDD families, the performance of UniScore does not appear to be sensitive to the component weights.

CHAPTER 3: UniAlign

3.1 Introduction

The importance of structural alignment and irreplaceable role for study the protein functions have been covered in the first introduction part. Basically, protein structures are usually modeled as rigid 3D coordinates of atoms. And the modeling of aligning two proteins structures can be stated in two ways:

1) The alignment of two protein structures can be modeled as an optimization problem to minimize the distance between two proteins structures after a specific rotation and translation.

2) The alignment of two protein structures is to find the optimal rotation and translation matrix which gives us the largest non-continuous fragments such that after rotation their distance is below a predefined threshold in 3D space [20].

Many automatic pairwise protein structure alignment programs have been developed, mainly varying in different representations, scoring functions, and optimization algorithms [6-11]. For example, DALI [12] uses Monte-Carlo procedure after the initial structural alignment to minimize the intra-structural distance of aligned substructures. CE [7] also uses fragment assembly to build the initial set of equivalences similar with DALI, yet generates the final alignment by gradually adding new eight-residue fragments to the existing alignment. TM-align [8] extends the approaches of STRUCTAL [10] and SAL [11] by using TM-score rotation matrix instead of RMSD rotation matrix and extend the initial guess of equivalent residues by iterative residue-level dynamic programming. FATCAT[9] is a flexible alignment, adopting aligned fragment pairs (AFP) – based dynamic programming and allowing multiple rotational frames resulting from protein flexibility or evolutionary divergence.

Speaking of what constitute a good alignment, we need to consider what leads to protein structural similarity. Structural similarities between different proteins could either result from the evolution from a same ancestor (remote structural homologs) or from convergent evolution (structural analogs). During the evolution of the protein, both the functional sites on the surface of the protein and the hydrophobic core which is essential to maintain the structural integrity are in general remain relative conserved [13]. However, most of the available algorithms align protein structures solely based on 3D geometric similarity, and are limited in their ability to find functionally relevant correspondences between the residues of the proteins, especially for distantly related homologous residues. For example, it is observed that pure geometric information based structure alignment methods are highly sensitive to conformational changes[15].

Previous studies [1, 14] discovered that even structural alignment can benefit from the evolutionary information provided by the alignment of homologous proteins. It is not surprising since information extracted from homologous proteins may represent general features of the family, and allow the prediction of similarity to a remote sequence or family, even when the similarity to each individual aligned sequence is not significant [16]. Two kinds of information can be extracted from the MSA of sets of evolutionarily related sequences: sequence profile and conservation score. The sequence profile specifies a preference for the 20 standard amino acid types at each position in the given MSA. And the residue conservation score of the protein indicates the possible contribution of each residue to maintain the structure and function of that protein.

Considering that protein structural alignment is widely used to identify homologous residues (encoded by the same codon in the genome of a common ancestor) of the proteins compared, structural similarity seems not enough to capture the similarity between amino acid residues. Thus it is essential to properly incorporate protein evolutionary information into structural alignment algorithm. In fact, determining the proper way to integrate evolutionary similarity metrics into the construction of the scoring function of the protein structure program has proved surprisingly difficult. Thus, the goal of our structure alignment model is to recognize the maximal number of evolutionarily important residues as being structurally equivalent with minimal spatial deviations after the optimal rotation and translation.

3.2 Methods

In the previous chapter, we introduced the UniScore, comprising of five sources of protein similarity: geometric, phylogenetic profile, conservation, secondary structure, and sequence similarity. Equally important as the scoring function of a protein structure alignment method is the heuristic search algorithm used to find an alignment with optimal score. The UniAlign algorithm consists of 4 main steps (Algorithm 1). First, an initial alignment is constructed as a set of residues pairs from the two proteins, using a fragment-based search. Second, the proteins are geometrically superposed based on this initial alignment. Third, the spatial proximity of the residues in the superposition, along with the other components of the UniScore are used to collect a new set of residue correspondences. The second and third steps are repeated until the UniScore of the alignment converges. We describe each of these steps in more detail below.

Algorithm 1 UniAlign Algorithm

```
Input:   Two protein structures, A and B of length $L_A$ and $L_B$.
Output:  Aligned residue pairs and rotation/translation matrices.
         Construct the initial alignment (Algorithm 2)
         while pairs have not converged do:
             Geometrically superpose the structures.
             Calculate the UniScore similarity matrix.
             Collect residue pairs using dynamic programming.
         Return aligned residue pairs and the final UniScore.
```

The UniAlign algorithm flowchart is illustrated in Figure 6. Unialign consists of four basic steps: 1). generating an initial set of evolutionarily equivalent residues proteins; 2). calculating the optimal UniScore rotation and translation matrix; 3). extending the initial seed alignment by iterative dynamic programming; 4). calculating the protein similarity score of current alignment.

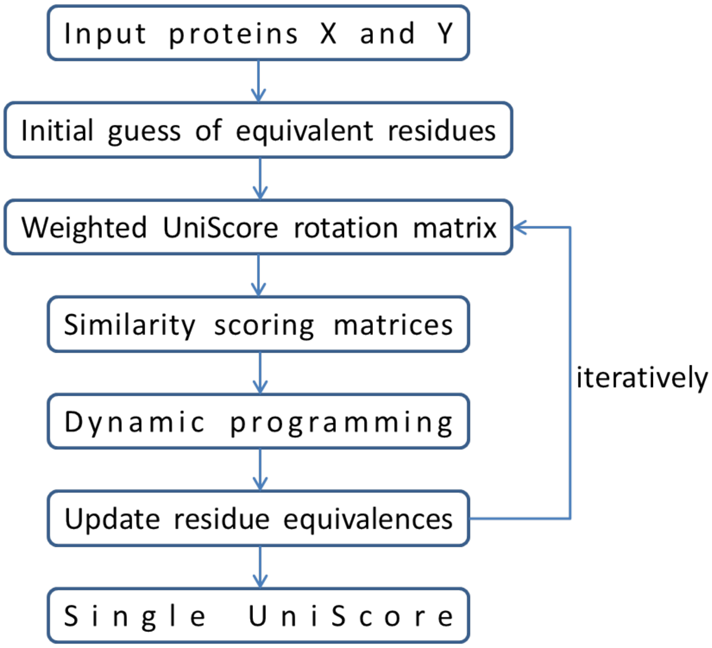

Input proteins X and Y
Initial guess of equivalent residues
Weighted UniScore rotation matrix
Similarity scoring matrices
Dynamic programming
Update residue equivalences
Single UniScore
iteratively

Figure 6 Pipeline of UniAlign

Initial Alignment

Since there is no *a priori* knowledge of equivalent position, the first step in most structural alignment programs is to generate the initial guess of equivalent residues. CE, Mammoth, and Fr-Tm-align uses fragment assembly to build the initial guess of the alignment, while TM-align employs a very simple gapless threading and secondary structure fitting to generate the initial alignment. All structural alignment methods that depend on dynamic programming will suffer from the choice of the initial alignment. In other words, if we just used one alignment (which is far from the correct alignment) as seed to start the structural alignment, then it will doom to a bad alignment. Thus, DP-based structure alignment programs all heuristically searched for the optimal initial alignment, trying various starting points and segment length and selected the one that gives the best score.

Three types of simply initial alignment methods are exploited in UniAlign.

(I) Gapless Threading [8, 11, 36]. The basic idea is shown in Figure 7. Basically, *O(n)* different initial continuous matching segments of size L from the larger protein is chosen (n: the length of the longer protein; L is the length of the smaller protein); and then each segment was extended find the maximum subset of residues below a predefined distance in 3D space. In the end, the rotation and translation matrix that gives the optimal similarity score together with that alignment were selected. Different similarity metrics were used, like RMSD in SAL [11], subset size in MaxSub [36], TM-score in TM-align [8], and we use UniScore in our UniAlign program. Whatever comparison scores are used, they are supposed to differentiate the more accurate alignment in terms of recognizing the largest subset of evolutionarily related structural-equivalent residues.

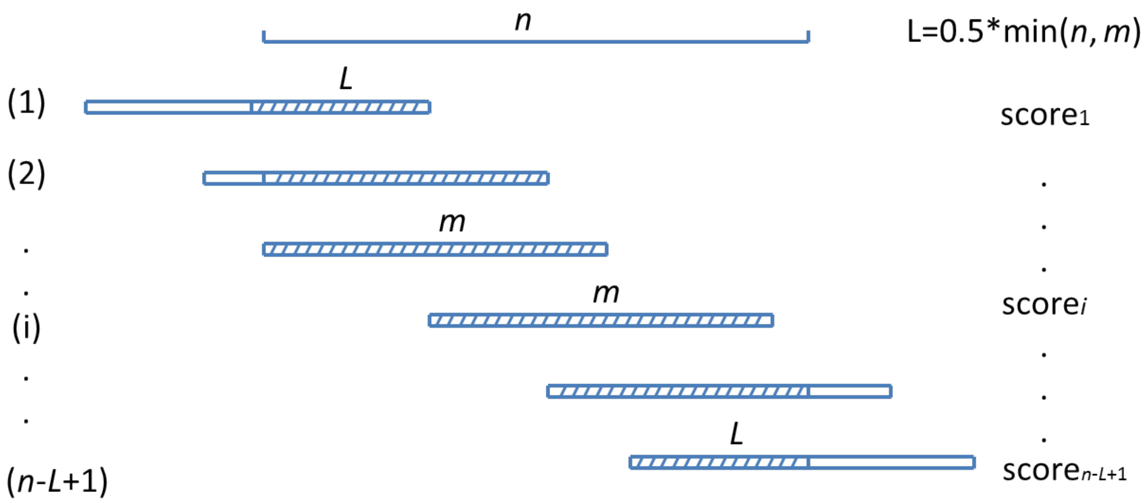


Figure 7 Gapless threading for the initial alignment

The second type of initial alignment was generated using the secondary structure assignments of two proteins by dynamic programming (DP). The score matrix of dynamic programming is composed of either 1 or 0, depending on whether the secondary structure element of the aligned residues is identical or not. The SS state (alpha, beta or coil) of a given residues were assigned based on the C_α coordinates of five neighbor residues, and an optimal gap penalty of -1 for gap-open was used [8].

The third initial alignment method combined the distance-based similarity score matrix which can be calculated from the gapless threading and the secondary structure assignment score matrix mentioned in the second initial alignment. Specifically, the inter-protein distance matrix calculated all the pairwise distances between each atom in the first structure and each atom in the second structure. Then for each entry in the distance-based score matrix, 0.5 was added if the secondary structure of the aligned residue pairs were the same. The third type of initial alignment is then obtained by using dynamic programming with the same gap-open penalty of -1.

We consider structure alignments resulting from gapless alignment of all pairs of fragments of length L_f $\mathrm{L_f}$from the two proteins. Similar to [65], we use L_f =8 if the smaller protein has less than 100 residues, and L_f =12 otherwise. Proteins shorter than 8 residues are used as a single fragment. Unlike TMalign and Fr-TM-align, which use different and possibly conflicting criteria for initializing and optimizing the alignment, we use the same UniScore evaluation for each of these steps. For each fragment pair, we use their alignment to obtain a 3D transformation and use this transformation to superpose the entire proteins (Algorithm 2). The calculation of UniScore and collection of residue pairs is done as in the main algorithm, except without the iterative optimization loop. The UniScore of the alignment is assigned into the corresponding alignment path in T_{init} initial alignment score table. If a residue pair $[i, j]$ is part of alignments resulting from multiple fragment pair alignments, we keep the largest UniScore in $T_{\text{init}}[i, j]$. Once all fragments are assessed, we use dynamic programming with free end gaps, on the T_{init} table to find an alignment path that produces the largest sum of UniScores. This alignment path is used as the initial set of residue correspondences, to be optimized by the rest of the UniAlign algorithm.

Algorithm 2 Generate the initial alignment

```
Input: Two protein structures, A and B of length $L_A$ and $L_B$.
  Out: Aligned residue pairs from the two proteins.
    $T_{\text{init}} \leftarrow L_A$ by $L_B$ initial alignment score table.
    foreach fragment $F_A$ of length $L_f$ from A do:
        foreach fragment $F_B$ of length $L_f$ from B do:
            Geometrically superpose $F_A$ and $F_B$
            Use this 3D transformation to superpose A and B
            Collect residue pairs $[i, j]$ from the UniScore similarity matrix
            Replace all smaller $T_{\text{init}}$ $[i, j]$ values with UniScore of this alignment.
    Return residue pairs from dynamic programming on $T_{\text{init}}$.
```

UniScore-enriched Gaussian-weighted RMSD Superposition

Once we get the initial structurally equivalent residues, the next task was to superpose the two substructures in 3D space and the standard RMSD fit is the most widely used method to calculate the rotation matrix which was previously described by Kabsch. The RMSD rotation matrix is the optimal transformation matrix with the minimal positional deviation. However, the RMSD fit is sensitive to outliers, and the result can be skewed by the flexible regions such as loops and hinged domains. In order to overcome this problem, many RMSD-based methods have been proposed, namely distance-cutoff method [66], number-cutoff method and position-weighted method [40]. For example, global distance test (GDT) algorithm identified multiple maximum subsets associated with different threshold cutoffs. Alternative solution is the position-weighted method, by assigning each superimposed position a single weight and iteratively updates the weights until the optimal solution is found. Current position-weighted methods calculate different weights directly based on the distance between two superimposed atoms, in form of Gaussian kernel [40] or simply the inverse form of the average distance.

Standard RMSD fit

Given two protein structures X and Y, each have n atoms. The centers of mass of both proteins are translated to the origin. The RMSD fit problem is to find an orthogonal rotation matrix R by minimizing the following root mean square deviation:

$$\mathrm{RMSD} = \frac{1}{n}\sum_{n}\sum_{i=1}^{3}(X_{n_i} - Y'_{n_i})^2 \tag{24}$$

where Y' is the new structure after rotation, n is the number of atoms.

There are four steps in standard-RMSD (sRMSD) optimization problem:

1. Calculate a 3×3 covariance matrix $C = Y^T X$.

2. Implement the SVD (singular value decomposition) of the covariance matrix: $C = UIV^T$, where U and V are the left and right singular vectors matrices respectively, and I is the positive semidefinite diagonal matrix singular values of R.

3. Update I= $\text{sign}(\det(R))$.

4. Calculate the rotation matrix $R = UIV^T$.

Even though TM-align claimed that one of their contribution was using the TM-score rotation matrix for their superimposing (rotation matrix maximize the TM-score after superposition the aligned residues), instead of the RMSD rotation matrix. In fact, what they used to get their rotation and translation matrix was still the standard RMSD, and just used the TM-score as the comparison metric to select the optimal solution. The issue of the flexible regions dominating the sRMSD fit problem is still there. The way TM-align avoiding this problem is heuristically search with different of fragment given an aligned pairs, and select the one with the optimal TM-score.

Weighted RMSD fit

In order to remove the outlier effects of sRMSD, we implemented the refined Gaussian-weighted RMSD algorithm in UniAlign, to bound the influence of flexible region residues through iteration. The Gaussian-weighted RMSD method first performed a sRMSD fit to bring the two structures into proximity and then conducted iterative wRMSD fit until convergence. Basically, wRMSD assigned residue pairs in close proximity relatively greater weight than those pairs further apart [40]. The optimal solution in Gaussian-weighted RMSD was the one maximized the sum of all weights (%SUM). The structural alignment method based on Gaussian-weighted RMSD performs pretty well to superpose two conformations or two homogenous proteins, with small and large displacements [67]. However, for two significantly different structures (low sequence identity$\leq 20\%$), they yield poor results, which is the practical limits of their method. This result was not surprising since it was inherent in the way they calculated the weights. When two distantly related protein structures were firstly aligned by sRMSD fit, their optional deviation after superposition would be very high, which in turn made the Gaussian weights very low. This equals to only few atoms would be included when calculating the weighted rotation matrix. Thus we should be very cautious to extending the Gaussian-weight RMSD into a structural alignment program, since the structural alignment are supposed to be able to find the evolutionarily related residues that are not evident from sequence alignment alone. If a protein structure alignment method can only homologous proteins, then no matter how good it performance it, its application will be very limited.

In UniAlign, we still adopted the same form of weights as to Gaussian weight factor:

$$w_n = e^{-(d_n)^2/c} \tag{24}$$

where, c is the scaling factor equaling to the standard RMSD value [67]; and d_n is the positional deviation after superposition.

What is not clearly mentioned in the Gaussian-weighted RMSD paper is the weighted center of mass of each protein at the origin:

$$wCM_x = \frac{\sum_n w_n x_n}{\sum_n w_n} \tag{25}$$

Similar to the idea of sRMSD, weighted RMSD fit tried to minimize the weighted sum of distance between two aligned residues in the following equation:

$$wRMSD = \frac{1}{n}\sum_n w_n \cdot \sum_{i=1}^{3}(X_{n_i} - Y'_{n_i})^2 \tag{26}$$

And the solution to this weighted optimization problem is to simply incorporate the weights term into the covariance matrix.

$$R = \mathrm{repmat}(w, 3,1).* Y' * X \tag{27}$$

Unlike sRMSD, there are multiple solutions to wRMSD, and thus a proper convergence metric is very important. Theoretically, the best superposition of two homologous structures should have the largest number of evolutionarily-related residues in close proximity, even if it is at the expense of a large deviation for those residues in the flexible domain. Thus UniAlign uses the UniScore to select out the optimal solution, and thus the rotation matrix is called: UniScore-enriched Gaussian-weighted RMSD superposition.

We observe that the original Gaussian-weighted RMSD [40] tends to yield poor results for significantly different structures (low sequence identity ≤20%). Several actions were taken in UniAlign to refine the performance of Gaussian-weighted RMSD. First, in order to avoid over-fitting to very few pairs, we resort to the standard RMSD if there are less than 10 pairs of residues aligned closer than *sqrt*(*RMSD*). Second, the UniScore of the aligned residue pairs is used as the convergence criteria during iterative superposition, ensuring the superposition step improves the same criteria as the rest of UniAlign algorithm. Third, whereas the original method uses standard RMSD superposition in its first step, we use the geometric superposition available from the previous iteration (of the iterative optimization in Algorithm 1) to speed up convergence and ensure a smooth exploration of the search space.

Figure 8 provides a thorough illustration of how weighted-RMSD outperforms standard-RMSD. This example is about homologs from the SpoU rRNA methylase family with 26% sequence identity (1ipa A:1-263 and 1gz0 A:2-243). For the initial alignment, we used global sequence alignment using default parameters (BLOSUM62), and the resulting standard and weighted superposition are shown in Figure 9 (A) and (B). We also provided the Gaussian form weight that is used for this weighted superposition (C).

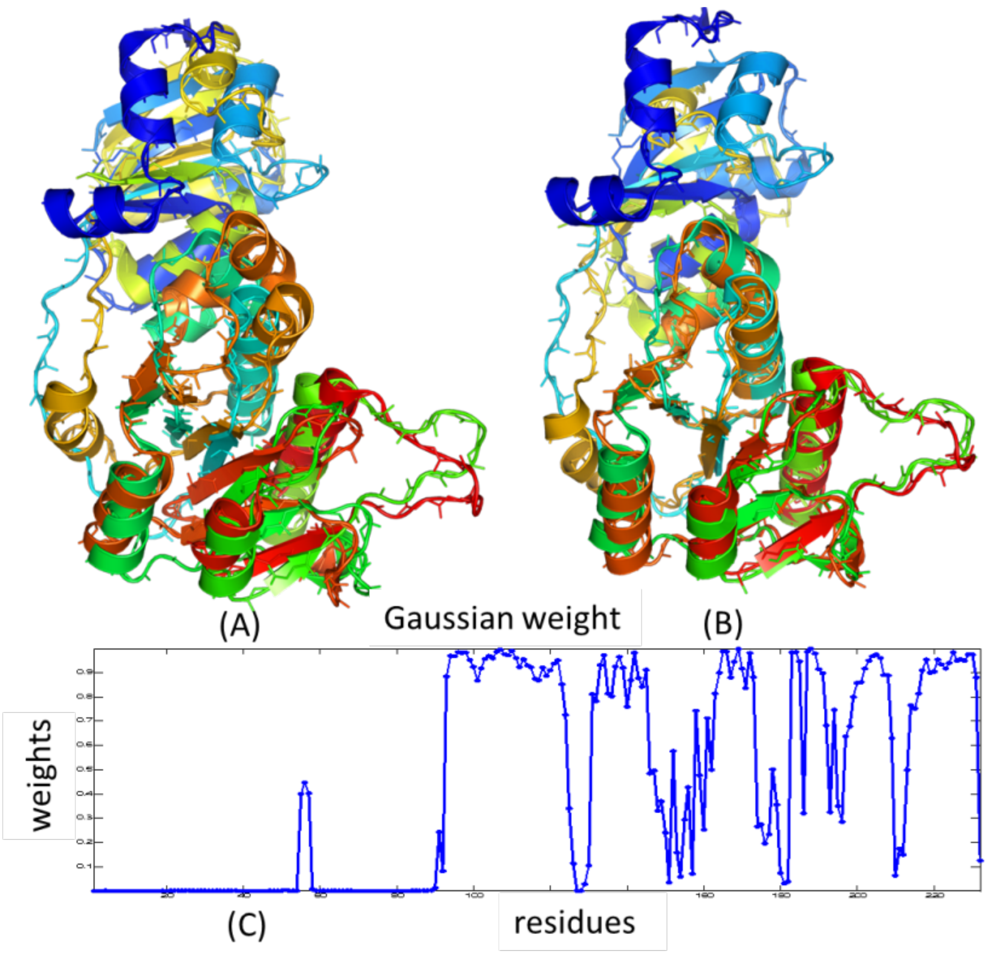


Figure 8 SpoU rRNA methylase family (26% ID). (A) Standard superposition of 1gz0 onto 1ipa. (B) weighted superposition obtained from the same initial sequence alignment. (C) Gaussian weights.

As we can see, although the standard RMSD fit minimized the sum of distance of entire atom pairs, it cannot guarantee the small residuals to the majority of atom pairs. In fact, the RMST fit is sort of the minimization in the sense of average. In addition, the Gaussian-form weights inherently selected out residue pairs with similar relative positions between two structures, while discounting loops and flexible regions. What can be observed is that the residue pairings in regions of good structural agreement will be heavily weighted in the wRMSD calculation and drive the superposition. There are two domains in the query 1ipa protein, and the bottom half is the alpha/beta knot fold. Compared to the standard superposition, the weighted superposition overrides all the ambiguities, and correctly pairs the β-sheet residues. Also, the calculated weights are directly related to the distance between two atoms in space. Consequently, atom pairs in close proximity have a greater weighting than those further apart, biasing the superposition toward the regions that remain relatively rigid between conformations.

Dynamic Programming

In this section, we will talk about collecting residue pairs from the geometric superposition. Geometric superposition, while bringing some of the residue pairs close to each other in space, may make or break other residue pairs. Thus, we collect new residue pairs that are consistent with the geometric superposition, along with the rest of the components measured by UniScore. We consider a score table T_{align} where each entry is the individual UniScores of the pairs of residues from the two proteins. Residues that have similar evolutionary, sequence, and secondary structure features, as well as that are close in space, will have high UniScore values. We use dynamic programming (with affine gap penalty) on T_{align} to collect a new set of correspondences, such that the sum of their UniScores is optimal. These new correspondences are then used for another round of superposition and residue collection; iteratively optimizing the alignment until the total UniScore cannot be improved further.

Starting with the initial guess of the equivalent residues, an iterative dynamic programming algorithm, which has been extensively used in many structure alignment methods such as CE, SAL, MaxSub and TM-align, was applied to generate and refine new equivalences that maximize our scoring function UniScore. This heuristic iterative dynamic programming algorithm consists of four steps. Firstly, a specific type of rotation and translation matrix was generated based on the initially aligned residue pairs using either standard RMSD or weighted RMSD fit. Secondly, the inter-protein distance matrix was generated by superposing one whole structure by the rotation matrix to another, in which all pairwise distances between every atom in the first structure and every atom in the second structure was calculated. Thirdly, the distance matrix is converted into the similarity score matrix. Fourthly, a new alignment was produced by a free end-gaps semi-global alignment. Unless two proteins to be compared are known to be single domains, it makes more sense to not penalize the end gaps. In our implementation we modified the Needleman-Wunsch algorithm [68]: we set the gap rows to zero to ignore the start gaps; and started the traceback with the maximum score at the end of either sequence to ignore the end gaps. As for the internal gaps, the affine gap penalty method was used: $g(k) = gap\_open + gapextend(k)$, where k is the number of gaps [69]. The empirically optimal gap-open penalties are -0.6 and 0.0.

Every time we obtained a newer alignment, we re-calculated the rotation matrix and then re-implemented the DP to the new score matrix. The iterative searching was repeated until the alignment becomes stable and then the alignment with the highest UniScore was returned. The scoring matrix used for DP was derived from the weighted-UniScore rotation matrix, which should result in faster convergence and more biologically meaningful structural alignment.

3.3 Parameter Optimization

The weights in UniScore controlling the contributions of different features and the gap penalties used in dynamic programming are optimized using grid search, on a small subset of the CDD, with the objective of maximizing the fraction of correctly aligned residues. The training dataset and the optimized parameters can be found in the supplementary file. For each of the UniScore similarity components, we calculate it for each of the aligned residue pairs in CDD pairwise alignments and collect the component score from all residues in all alignments. The mean and standard deviation for each component is calculated from this collected reference set of scores. Before calculation of UniScore, the score from each component is normalized, using these statistics, to zero mean and unit standard deviation. This normalization is necessary to ensure different components are on the same scale and that adjusting their weights in UniScore is more meaningful. The table below shows the mean and standard deviation of each of the UniScore components, as calculated from reference CDD alignments and the weights of these components used in UniScore. In the dynamic programming used in UniAlign, an affine gap model was used, with the gap-open and gap-extend penalties of -0.82 and -0.0025, respectively. The component weights and the gap penalties were determined empirically from the training dataset by a grid search parameter optimization, optimizing for the fcar0 achieved on the training dataset (Table 4).

Table 4 The mean and standard deviation for the UniScore components, optimized from the training dataset.

abbreviation	**description**	**mean**	**std. dev.**	**weight**
geo	Geometric component	0.0908	0.1088	0.290
pro	Sequence profile component	0.1379	1.2151	0.345
con	Conservation similarity component	-0.037	0.2925	0.200
sse	Secondary structure similarity component	0.0097	0.1625	0.070
seq	Sequence similarity component	-0.9517	2.1125	0.075

Weight Sensitivity Analysis

For the optimized parameters of UniScore, which we will talk about more in specific aim 2, it is important to make sure that the optimized weights are robust. Thus, we did a weight sensitivity analysis experiment.

We randomly selected a protein pair from each of the 91 protein families in the CDD benchmark. This dataset is independent from the training dataset, and is representative of the CDD families. The protein pairs used in this analysis were shown in Appendix A.

For each UniScore component, we varied the weight from 0 to 1 and recorded the average accuracy on the representative dataset. The results are show in Figure 9 below. The change in UniAlign's performance for small increments of these weights is small. Except for the SSE component, the variation in UniAlign's performance is "smooth". We attribute the non-continuous variations in the SSE component to the small size of the representative dataset we used. Note that in SSE, even though the fcar measure appears to fluctuate, this fluctuation stays within 1% in fcar measure.

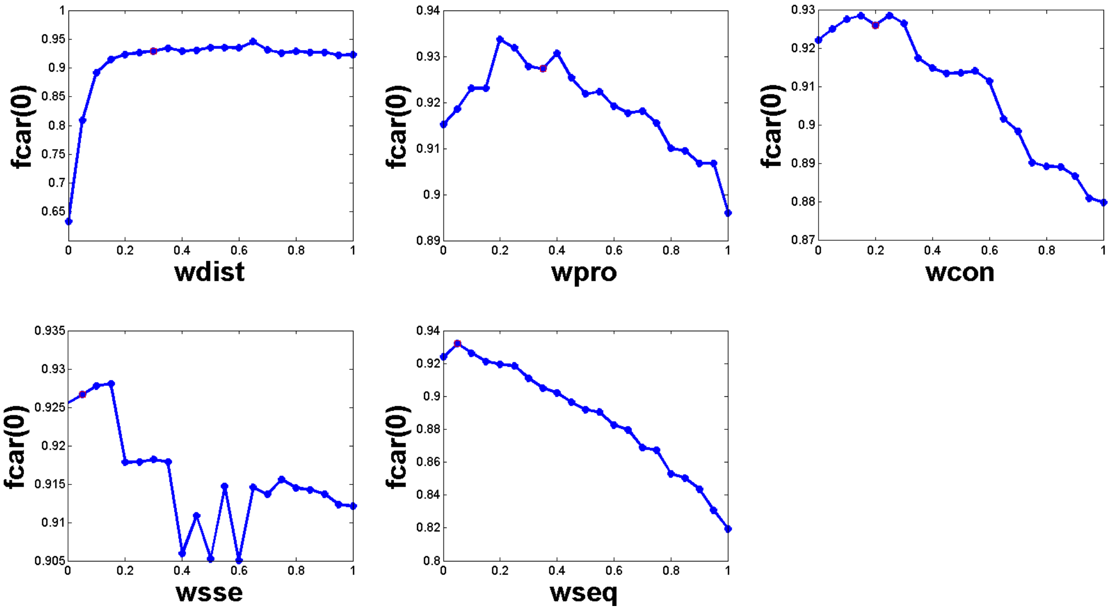


Figure 9 Weight sensitivity analysis on a representative subset of the CDD benchmark database.

3.4 Experiments

Benchmark Datasets

We evaluated UniAlign on three large-scale datasets that are commonly used to assess sequence and structure alignment methods: CDD, HOMSTRAD, and BAliBASE. The subset of NCBI's human-curated Conserved Domain Database (CDD) [70] used in [14] contained a total of 3591 pairwise alignments with the corresponding ASTRAL SCOP domains [71]. HOMSTRAD [72] is a curated database of structure-based alignments for 3454 homologous structures from 1032 protein families; giving a total of 9536 pairwise alignments of protein structures. BAliBASE [73] contains 162 multiple alignments, involving 1944 pairwise sequence alignments from five different reference sets indicating various divergence levels.

A global sequence alignment with the PDB sequences was used to identify start and end positions of the sequences listed in these benchmark datasets and to correct for discrepancies such as missing residues. Unlike other assessment studies that use pre-segmented domains, we use the full length chains as the input structures. We denote the alignment generated by different structure alignment methods as the test alignment. Only the homologous regions marked in the reference datasets were used to evaluate the test alignments.

Comparison with Other Methods

Three widely used structure alignment methods were chosen for comparison: DaliLite, TMalign, and Deepalign. DaliLite is a classical geometry-based alignment method that uses a Monte-Carlo procedure to minimize the intra-structural distances of aligned substructures and generates the final alignment by gradually adding new eight-residue fragments to the existing alignments [12]. TM-align is another method that has been shown to perform well as a purely geometric information based structure alignment method [8]. TM-align utilizes the TMscore, a length-normalized geometric scoring measure that, compared to RMSD, attenuates the contribution of large distances. Deepalign is a recently developed method that integrates the BLOSUM mutation matrix, local substructure mutation matrices, and hydrogen-bonding similarity in its scoring function [43]. We note that the so-called evolutionary distance in Deepalign is only a simple transformation of the BLOSUM mutation matrix and does not utilize the evolutionary history of the proteins being aligned.

Accuracy of the Alignment

As we have mentioned before, a good structural alignment should find the maximal number of structurally equivalent residues which are also evolutionarily related. Generally, three tasks need to be accomplished when evaluating structural alignment methods: the accuracy of the alignment in terms of the sequence alignments it produces; the quality of the alignment in terms of the geometric proximity in 3D space, either using every method's own scores used to optimize the alignment or rescoring the generated alignments with another scoring function, e.g. root mean square distance based scores [5]; and the capability of the scoring function to discriminate homologous proteins from randomly related proteins in a database-wide comparison. The first measure depends on a manually-curated reference alignment, for example, HOMSTRAD [72], CDD [14, 74], Sisyphus[75]; and the third metric needs a standard of truth to estimate the rates of true and false positives with receiver operating characteristic (ROC) curves with either SCOP [76] or CATH [77] or both [78] classification. Only the second evaluation method is reference-independent, yet finding a universally acknowledged, objective measure is not easy.

In our study, we used the fraction of correctly aligned residues of all aligned residues as the evaluation metric, to measure the deviation of a structure-based sequence alignment from the correct manually curated alignment [4, 14, 79]. The residue pair is defined as correctly aligned in the sequence alignment generated by structural alignment if the same pair is also aligned in the corresponding reference alignment. The reference alignments were extracted from three gold-standard benchmark database: CDD.

In CDD, the unaligned residues in the reference alignment refers to those residues either without crystal structure or the ASTRAL domain spans less than the whole reference aligned span, and are not considered for shift error further.

There are four steps to calculate the fraction of corrected aligned residues (f_{car}) (for both the reference and test alignments: (Figure 10)

1) Assign the serial numbers to both sequences, in both alignments.

2) Extract the aligned regions in the reference alignment (remove insertion or deletion).

3) Calculate the shift error vector for both sequences. Take the first sequence as an example, the corresponding aligned regions from the second sequence were extracted in both the reference alignment and test alignment. After that, the shift error vector of the first sequence was generated: a. whenever there is a gap, -1 is assigned at the position in the shift error vector; b. otherwise, the absolute value of the difference of the serial numbers of the two residues that it aligned in the second sequence. Thus a -1 in the shift error vector means insertion or deletion while a 0 means correctly aligned residues.

4) Calculate the fraction of correctly aligned residues, by the ratio of the number of residues correctly aligned in the test alignment, divided by the total number of aligned residues in the reference alignment: $f_{car}(\delta) = \frac{m(\delta)}{2r}$, where $m(\delta)$ is the number of aligned residues in both sequences with shift error up to δ, r is the length of the aligned regions in the reference alignments. When$f_{car}(\delta = 0)$, it is also called the sensitivity of sequence alignment. In terms of homology modeling, we want the structural-derived sequence alignment to be both as accurate as possible and include the maximum number of

residues from the reference alignment. In addition to f_{car}, we also defined the relative alignment length as $l = \frac{t}{r}$, where t is the number of aligned residues in the shift error vector.

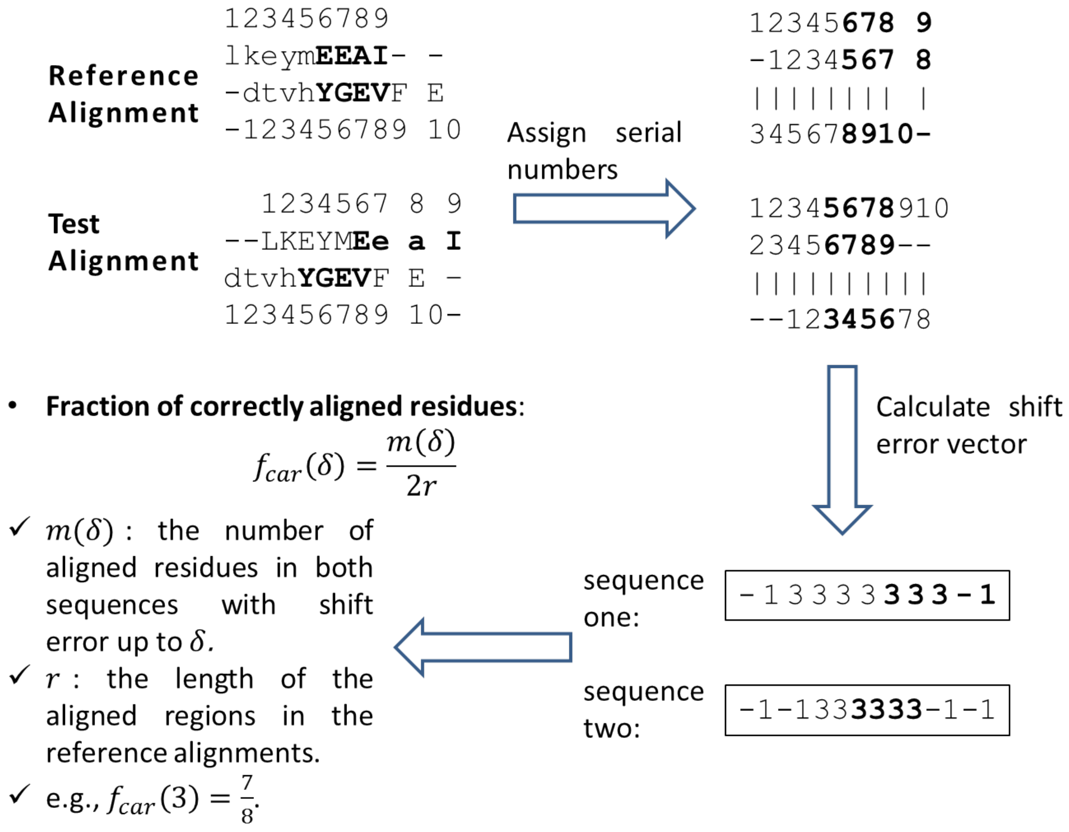

Reference Alignment
123456789
lkeymEEAI- -
-dtvhYGEVF E
-123456789 10
Test Alignment
1234567 8 9
--LKEYMEe a I
dtvhYGEVF E -
123456789 10-
Assign serial numbers
12345678 9
-1234567 8
|||||||| |
345678910-
12345678910
23456789--
||||||||||
--12345678
Calculate shift error vector
sequence one:
- 1 3 3 3 3 3 3 3 - 1
sequence two:
-1-1333333-1-1
Fraction of correctly aligned residues:
$f_{car}(\delta) = \frac{m(\delta)}{2r}$
✓ $m(\delta)$: the number of aligned residues in both sequences with shift error up to δ.
✓ r : the length of the aligned regions in the reference alignments.
✓ e.g., $f_{car}(3) = \frac{7}{8}$.

Figure 10 Explicit example illustrating the calculation of $\boldsymbol{f}_{\mathbf{car}}$.

3.5 Results

Performance on Benchmarks

The results from running UniAlign and other structure alignment methods on the three benchmark datasets are summarized in Table 5.

Table 5 Comparison of the performance of four structure alignment methods on three benchmark datasets. For each dataset, the best performance values are shown in bold. The scores of the reference database alignments are also shown in bold, when it is better than the best value from the structure alignment methods.

Method	$fcar_0$	UNI	GEO	SSE	SEQ	PRO	CON
CDD core regions (3591 pairs with sequence identity 21.7%)							
UniAlign	**93.7%**	**2.09**	0.682	0.141	**0.282**	**0.163**	**0.054**
Deepalign	91.5%	1.99	0.654	**0.151**	0.265	0.123	0.045
DaliLite	92.1%	2.00	0.662	0.141	0.095	0.041	0.041
TMalign	85.1%	2.05	**0.684**	0.143	0.047	-0.002	0.038
CDD core	**100.0%**	1.09	0.341	**0.232**	**0.688**	**0.566**	**0.071**
HOMSTRAD (9536 pairs with sequence identity 35.7%)							
UniAlign	**91.4%**	**2.74**	**0.802**	0.134	**1.696**	**1.236**	**0.168**
Deepalign	90.3%	2.69	0.789	**0.136**	1.685	1.215	0.163
DaliLite	83.1%	2.68	0.797	0.134	1.526	1.108	0.158
TMalign	87.0%	2.68	0.793	0.134	1.559	1.135	0.159
HOMSTRAD	**100.0%**	2.67	0.762	**0.135**	1.658	1.210	0.164
BAliBASE (1944 pairs with sequence identity 23.4%)							
UniAlign	**73.5%**	**2.36**	**0.733**	0.121	**0.504**	**0.626**	**0.114**
Deepalign	71.6%	2.26	0.706	**0.126**	0.487	0.585	0.104
DaliLite	68.9%	2.26	0.712	0.119	0.283	0.478	0.097
TMalign	68.3%	2.30	0.729	0.120	0.275	0.461	0.096
BAliBASE	**100.0%**	2.00	0.601	**0.126**	0.414	0.558	0.101

For all three datasets, UniAlign aligned a higher fraction of the residues correctly, achieving *fcar*$_0$ of 93.7%, 91.4%, and 73.5% for CDD, HOMSTRAD, and BAliBASE datasets, respectively. Since UniAlign optimizes for the UniScore, it is no surprise that it generates alignments with the best UniScore. Geometric quality of the UniAlign alignments, as measured by TMscore, was also comparable to or better than those generated by other alignments. This indicates that incorporating evolutionary information did not deteriorate the performance of UniAlign as a structure alignment method.

Secondary structure states of the aligned residues were best matched by Deepalign, likely due to the consideration of hydrogen bonding in its scoring function. Residue pairs in the UniAlign alignments had the highest scores for their sequence, profile, and conservation scores. Note that Deepalign utilizes sequence information, whereas DaliLite and TMalign make use of only the geometric information. Consequently, the sequence and evolutionary scores of the alignments from DaliLite and TMalign are significantly lower than those of UniAlign and Deepalign.

BAliBASE reference alignments were more difficult to reproduce by the structure alignments, even though these sequences were not more remotely related to each other than those in the other databases, as measured by sequence identity. CDD database contained alignments with poorer geometric similarity than the other databases, as measured by TMscore. On the other hand, CDD had a higher secondary structure score, indicating that its human curators may have paid special attention to the secondary structure elements when constructing the alignments and determining the core regions.

Compared to the other databases, HOMSTRAD alignments contained proteins that were more similar to each other in both sequence and structure.

Note that DaliLite failed to report any result for 14 pairs from CDD, 646 pairs from HOMSTRAD, and 51 pairs from BAliBASE. These missing alignments were excluded when calculating the average scores, inflating the reported statistics for DaliLite. Deepalign failed to produce an alignment for one CDD pair. UniAlign and TMalign generated an alignment in all cases.

Whereas $fcar_0$ evaluates the exact agreement of the residue correspondences with respect to the reference alignment, it is suggested that an approximate alignment that superposes the correct regions of the proteins may be sufficient in certain applications, such as fold recognition. Figure 11 shows the accuracy of the CDD alignments under different allowed shift errors δ.

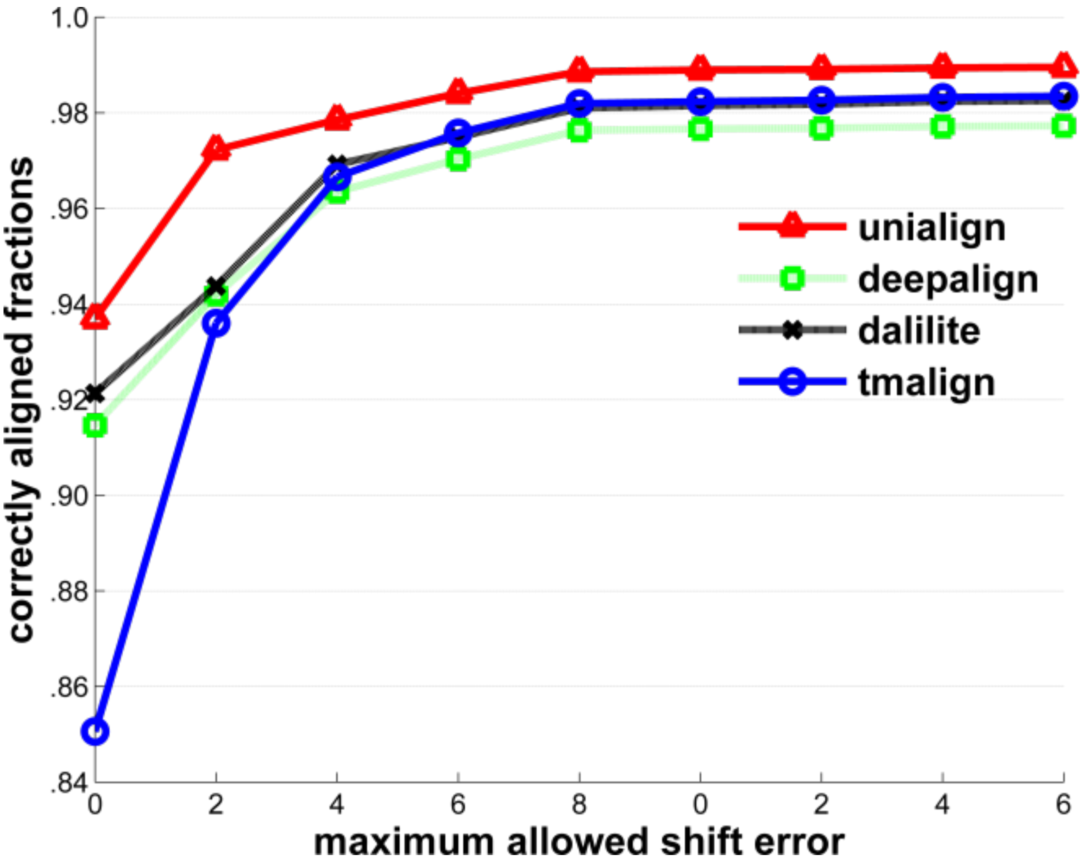


Figure 11 Average *fcar* of CDD alignments of different methods as a function of the shift error tolerance level δ. The y-axis starts from 0.84 to 1.0.

UniAlign outperforms other methods for all shift error tolerance levels. The accuracy of UniAlign, Deepalign and DaliLite increased by 2-3% when a single shift error was allowed, whereas the accuracy of TM-align increased by 8% for δ=1. This suggests a substantial room for improvement of existing TM-align alignments by considering single-residue shifts. Whereas an additional 3% of the residues from Deepalign, DaliLite, and TM-align had a shift error of δ=2, UniAlign alignments contained fewer residues with two-residue shift error. The fraction of residues with a shift-error of 3-8 were small for all methods, indicating a deficiency in generation of initial alignments, such that the remaining cases misaligned by each method cannot be corrected by small adjustments of their existing alignments.

Dependence of Performance on Sequence and Structure Similarity

A successful structure alignment method should be able to generate accurate alignments for different types of proteins it is applied to. Here, we characterize the performance of UniAlign and other structure alignment methods with respect to the level of sequence and structure similarity levels of the aligned proteins.

Figure 12 illustrates the accuracy of aligning proteins with different sequence similarity levels, where similarity is measured as the fraction of identical amino acid residues in the reference alignment. UniAlign is robust with respect to the homology level of the proteins it is applied to and consistently produces good alignments at all sequence identity levels. The other methods perform poorer on more remotely homologous proteins, consistent with the commonly accepted notion that closely related proteins are easier to align. Surprisingly however, the performance of other methods decreases for proteins with 45+% sequence identity compared to those with 40-45% identity. We attribute this to the presence of equally good alignments when only geometric similarity is considered -- UniAlign is able to distinguish among these alternatives by utilizing additional non-geometric information. TMalign was the most sensitive to the sequence similarity level of the proteins and performed its best when the proteins were 35-40% identical.

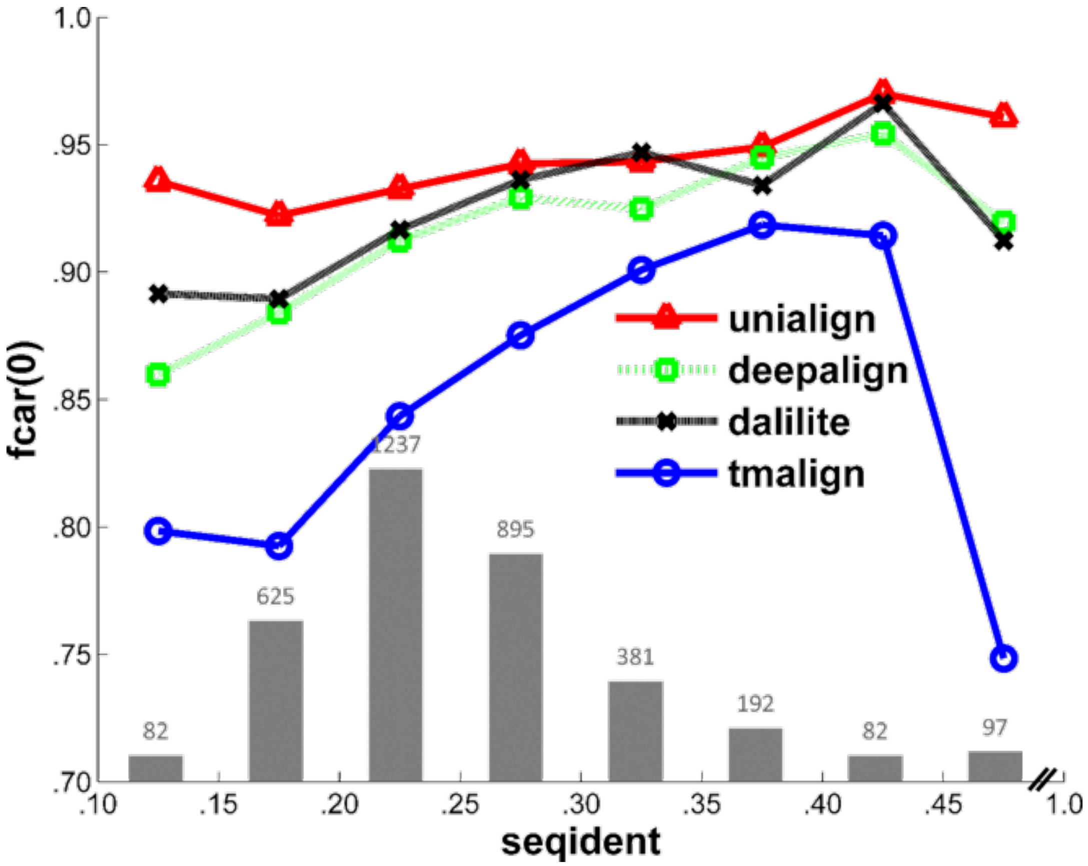


Figure 12 Dependence of alignment accuracy on the level of homology of the proteins from the CDD dataset. Alignments were grouped into sequence identity bins of 5% width. Line plots show the average *fcar*$_0$ values of various methods, whereas the histogram shows the number of alignments in each bin.

In order to characterize the geometric similarity of the dataset proteins, we used the TMscore measure of the superposition produced from the residue correspondences of the reference alignments (using the entire CDD alignments, not just the core regions). For structurally highly similar proteins (TMscore>0.5), the performance of all the methods were similar (See Figure 13).

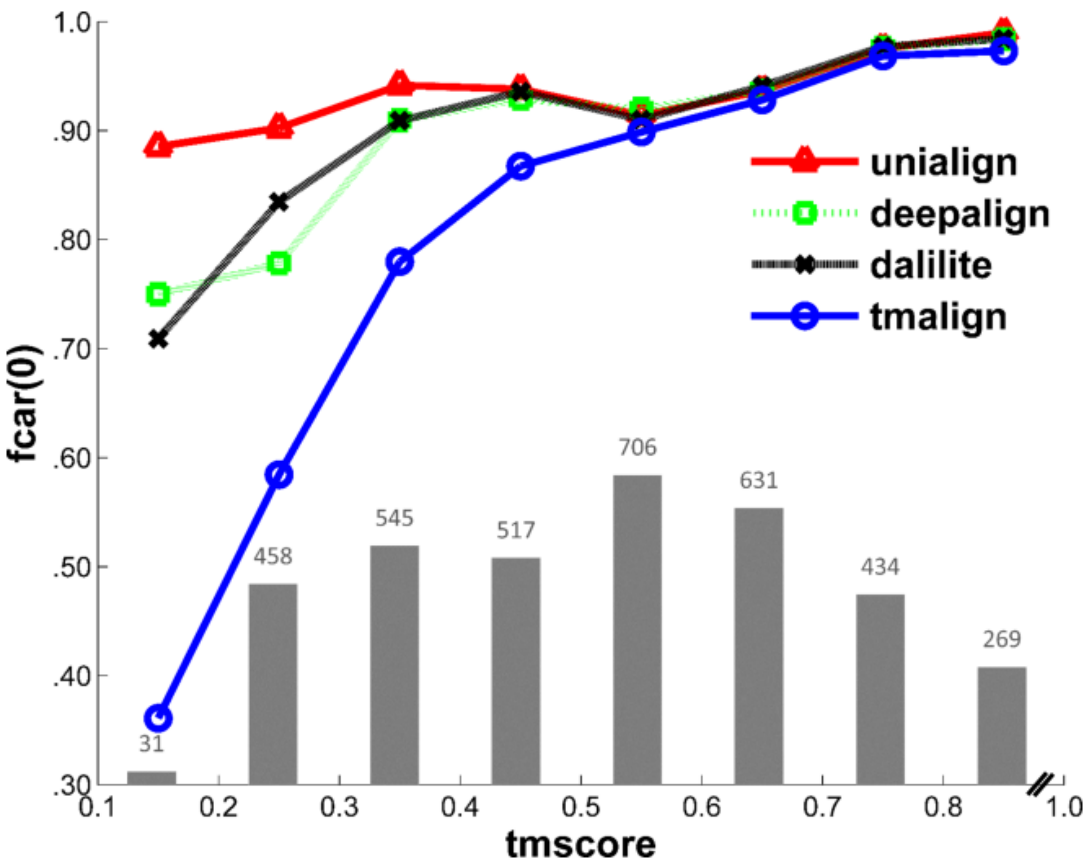


Figure 13 Dependence of alignment accuracy on the level of structural similarity of the proteins from the CDD dataset. Structural similarity is measured by the TMscore of the superposition generated from the reference alignments. Proteins are grouped into structural similarity bins of size 0.1. Line plots show the average *fcar*$_0$ values of different methods, whereas the histogram shows the number of alignments in each bin.

At lower structural similarity levels Deepalign, DaliLite, and TM-align produced significantly less accurate alignments. On the other hand, UniAlign was robust with respect to structural similarity level of the proteins it was applied to, consistently producing highly accurate alignments. We attribute this to the fact that at lower structural similarity levels, geometric information alone is not sufficient for identifying functionally equivalent residues and there is a greater benefit from incorporating evolutionary information. Note that although Deepalign utilizes sequence information, its behavior and performance at different sequence and structure similarity levels were similar to those of DaliLite, which uses geometric information alone.

Case Study

We demonstrate the advantage UniAlign has over other structure alignment methods using a case study of proteins from the immunoglobulin superfamily: a heterogeneous group of proteins built on a common fold comprised of a sandwich of two beta sheets, listed in CDD with the identifier CD00096. The residue correspondences of the reference alignment and of the test alignments from structure alignment methods are shown in Figure 14a. Here, the accuracy of a test alignment is determined by the agreement of the core residue correspondences with those in the reference alignment (shown with capital letters).

UniAlign aligns all of the core residues correctly, whereas TM-align produces one-residue shifted alignments and DaliLite and Deepalign produces two-residue shifted alignments. From the geometric similarity point of view, all of these shifted alignments are as good as the reference alignment, yet they misalign functionally equivalent residues, including the highly conserved cysteine bridge and tryptophan residues. This demonstrates that geometric information alone is insufficient in recognizing biologically relevant alignments and additional sequence and evolutionary features need to be considered in order to obtain accurate alignments.

Figure 14b and Figure 14c show the geometric superposition of the two proteins resulting from DaliLite and UniAlign. UniAlign aligns all of the beta strands correctly, whereas

DaliLite misaligns them as can be observed by focusing on the ends of these beta strands. The regularity of the beta strand elements in general is an important factor for DaliLite (and other pure geometry based structure alignment methods) to produce such incorrect alignments of residues that otherwise superpose tightly in 3D space.

Family-Specific Optimization

There were several reference alignments for which none of the structure alignment methods (including UniAlign) was able to produce the correct alignment. Among these were alignments of the proteins from the calmodulin-like (CBP) protein family in HOMSTRAD database. The CBP family contained 8 all-alpha protein structures, with an average pairwise sequence identity of 38%. Calmodulin has a flexible linker connecting two globular calcium-binding domains, which throws a wrench into the structure alignments, because of the difficulty of simultaneously superposing the two domains with a rigid alignment. The accuracy of the alignments obtained by Deepalign, TM-align, DaliLite, and UniAlign were 45.8%, 56.6%, 65.9%, and 66.2%, respectively. Although flexible structure alignment may be the natural solution to align these proteins, Fr-TM-align [65], which is popular for its support of flexible alignments, also failed to align these globular domains, with $fcar_0 = 41.49\%$.

(a) Sequence Alignments

A. CDD alignment (*fcar*$_0$=100%, TMscore=0.269)

```
      1           12                29                46
1clzH tttapsvyplv pgcsdtsgssvTLGCLv kgyfpepVTVKWNygal ssgvrtvs---------
1nfdB ---dsgvvqsp rhiikekggrsVLTCIp is---ghSNVVWYqqtl gkelkfliqhyekverd
      63          74                91                108
1clzH ------svlqs gfyslsSLVTVpsst-w psqTVICNVAhpaskte LIK-RIEpr---
1nfdB kgflpsrfsvq qfddyhSEMNMsalele dsaMYFCASSlrwgdeq YFGpGTRltvle
```

B. UniAlign alignment (*fcar*$_0$=100%, TMscore=0.560)

```
      20          32                41                58
1clzH ssvTLGCLvkg yfpepVTVKWN------ ---yg------------ als-s--gvrtvssvlq
1nfdB grsVLTCIpis g---hSNVVWYqqtlgk elkfliqhyekverdkg fl-psrfsvqqfdd---
      75          86                103               120
1clzH sgfyslsSLVT vp-sstwp--sqTVICN VAhpa-sktelIK-RIE pr-----
1nfdB -----yhSEMN msalel--edsaMYFCA SSlr-wgdeqYFGpGTR --ltvle
```

C. TMalign alignment (*fcar*$_0$=0%, TMscore=0.540)

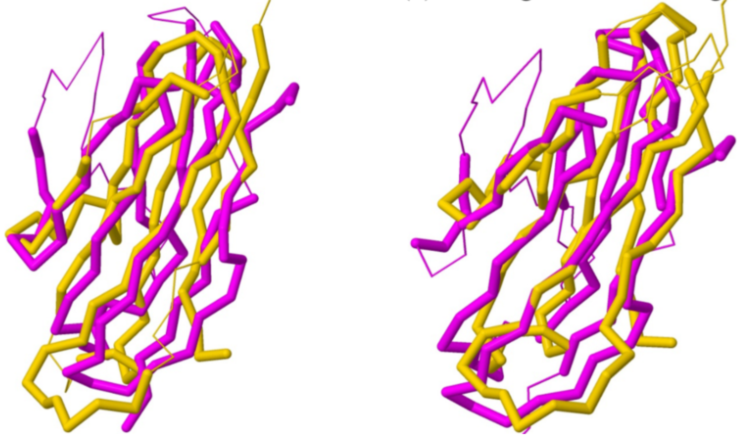
```
1clzH tttapsvypl- vpgcsdtsgssvTLGCL vkgyfpepVTVKWN--- ---yga-----------
1nfdB --dsgvvqspr hii-k--ekggrsVLTC IpisghSNVVWYqqtlg kelkfliqhyekverdk
1clzH -lss-gvrtvs svlqsgfyslsSLVTvp sstwpsqTVICNVAhp- -askteLI-KRIEp---
1nfdB gflpsrfsvqq fdd----yhSEMNMsal eled--saMYFCASSlr wgdeqYFGPGTRltvle
```

D. DaliLite alignment (*fcar*$_0$=0%, TMscore=0.543)

```
1clzH tttapsvyplv pgcsdtsgssvTLGCLv kgyfp-epVTVKWN--- ----------------y
1nfdB -dsgvvqsp-- --rhiikekggrsVLTC IpisghSNVVWYqqtlg kelkfliqhyekverdk
1clzH galssgvrtvs svlqsgfyslsSLVTVp sstwpsqTVICNVAhpa ----skte-LIKRIEp
1nfdB gflpsrfsvq- qfdd---yhSEMNMSA- --leledsaMYFCASSl rwgdeqYFGPGTRltv
```

E. Deepalign alignment (*fcar*$_0$=0%, TMscore=0.547)

```
      11          23                40                57
1clzH vp--gcsdtsg ssvTLGCLvkg-yfpep VTVKWN----------- ---------ygalssgv
1nfdB hiik------e kggrsVLTCIpisghSN VVWYqqtlgkelkfliq hyekvqrdk-gflpsrf
      74          85                102               119
1clzH rtvssvlqsg  fyslsSLVTVpsstwps qTVICNVAhp----ask t-eLIKRIEp--r
1nfdB svqqfd----  dyhSEMNMsaleled-- -saMYFCASSlrwgdeq YFGPGTRltvle-
```



(b) DaliLite Structural Alignment

(b) UniAlign Structural Alignment

Figure 14 Case study: comparison of structure alignments of proteins 1clz H:115-231 and 1nfd B:1-117 from the cd00096 family of all-beta immunoglobulin proteins. (a) Residue correspondences from the reference CDD alignment and structure alignments. The core residues are shown in capital letters and marked in purple. (b) and (c) display the 3D geometric superposition of the DaliLite and UniAlign alignments. Structures are drawn in Jmol [80], with the aligned residues shown in thicker

Since the weights in our UniScore formulation control the contributions from different types of information, we can adjust these parameters to better align protein families with unique requirements. Setting aside a single test protein from the CBP family, we optimized the parameters using the rest of the CBP proteins. The test protein is then aligned with each of the CBP proteins and the accuracy is recorded. This optimization and testing procedure is repeated with each CBP protein set aside as the test case. The accuracy of the structure alignments obtained before and after this optimization is shown in Table 6. UniAlign achieves a boost of 27.4% on the average, when the family-specific optimization is performed.

We observed that optimization of the parameters for the CBP family reduced the weight of the geometric component from 0.29 to 0.06, while increasing the weights of the other components. This again illustrates the benefit of incorporating non-geometric features into a structural alignment method to detect functionally equivalent residues even under big conformational differences in the structures.

One such shifted alignments as mentioned previously are shown in Figure 14, which is a pair of immunoglobulin folds from cd00096. Only UniAlign correctly aligns all of the core residues, where the other methods suffer from different magnitude shift errors. The generated TMscore for all the methods are: 0.5417, 0.5396, 0.5428 and 0.5465. This means these alignments are equally good in terms of geometric similarity, yet their sequence alignment are quiet different.

Table 6 Comparison of performance on CBP family before (lower triangle) and after (upper triangle) the family-specific optimization. For each protein pair, better of the two alignments is shown in bold.

$fcar_0$	1aj4	1br1	1tn4	2sas	2scp	3cln	4cln	5tnc
1aj4		**.952**	**.975**	**.946**	**.785**	**.965**	**.966**	**.975**
1br1	.425		**.959**	**.688**	**.837**	**.972**	**.972**	**.959**
1tn4	**.975**	.338		**.952**	**.859**	**1.00**	**1.00**	**1.00**
2sas	.390	.382	.418		**.966**	**.957**	**.963**	**.946**
2scp	.444	.433	.454	.931		**.842**	**.843**	**.854**
3cln	.865	.958	.539	.454	.453		**1.00**	**1.00**
4cln	**.966**	.958	.898	.472	.450	1.00		**1.00**
5tnc	.968	.476	1.00	.432	.473	1.00	.898	

3.6 Conclusion

In this chapter, we have introduced **UniAlign**, a new structural alignment method that integrates different sources of information in order to achieve a more accurate alignment. Compared to classical methods that utilize only the geometry of the proteins and the recently developed methods that incorporate sequence information; UniAlign produces alignments that are in better agreement with expert-curated datasets. UniAlign is robust with respect to the sequence homology or the geometric similarity levels of the proteins being aligned. Furthermore, adjustment of UniAlign's parameters allows for development of family-specific models that highlight the features most relevant to the proteins in that family.

The increased accuracy achieved by UniAlign is at the cost of increased demands in computing time. For an average sized pair of proteins, it can take up to 15 minutes to calculate a structure alignment, with most of this time spent on the homology search to construct a multiple sequence alignment. The running times can be significantly reduced by caching and re-using the evolutionary information calculated for each protein in their alignments with different proteins. A detailed running time analysis is provided in the supplemental data.

We expect a number of downstream applications to benefit from the additional accuracy provided by UniAlign. Ability to develop family-specific alignment models will find use in structure classification problem. Integration of evolutionary information is likely to improve the protein-protein interaction prediction protocols that rely on structural alignment.

CHAPTER 4: PREDICTION OF HIV-1, HUMAN PROTEINS INTERACTION

4.1 Introduction

Human immunodeficiency virus type I (HIV-1) uses host surface proteins to gain entrance into the host cell. Interaction between HIV-encoded proteins and human proteins is important in the course of HIV-1 infection [21]. Thus, understanding the protein-protein interaction (PPI) between HIV-1 and human proteins provides critical insights into how the pathogen manipulates the biological pathways and processes of the host and subsequently helps the design of new therapeutic approaches. Computational approaches for protein interaction in the pathogen-host context are of significant value as large-scale experimental characterization of these interactions is expensive in terms of time and money [22].

Several computational PPI methods have been applied for HIV-1 - human interactions. Tastan et al. integrated multiple features information including Gene Ontology (GO), properties of human interactome and sequence motifs, and employed random forest method to predict protein-protein interactions [23]. Evans et al. predicted possible interactions using the presence of conserved sequence motifs and counter domain in both HIV-1 and human proteins [24]. The rapid progress in structure determination technologies gave rise to the establishment and deposition of large-scale protein structure in Protein Data Bank (PDB), with over 80,000 protein structures currently deposited [25]. The central assumption in predicting in pathogen-host interaction prediction based on structural similarity is that, for those defined structures and associated interactions, proteins with similar structures or substructures might share same interaction partners. Doolittle et al has already applied structure similarity based method to predict interactions between HIV-1 and human proteins, using the Dali Database for structure comparisons [26].

For the existing structure search approaches, pairwise structure alignment is the basic step for calculating the similarity between two structures. Structure alignment is also a better way to find distantly similar biological functions and evolutionary relationships than sequence alignment, considering that structure is more conserved than sequence. Several popular structure alignment methods have been developed, such as DALI [12, 81], CE [7], TM-align [8] and etc.. Although structure alignment methods are critically useful in discovering and understanding evolutionary relationships between proteins; available structure alignment methods use only the geometric information contained in the protein structures and do not incorporate known evolutionary information, e.g., as can be extracted from multiple sequence alignments.

Based on the hypothesis that proteins with similar structures share similar interacting partners, our previous study used a novel evolution-aware structure alignment method (UniAlign) to predict the interaction map between HIV-1 protein gp41 protein and all the human proteins [28]. First we retrieved all the human proteins sharing high structure similarity with gp41, by using both Dali and UniAlign. Second, we extracted all the known interactions for those HIV-1 similar human proteins, as the interacting partner candidates of this HIV-1 protein. Evaluation of the predictions showed a statistically significant overlap between the majority of our predictions and the HIV-1, human interactions verified experimentally. The predicted host proteins list could be very effective in assisting validation of interaction partners of HIV-1 experimentally by prioritizing those predicted protein – protein interactions. Our previous study also showed that UniAlign outperformed Dali in terms of finding the human proteins sharing structural similarity with HIV-1 gp41 protein, with better classification accuracy as measured by the area under curve (AUC) of the precision recall curve. We concluded that a structure alignment algorithm incorporating conservation profiles of the protein would better capture the similarity of the structures, especially in the context of protein – protein interactions.

With the increasing amount of protein structural data, we gain more knowledge about protein-protein interactions. For example, there are localized regions on the protein surfaces that are conserved among structural neighbors that participate in PPIs [29]. In other words, a protein interacts with its partners through the interface region on its surface. The protein-protein interface is defined as the contact region between two interacting proteins or two complementary chains [82]. And the properties of the protein – protein interface have been deeply studied. For example, compared with non-interacting residues, interacting residues are evolutionarily more conserved than the other surface regions [31]. Also, amino acid propensities vary significantly between interface region and other surface residues [32]. From an energetic perspective, the residues in the protein interface regions contribute unequally to binding, among which some of these residues, called 'hot spots', play exceptional roles [33]. PRISM is the first algorithm that uses structure and sequence conservation in protein interfaces for protein-protein interaction prediction [30]. The advantage of studying interface scaffold is that regardless of dissimilar global sequence or structure folds, proteins can still interact through similar interface scaffold [34]. Therefore, interface architectures, rather than global sequence or structure similarity, is used in our study to model protein complexes.

In this study, we computationally predicted the interactions between HIV-1 and human proteins, based on the hypothesis that proteins with similar interface architecture share similar interaction partners. We made use of all the known interfaces extracted from all the complexes deposited in protein data bank (PDB). When the query HIV-1 protein is structurally similar to the interface architecture of either partner of an already known interface, then the HIV-1 protein may also interact with the complementary partner, through the known interface scaffold pattern, to form a complex, regardless of the global structure similarity. In order to get the similarity between two interface architectures, we used our evolution-aware structure alignment method UniAlign, since UniAlign integrates multiple forms of evolutionary information and thus can better capture equivalence. Using those experimentally verified HIV-1, human protein-protein interactions data, we first did feature selection to narrow down to 12 features, including geometric similarity, conversion similarity and etc.; we then trained a support vector machine (SVM) with Gaussian kernel for the binary classification problem: whether a given protein pairs ‘interact’ or ‘no interact’. We used the trained and tuned SVM classifier to discover potential novel HIV-1 interacting partners for human proteins. Many predicted interactions had significant literature support, and we modeled the novel 3D interacting complex for HIV-1 envelope gp120 and gp41 proteins. We provided the first structural evidence for those interactions.

4.2 PPIs Prediction Based on Structural Similarity

4.2.1 Methods

The pipeline of our method to predict HIV-1, human protein interactions are shown in Figure 15.

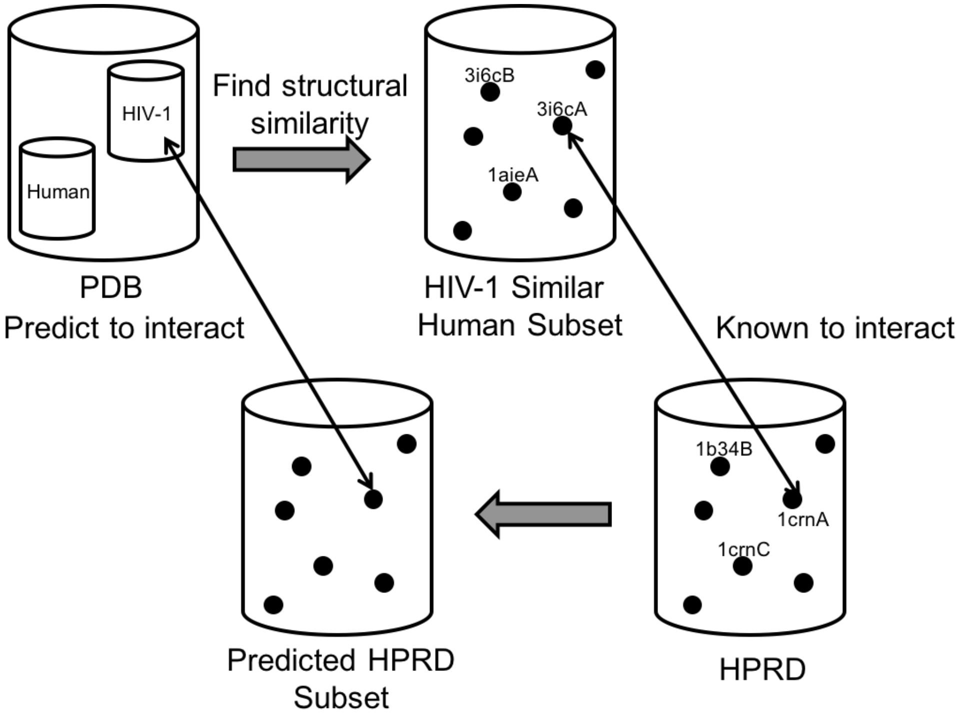


Figure 15 Protein-protein interaction prediction pipeline

Datasets

We downloaded the HIV-1 and Homo sapiens protein structures from Protein Data Bank (PDB) [25]. In order to compare the host protein prediction lists generated by two different structure alignment methods, we extracted all PDBs used in Dali Database (updated in 2011) [81]. Dali Database contains the structural alignments of PDB90 against the all PDBs, as well as the corresponding Z-scores, indicating the structural similarity. Only those protein pairs with Z-score above 2.0 are saved in the Dali Database. Also those human proteins in Dali without Refseq IDs were also eliminated. Then we took the intersection of those two human protein lists as all human protein structures for our calculation. There are 5659 unique human proteins, with 29041 unique structure chains. The various databases we used for our prediction and evaluation contains their own different identifiers. PDBs for each structure obtained from either Dali Database or PDB were mapped to the Uniprot accessions by using PDB/UniProt Mapping [83]. And the conversion between Uniprot accessions and Refseqs were realized by using Uniprot ID mapping Database [84].

HIV-1, Human Interaction Database

In the HIV-1, human interaction database, each interaction between HIV-1 and human proteins is represented by one or more descriptive key phrase, such as “increases”, “unregulated by” or “phosphorylates” [85]. Only the direct interactions defined by Tastan et al [23] were included for our prediction validation since we tried to predict physical interactions. More constraints were added to the use of HHPID. For example, the HIV proteins in HHPID should be represented among the crystal structures from PDB that are included within the Dali Database. Besides, any host proteins shown to interact with HIV-1 in HHPID must have at least one known interaction with another human protein included in HPRD, and what’s more, each of these proteins must also have representative structures in Dali Database. Take the ENV’s cleavage products, gp41 as a case study, 7 different proteins with 41 structures existed in PDB. However, only one protein P04578 out of seven was verified in HHPID, showing the limits of experiments and at the same time the importance of predicting the interaction using computational methods.

Representation of Virus-Host Interaction Prediction

We predicted the map of virus-host interaction for each HIV-1 protein. Multiple structures may represent the same protein, while different structures have different multiple sequence profiles, resulting in different conservation weights, thus the predicted interactions for the same protein's different structures were slightly different, yet some of them are redundant. Therefore, we used the structures (pdbchains) to evaluate the two structures alignment methods, and identified all unique pairs of Uniprot accessions to evaluate the interaction prediction performance, as with what Doolittle did in [26].

4.2.2 Results and Discussion

Identification of HIV-1 structure-similar Human Proteins

For each HIV-1 protein, all the different structures were aligned versus all the human protein structures using two different pairwise structure alignment methods. The gp41 protein P04578 we used has five different PDBs: 1df4A, 1df5A, 1dlbA, 1k33A and 1k34A. For Dali Database, the HIV-1 structure similar human proteins were defined as those having a Z-score higher than 2.0, with the HIV-1 protein being either the query or the hit. For Unialign, we used uniscore as the structural similarity metric, a weighted version of TM-score, giving different weights to residues according to their conservation. And all the human proteins with uniscore above 0.72 were defined as HIV-1 structure similar human proteins. We chose 0.72 to generate comparably size of both prediction lists. Table 7 shows the number of HIV-1 structure-similar human proteins calculated by both Dali and Unialign. We could see that this specific gp41 protein has regions of high similarity to human proteins.

Table 7 The number of HIV-1 structure-similar human proteins calculated by Unialign and Dali

pdbchain	UniAlign	Dali
1DF4A	121	29
1DF5A	34	52
1K33A	37	27
1K34A	119	59
1DLBA	57	67

Prediction of human proteins interacting with HIV-1 proteins

After obtaining the human proteins that sharing high structural similarity with each specific HIV-1 protein structure, the interaction partners of each HIV-1 structure-similar human proteins were obtained using Human Protein Reference Database (HPRD), which contains 38,989 unique documented protein-protein interactions [86]. We denote the predicted target human proteins as the subset HP. Our hypothesis was that proteins with similar structures or substructures might share the same interaction partners. Besides, during the HIV-1 infection, the virus modifies or destroys the already existing interactions between human proteins thus it could use the existing communication pathways within the cell for its own reproduction. Human proteins and HIV-1 proteins, in a way, compete for the same interactions. Thus, HP was treated as the potential targets for the corresponding HIV-1 proteins, and thus the interaction map between each HIV-1 protein and target human protein was established.

Validation of predictions using the HIV-1, Human Interaction Database

For validating the predicted interactions, we compared the predicted target human protein HP set with the experimentally acquired human protein interactions with HIV-1, which are compiled in the Human Protein Interaction Database (HHPID) [85]. There are 1036 known host-pathogen interactions in HHPID satisfied our criterion (seen in Methods), between 20 HIV-1 proteins and 528 human proteins, denoted here as the HE set. Then the p-value for the overlap between computational sets HP and experimental sets HE was calculated using the hyper-geometric test, showing the probability to obtain our predictions simply by chance. A total of 922 unique target human protein (HP) were predicted to potentially interact with gp41 protein P04578 and matched 15 out of 68 experimentally verified interactions. Four out of five predictions generated by Unialign has a statistically significant overlap ($p<0.05$) with the experimentally known ones, yet only two predictions generated by Dali are statistically significant (Table 8). Thus UniAlign's performance is better than Dali in terms of the interaction partner prediction of each HIV-1 structure. To predict the interaction for a specific structure has huge practical meaning, since in reality, we want to predict the host proteins of known, highly-mutated virus structure as accurate as possible, and thus prioritizing those predicted protein-protein interactions for experimental validation.

Table 8 The number of HP, HE and Match of each gp41 protein structure, as well as the p-values for the overlap between HP and HE calculated by hyper-geometric test.

Method	pdbchain	HP	HE	Match	p-value
UniAlign	1DF4A	759	68	14	3.31E-02
	1DF5A	508	68	10	3.82E-02
	1K33A	528	68	13	3.45E-03
	1K34A	644	68	10	1.45E-01
	1DLBA	649	68	12	4.28E-02
Dali	1DF4A	619	68	5	7.69E-01
	1DF5A	1003	68	15	1.36E-01
	1K33A	587	68	15	1.25E-03
	1K34A	783	68	16	9.55E-03
	1DLBA	1344	68	16	4.50E-01

Comparison of the two structure alignment methods

In order to better assess the performance of Unialign and Dali, we also employed the Area under curve (AUC) to summarize the precision vs. recall curve, the range of which ranges between 0 and 1. The precision vs. recall curve of these two methods is shown in Figure 16 and the AUC scores of each gp41 structure is summarized in Table 9.

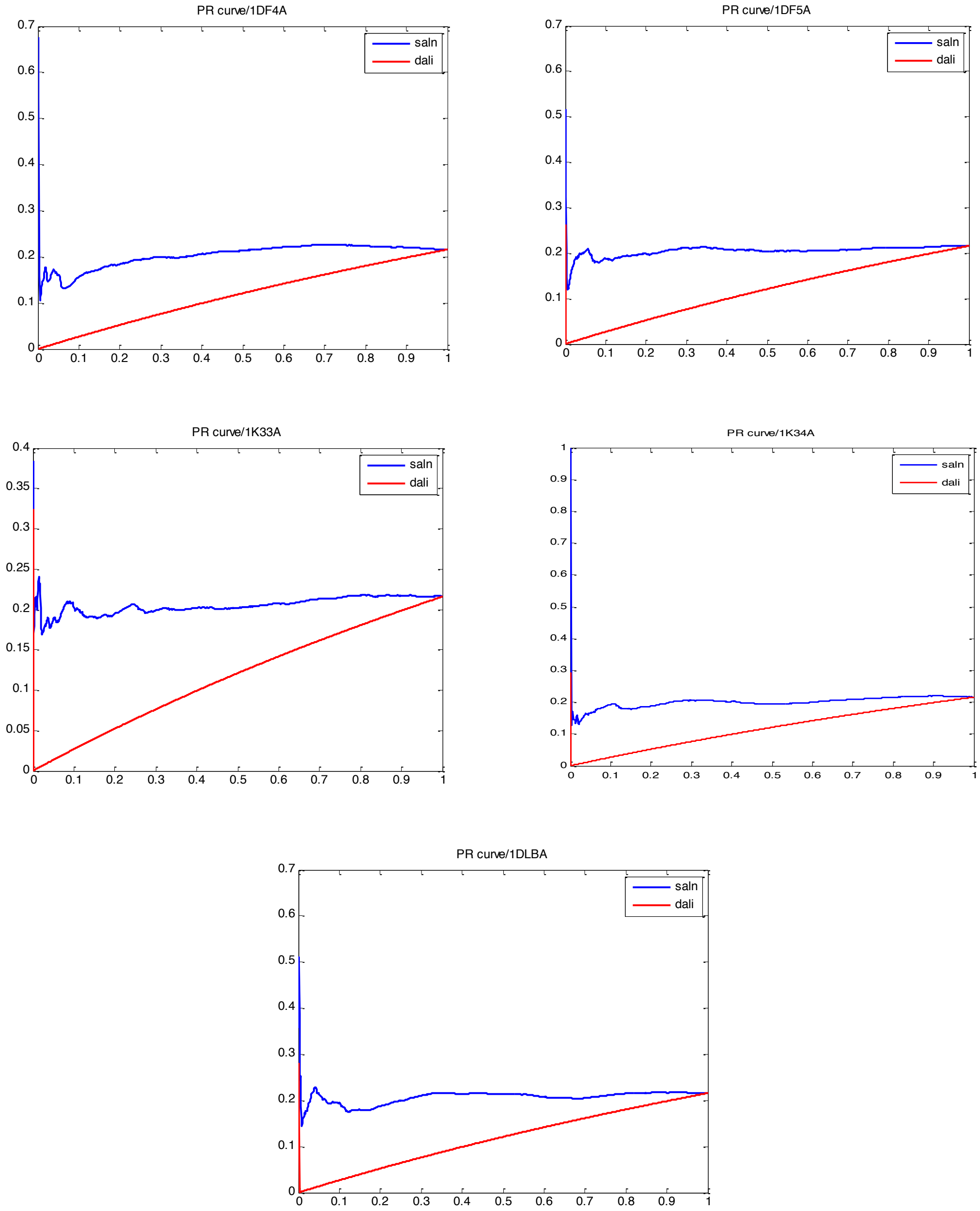


Figure 16 The precision vs. recall (PR) curve of Unialign (blue) and Dali (red). From the top left are 1DF4A, 1DF5A, 1K33A, 1K34A and 1DLBA.

Table 9 Area Under Curve (AUC) Score of the Precision vs. Recall Curve for Each Structure

pdbchain	**UniAlign**	**Dali**
1DF4A	**0.2043**	0.1167
1DF5A	**0.2051**	0.1170
1K33A	**0.2056**	0.1170
1K34A	**0.2015**	0.1170
1DLBA	**0.2075**	0.1172
mean	**0.2048**	0.1170

We could observe that Unialign performed much better than Dali to calculate the human proteins sharing high structural similarity with the HIV-1 protein. The reason might be that Dali only used geometric information to align structures. However we believe that a structure alignment method that considers conservation profiles of the protein would better capture the similarity of the structures, especially in the context of protein-protein interactions. In fact, the mean AUC scores of Unialign is 20.48%, indicating that of all pairs that were predicted as interacting, 20.48% on average are correct, compared to the average 11.7% TP of Dali.

4.3 PPIs Prediction Based on Interface Architecture Similarity

4.3.1 Methods

There are 6 steps for the binary classification problem of HIV-1, human PPIs based on interface architecture similarity: (1) extracted the known interfaces from PDB; (2)

collected experimentally verified HIV-1, human PPIs as the ‘interacting’ training data; (3) for each known interface, compare the HIV-1 protein with either partners using UniAlign; (4) feature extraction from multiple biological sources; (5) feature selection to select those most informative subsets; (6) trained and tuned the SVM, evaluate the performance and ready for further use. Figure 17 shows the flow chat of all the steps.

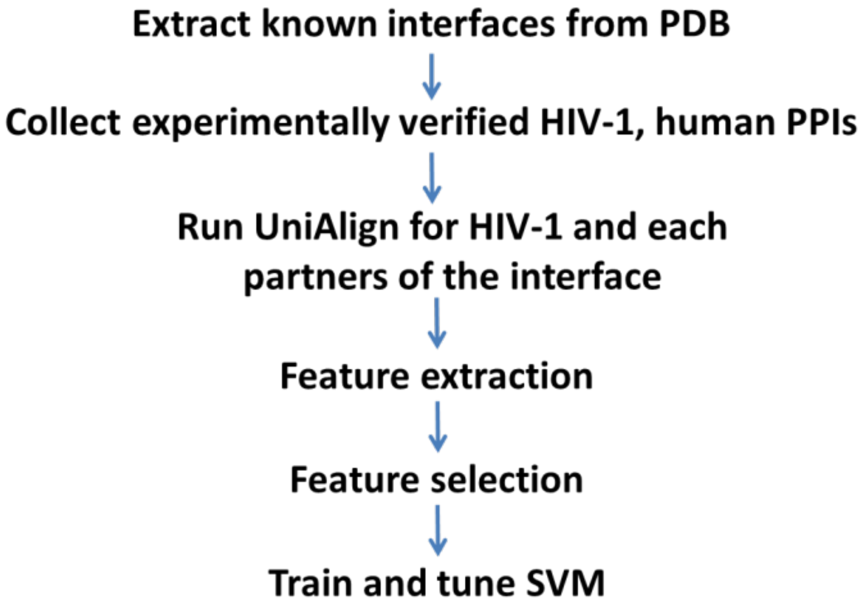

Extract known interfaces from PDB
Collect experimentally verified HIV-1, human PPIs
Run UniAlign for HIV-1 and each partners of the interface
Feature extraction
Feature selection
Train and tune SVM

Figure 17 The flow chats of the supervised classification of HIV-1, human PPIs using support vector machine

Extract known interfaces from PDB

Although the information needed to predict whether two proteins interact seems to be in the PDB, yet the question is how to mine the data. Existing protein-protein interface scaffold patterns can be used to model the complex structures between two query proteins, despite their global structures similarity. Thus we need to collect all the known interfaces from all the complexes deposited in the protein data bank (PDB). We checked all the PDB entries as of January 14th 2013 to extract the interface information. Even though we would like to generate all binary interactions between every single chains and check for interacting residues, yet due to the time limits, we adopted the structurally non-redundant interface dataset from [87]. We used this dataset to be able to search for interactions patterns in reasonable time.

Benchmarks

We collected the experimentally verified HIV-1, human protein-protein interactions from four widely used benchmarks: BioGRID, HHPID, IntAct and DIP. Since we tried to predict physical interactions between proteins, only the direct interactions, either defined by Tastan et al [23] or labeled as MI: 0407 (direct interaction) (defined by the HUPO Proteomics Standards Initiative PSI), were included in our benchmark datasets.

The Biological General Repository for Interaction Datasets (BioGRID) is a public database that archives both genetic and protein interaction data from model organisms and humans [88]. There are 759 direct HIV-1, Human PPIs represented in GeneID – GeneID pairs. In the HIV-1, human interaction database (HHPID), each interaction between HIV-1 and human proteins is represented by one or more descriptive key phrase, such as “increases”, “unregulated by” or “phosphorylates”[85]. There are 4868 direct HIV-1, human PPIs represented in GeneID – GeneID pairs. The IntAct database of EMBL-EBI is an open source database for molecular interactions [89]. There are 8 direct interactions between HIV-1 and human, represented by UniProt – UniProt pairs. The

Database of Interacting Proteins (DIP) is a biological database which catalogs experimentally determined interaction between proteins [90]. And there are 3 direct interactions between HIV-1 and human, represented by UniProt – UniProt pairs.

Representation of Virus-Host Interactions

It is essential to map the virus-host interactions identified from different databases to UniProt accession numbers and then using pairs of UniProt accession numbers as the unique identified of all PPIs. The conversion between UniProt accessions and GeneIDs were realized by using UniProt ID mapping Database [84]. Second, the UniProt – UniProt pairs were converted into non-redundant PDB chain – PDB chain pairs. PDB chains for each structure were mapped to the UniProt accessions by using PDB/UniProt Mapping [83].

Due to the high redundancy of structures deposited in the PDB for UniProt accession, for each UniProt protein, we filtered out highly similar structures (sequence similarity 95%); and preferably chose higher resolution structures. We further filtered out structures without any interface information, with the help of the interface template dataset. Take HIV-1 gag protein P04585 as an example, it has 158 deposited structures in PDB, and after the filtering to remove highly similar structures/substructures, it has 4 non-redundant structures. There are 694 unique HIV-1, human UniProt-UniProt interactions, corresponding to 1930 non-redundant HIV-1, human PDB-PDB interaction pairs. Figure 18 shows the process of collecting interactions from different databases and further converting them into PDB – PDB format.

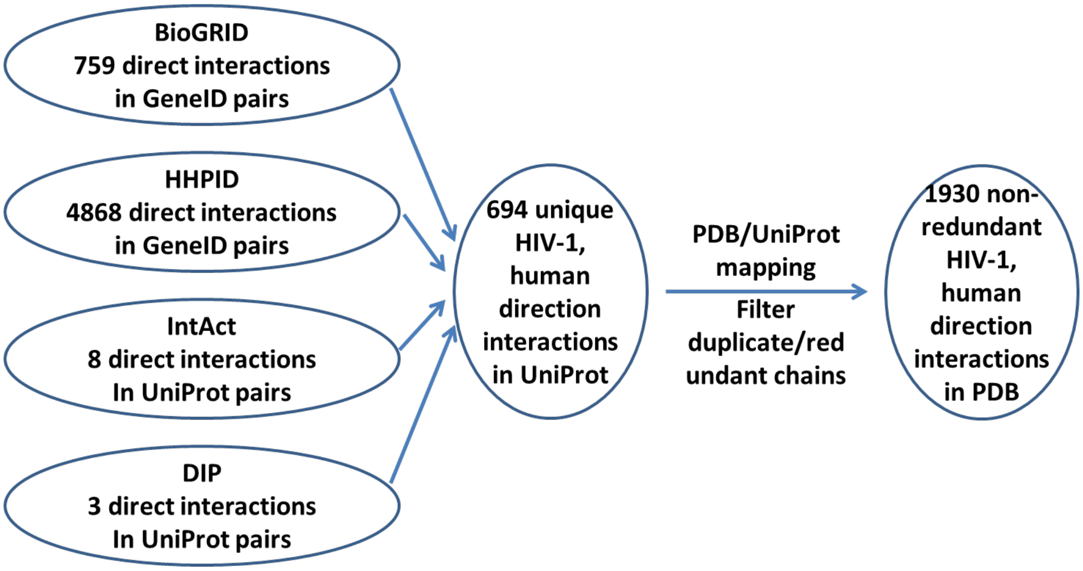


Figure 18 Collecting experimentally verified HIV-1, human PPIs from four benchmarks, and ID mapping into PDB chain-PDB chain format.

Non-interacting Dataset

We used all the non-redundant, experimentally verified HIV-1, human protein-protein interactions in PDB chain- PDB chain format as our true positive dataset. And as mentioned before, there are 694 unique HIV-1, human UniProt-UniProt interactions, corresponding to 1930 non-redundant HIV-1, human PDB-PDB interaction pairs.

It is impossible to prove that two proteins do not interact, thus no “gold standard” negative dataset is available. Cukuroglu et al collected all possible binary interactions of the protein structures deposited in the Protein Data Bank (PDB) as of January 14th 2012 and further clustered all PPIs by interface structures by a community finding algorithm in graph theory [87]. We decided to make use of the clustering knowledge to better generate the ‘true negative’ dataset. We downloaded the interface clusters dataset “finalInterfaceClusters_2013_January_24_.txt” and there are 22604 structurally non-redundant interface structures in this dataset.

To be specific, for each positive interaction pair, we randomly chose human protein structures outside the interface cluster of the interacting human partner, and considered it as negative interactions. So that the similar interface pattern is not considered for contracting the negative dataset. The assumption behind this decision is that the probability of two randomly chosen proteins to interact is small. Thus we constructed 1930 non-interacting pairs.

UniAlign: evolution-aware structure alignment method

Interface Architecture Definition

In our study, the interface architecture is defined as the surface residues together with their neighboring residues in sequence order. The surface residues are calculated based on the relative accessible surface area (RASA) values calculated by NACCESS [91]. Specifically, we extracted the RASA value from the generated *.rsa file, which is the relative accessibility of each residue calculated as the accessibility compared to the accessibility of that residue type in an extended ALA-x-ALA tri-peptide (for amino acids). And a residue is defined as surface residue it its RASA value is more than 40% [87]. The surface residues are normally isolated residues, scattered throughout the whole protein structures. Thus, we extend the nearby residues which are up to 3 residues away from the surface residues in sequence order, instead of spatially neighboring residues. We further fill in small gaps in the sequence order which are smaller than 25 residues for better consistency. Adding neighboring residues makes the interface scaffold consistent, which is important to get a proper structural alignment of the interface architectures. Different studies defined the neighboring residues in different ways [92-94].

Calculate Interface Similarity using UniAlign

First, I would like to clarify the representation of a interface. Take the 1crnLR as an example, 1crn denotes any PDB IDs, L denotes the ‘left chain’, also means the ‘template chain’, and R denotes the ‘right chain’, also means the ‘interact chain’.

To calculate the structural similarity between two interface architectures, we chose UniAlign as our structural alignment approach. The advantage of UniAlign as a structure alignment method is that it integrates multiple forms of evolutionary information, such as sequence profile similarity, conservation similarity and secondary structure similarity, during the process of searching the optimal rotation/translation matrix [95]. Specifically, it is mentioned that even remotely related proteins often use regions of their surface with similar arrangements of secondary structure elements to bind to other proteins [29, 96]. Besides, since interacting proteins tend to co-evolve, proteins with similar sequence profiles are predicted to interact. Thus the phylogenetic profile similarity score was further calculated given a pair of phylogenetic profile vectors. Thus UniAlign generated the most biologically meaningful correspondence for the given protein pairs.

4.3.2 Feature Extraction

After structurally transforming the HIV-1 proteins onto the ‘template chain’ of the human protein, together with the generated complex with complementary ‘interact chain’, we are ready to extract all the features we need. It is worth noting that one 'interact chain may have multiple binding patterns, and we took all the available binding 'templates' and run UniAlign. There are 2673 UniAlign records for the interacting pairs and 3199 for non-interacting pairs.

Calculate the Number of Contacting Residues for a Given Complex Model

Contact residues refers to the residues from different chains if the distance between any two atoms of them is less than the sum of their corresponding van der Waals radii plus 0.5 Angstrom [87].

Feature Sets

Considering that one HIV-1 protein (*hivprotein*), and a known interface (*pdbaLR*, where *pdba* represents the PDB ID, *L* represents the template chain, and *R* represents the interact chain). We ran UniAlign for *hivprotein* and *pdbaL* to calculate the structure similarity of the interact architectures. We further superimposed the *hivprotein* onto the template chain *pdbaR* to form a new complex *hivproteinRotate-pdbaR*. We also included the already known complex *pdbaL-pdbaR*. The calculated interface architecture of *pdbaL* is denoted as *pdbLinterface*. All the 18 features we calculated were listed in Table 10, ranging from various similarity scores to length information of query protein as well as query protein's surface region.

Table 10 Features extracted for classification

Feature	**interpretation**
hivsize	*length*(*hivprotein*)
templatesize	*length*(*pdbaL*)
templatesurfacesize	*length*(*pdbLinterface*)
interactorsize	*length*(*pdbaR)*
humanoverhivsizeratio	*length*(*pdbaL*) / *length*(*hivprotein*)
humansurfaceoverhivsizeratio	*length*(*pdbLinterface*) / *length*(*hivprotein*)
alignedpairsnogap	refer to the result of UniAlign in specific aim 2.
alignedpairsnogapoverhiv	*Alignedpairsnogap / length*(*hivprotein*)
alignedpairsnogapoverhuman	*Alignedpairsnogap / length*(*pdbaL*)
contactnum	overlap between *contacttemplate* and *contactunialign*
contacttemplate	number of contacting residue for complex *pdbaL-pdbaR*
contactunialign	number of contacting residue for complex *hivproteinRotate-pdbaR*
uniscore	refer to the calculation of UniScore in specific aim 2.
tmscore	refer to the calculation of UniScore in specific aim 2
unipro	refer to the calculation of UniScore in specific aim 2
unicon	refer to the calculation of UniScore in specific aim 2
unisse	refer to the calculation of UniScore in specific aim 2
uniseq	refer to the calculation of UniScore in specific aim 2

4.3.3 Support Vector Machine

The Support Vector Machine (SVM) has been widely in machine learning field for classification problems. A SVM classifies data by finding the best hyper plane that separates all the features of one class from the data points of another class. The best hyperplane for an SVM represents the one with the largest margin between the two classes, where margin is the distance from the separating plan to the closest vector. To train the SVM, the *n*-dimension feature vectors are used as the input to the SVM, and are mapped to vectors of a high-dimensional space using the kernel function. The SVM attempts to construct a hyperplane in that space that separates the vectors associated with interacting pairs from those that are non-interacting.

Same with any supervised learning problem, we first trained a support vector machine with the feature set derived from several biological information sources, using the *fitcsvm* function in MATLAB. In order to obtain good predictive accuracy, we then tuned the parameters of the Gaussian kernel functions, by pass the feature data to *fitcsvm* function in MATLAB with the *KernalScale* to auto. Thirdly, we cross validated the trained classifier.

4.4 Experiments

4.4.1 Feature Selection

Feature selection methods reduces the complexity of the interacting classification, which can be broadly classified into two approaches: classifier-dependent (‘wrapper’ and ‘embedded’ methods) and classifier-independent (‘filter’ method) [97]. In our study, we implemented the sequential backward selection (SBS), so that features are sequentially removed from an initially full candidate set until the removal of further features no longer improves the prediction.

We first tuned the SVM with all the features: to tune the SVM classifier, we pass the feature data to *fitcsvm* function in MATLAB with the *KernalScale* to auto. And Matlab uses a heuristic method to select the kernel scale, with the *BoxConstraint* = 11.91 to preventing overfiiting, and *KernelScale* = 42.78 for the Gaussian kernel function.

For the input feature matrix, each row corresponds to one feature vector, which are 19 dimensions. In order to balance the sensitivity and specificity, we selected area under the curve (AUC) value as our objective function / selection criteria. To be specific, start with the full candidate feature subset, sequential feature selection methods performs a 10-fold cross-validation by repeatedly evaluating different training subsets of the input matrix and corresponding response variable. In the cross-validation calculation for a given candidate feature set, we used the mean value of all the return criterions to evaluate each candidate feature subset. For each round, the sequential backward feature selection chooses the candidate feature subset that minimized the mean criterion value. And the searching continues until removing more features increase the loss criterion.

Figure 19 shows the performance of the SVM with the removing of the features at each step. SVM classifier performance was evaluated with 10-fold cross validation to obtain average values. On the x axis, we have the number of features selected, and on the y axis we have the evaluation criterion, based on the AUC score. And we evaluated the quality of our predictive model using the received operating curve (ROC) as well as the area under curve (AUC) to summarize the ROC curve as a scalar score ranging between 0 and

1. As a result, we selected 13 out of the 18 features which are more informative to discriminate whether two protein structures interact or not. The SVM achieved the best 10-fold cross validation performance with an optimal combination of 13 features: *templatesize, interactorsize, templatesurfacesize, humansurfaceoverhivsizeratio, alignedpairsnogap, alignedpairsnogapoverhiv, contacttemplate,tmscore, alignedpairsnogapoverhuman, uniscore, unicon, and unisse*.

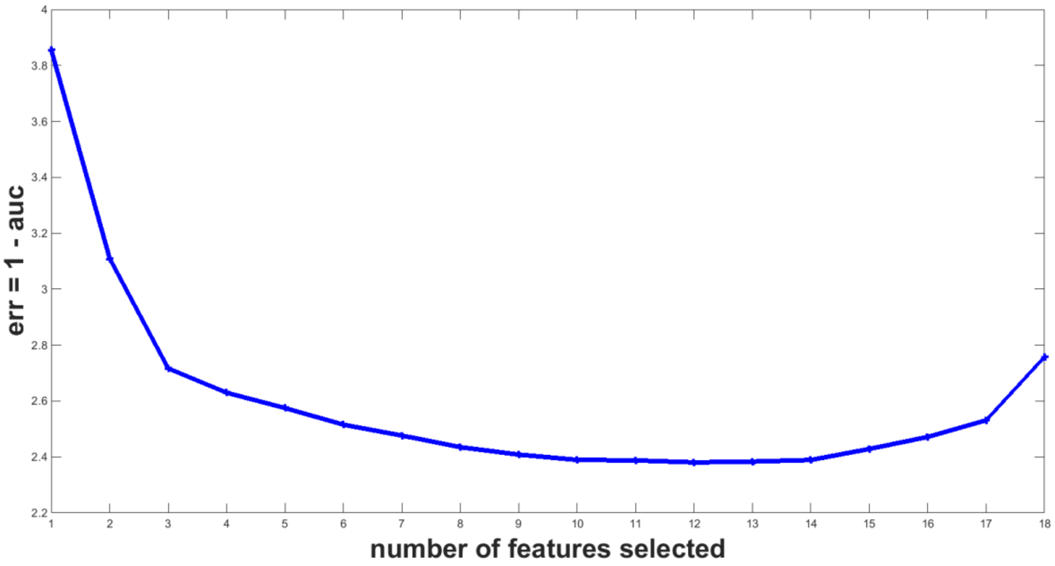


Figure 19 The performance of SVM with the changing of features number

Furthermore, Figure 20 shows the 10-fold cross-validated performance of all 18 features (dash line), and also the ROC curve with those selected 13 features (solid line). On the x axis we have the true positive rate, and on the y axis we have the false positive rate. The area under curve (AUC) of the ROC for all features is 0.72 and the AUC of the ROC with selected features is 0.82.

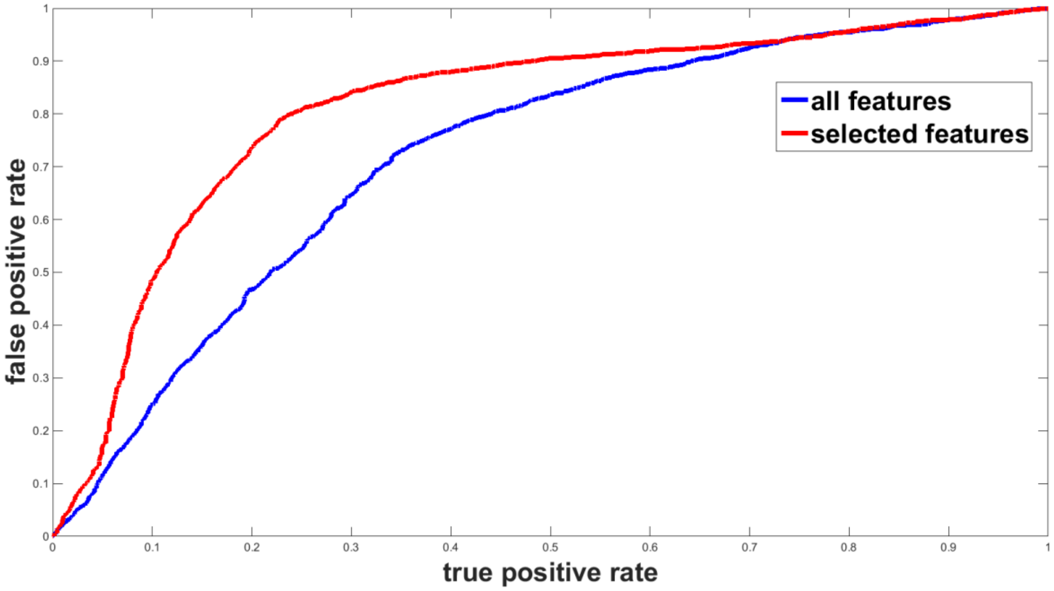


Figure 20 The comparison of AUC with all features and will selected features.

4.4.2 Discovering new PPIs

Due to the number of naturally occurring architectural motifs in protein-protein interfaces, we used the interface scaffold to model complexes between two query proteins, instead of the whole structures [92, 93]. To find the potential novel HIV-1, human PPIs, first we need to generate a list of candidate human proteins with existing interface information documented in the PDB.

There are three steps to generate this list. First, we downloaded all the human PDB structures (13914 non-redundant PDB IDs) as of February 25th 2013. Second, each interface in the interface template dataset mentioned in 2.3.1 is split into two constituent chains. Third, we filtered out PDBs without any interface information from the interface template dataset. One protein can interact with multiple partners using different regions of their surfaces, and different interfaces can be associated with different functions [98].

Therefore, a given template chain may correspond to multiple interact chains. And there are 28979 unique human structures in the list, represented as *PDBID – template chain* identifier – *interact chain* identifier.

For the envelope proteins of HIV-1, we chose the 3j70D, which contains the structure of both gp120 and gp41. Then we ran UniAlign for each human protein, extracted all the 13 features, and used the trained SVM classifier to calculate the posterior probability for each newly modeled complex. The higher the probability is, the more possible 3j70D directly interact with that human protein. These predicted human proteins list could be very effective in assisting identification of interaction partners of HIV-1 experimentally.

4.5 Results

Now we focused our discussion on some predicted interactions that involved human proteins known to be critical for HIV replication and propagation. Examples of predicted interactions with support in the literature include those necessary for viral attachment to the host membrane and subsequent invasion of the host cell.

4.5.1 Predicted Complex Formed Between 3j70D and 1btkB.

The SVM score for this prediction is 0.9903. The interface template used here is between chain A and chain B of 1btk. Both chains correspond to the PH-PH domain of the Tyrosine-protein kinase BTK (Q06187).

1) Binds GTF2I through the PH domain. Interacts with SH3BP5 via the SH3 domain. Interacts with IBTK via its PH domain.

2) Using high-throughput proteomic assays, [99] identified Bruton's tyrosine kinase (BTK) as a host protein that was uniquely upregulated in the plasma membrane of human immunodeficiency virus (HIV-1)-infected T cells. Significant upregulation of the phosphorylated form of BTK was observed in infected cells. They have found that HIV-1-infected cells are sensitive to apoptotic cell death and result in a decrease in virus production. We for the first time provided the structural evidence for the interaction between BTK_Human and HIV-1 gp 120.

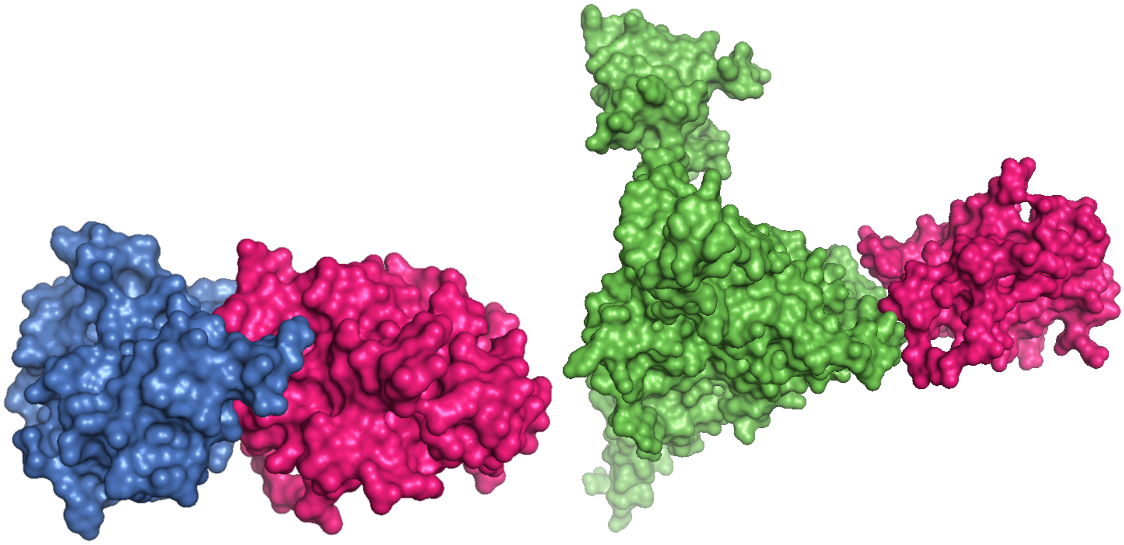


Figure 21 The complex formed by 1btkA_B (on the left) and the predicted complex formed by rotated-HIV1 and 1btkB (on the right).

4.5.2 Predicted Complex Formed Between 3j70D and 2j4eB.

The SVM score for this prediction is 0.9870. The interface template is formed between chain A and chain B of 2j4e. Both chains correspond to the Inosine triphosphate pyrophosphatase - Q9BY32 (ITPA_HUMAN). ITPA gene polymorphisms significantly affect hemoglobin decline and treatment outcomes in patients coinfected with HIV and

HCV [100]. It is also associated with early virologic outcomes. Determination of ITPA polymorphisms may allow prediction of RBV-induced anemia and earlier initiation of supportive care to ensure optimal therapeutic outcomes. Again, we provided the first structural evidence for the potential interacting between the gp120 and ITPA.

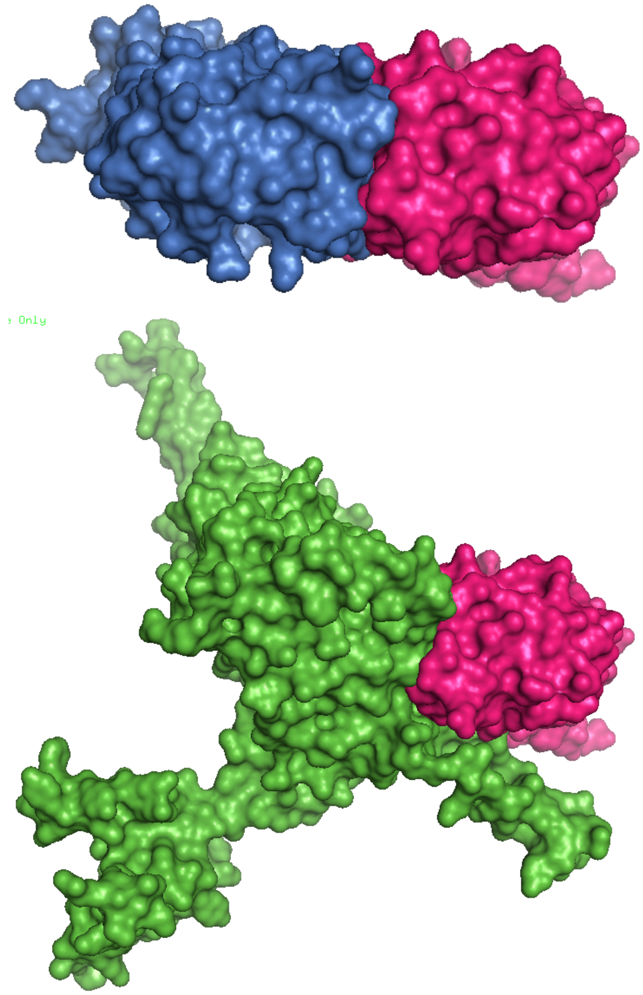


Figure 22 The complex formed by 2j4eB and 2j4eA on the top. And the newly generated complex formed by gp120 and 2j4eB on the bottom.

4.5.3 Predicted Complex Formed Between 3dnlB and 2hy3A.

The SVM score for this prediction is 0.9917. The interface template is formed by chain A and chain B of 2hy3. Both chains correspond to the Receptor-type tyrosine-protein phosphatase gamma - P23470 (PTPRG_HUMAN). For example, HIV-1 gp120 downregulates the expression of protein tyrosine phosphatase, receptor type C (PTPRC; CD45) in human B cells [101]. Also, CD45 modulates HIV-1 gp120-induced apoptosis by regulating Fas ligand induction and activation of the phosphoinositide 3-kinase/Akt pathway [102]. And our study shows the first structural evidence for these interactions.

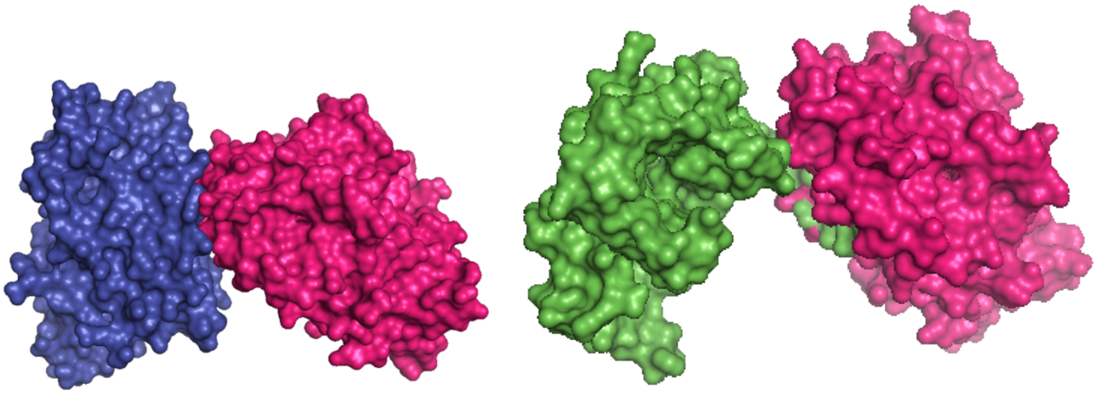


Figure 23 The complex formed by 1lf8B and 1lf8C on the left. And the newly generated complex formed 3dnlB and 1lf8B on the right

4.5.4 Predicted Complex Formed Between 3dnlB and 1lf8B.

The SVM score for this prediction is 0.9912. The interface template is formed by chain B and chain C of 1lf8. Both chains represent the ADP - ribosylation factor-binding protein GGA3 - Q9NZ52 (GGA3_HUMAN). GGA overexpression led to the formation of large, swollen vacuolar compartments, which in the case of GGA1 sequestered HIV-1 Gag

[103]. In addition, the following are the protein interactions in the NCBI website. siRNA-mediated depletion of GGA3 leads to a significant increase in particle release in an HIV-1 Gag late domain-dependent manner, while GGA3 overexpression severely reduces virus particle production by impairing Gag trafficking to the membrane [104]. Expression of a patient-derived HIV-1 Vpr protein reveals a significant reduction in Vpr nuclear import. Vpr F72L mutation is responsible for this decreased Vpr nuclear import and the F72L mutant is co-localized with gamma-adaptin in the Golgi apparatus [105].

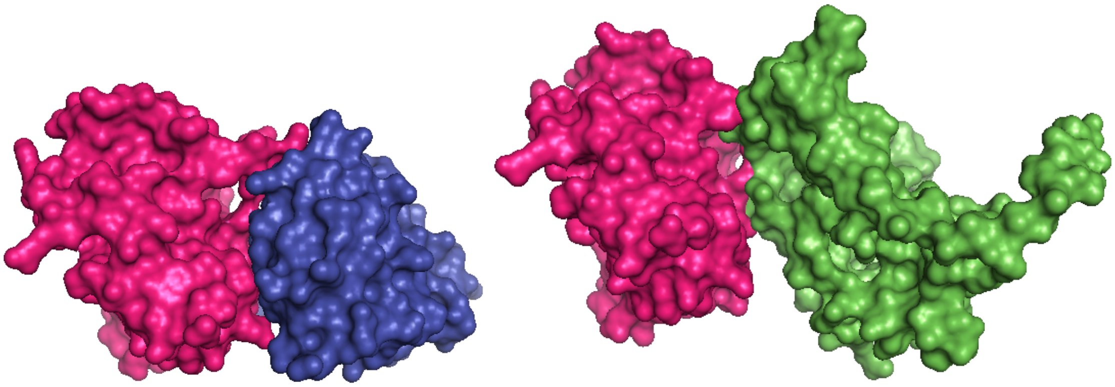


Figure 24 The complex formed by 2hy3A and 2hy3B on the left. And the newly generated complex formed by gp120 and 2hy3A on the right

4.5.5 Predicted Complex Formed Between 3j70D and 3a6nH.

The SVM score for this prediction is 0.9930. The interface template is formed by chain H and chain F of 3a6n. 3a6nH represent Histone H2B type 1-J - P06899 (H2B1J_HUMAN), while 3a6nF represent Histone H4 - P62805 (H4_HUMAN). There have been no literatures about potential interaction gp120 and 3a6nH.

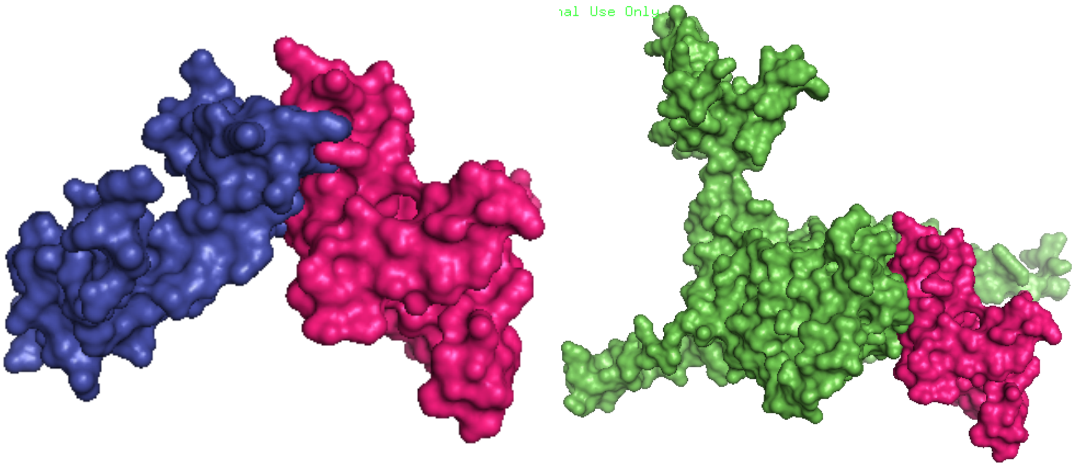


Figure 25 The complex formed by 3a6nH and 3a6nF on the left. And the newly generated complex formed by 3j70D and 3a6nH on the right

4.6 Conclusion

In this paper, we discovered potentially new virus-host interaction map between HIV-1 and human. Our method based on the assumption that human host proteins may be influenced during the HIV-1 infection, by interacting with certain HIV-1 proteins. Computational methods could be very effective in helping the experimental identification of these interaction, discovering novel virus-host interactions as well as potential clinical targets for therapeutic intervention. In the context of host-pathogen interaction prediction, especially for those highly mutated viruses such as HIV-1, our structural alignment method UniAlign could better capture the similarity of the more conserved residues and enjoy better prediction accuracy. Compared to other structural alignment methods, UniAlign employed the information of the conservation profiles of the proteins instead of only using the geometric information. Using those experimentally verified HIV-1, human protein-protein interactions data, we first did feature selection to narrow down to 13 features, including geometric similarity, conversion similarity and etc.; we then trained a support vector machine (SVM) with Gaussian kernel for the binary classification problem: whether a given protein pairs ‘interact’ or ‘no interact’. We used the trained and tuned SVM classifier to discover potential novel HIV-1 interacting partners for human proteins. Many predicted interactions had significant literature support, and we modeled the novel 3D interacting complex for HIV-1 envelope gp120 and gp41 proteins. We provided the first structural evidence for those interactions. Our method does not count on other functional genomic information, such as co-expression or cellular localization, and may be served as an addition contribution into an integrative computational framework for predicting novel PPIs based on information from multiple sources.

5. CONCLUSIONS AND FUTURE WORK

In this study, we first defined a per-residue (UniScore), a protein similarity score that incorporates additional evolutionary information captured in the form of sequence similarity, sequence profiles and residue conservation. UniScore is a weighted sum of all these the other features. We further introduced UniAlign, a new structural alignment method that integrates different sources of information in order to achieve a more accurate alignment. Compared to classical methods that utilize only the geometry of the proteins and the recently developed methods that incorporate sequence information; UniAlign produces alignments that are in better agreement with expert-curated datasets. UniAlign is robust with respect to the sequence homology or the geometric similarity levels of the proteins being aligned. Furthermore, adjustment of UniAlign's parameters allows for development of family-specific models that highlight the features most relevant to the proteins in that family.

The increased accuracy achieved by UniAlign is at the cost of increased demands in computing time. For an average sized pair of proteins, it can take up to 15 min to calculate a structure alignment, with most of this time spent on the homology search to construct a multiple sequence alignment. The running times can be significantly reduced by caching and re-using the evolutionary information calculated for each protein in their alignments with different proteins. A detailed running time analysis is provided in the Supplemental Data.

We expect a number of downstream applications to benefit from the additional accuracy provided by UniAlign. Ability to develop family-specific alignment models will find use in structure classification problem. Integration of evolutionary information is likely to improve the protein-protein interaction prediction protocols that rely on structural alignment.

We presented an application of UniAlign and support vector machine (SVM) for predicting physical interactions between HIV-1 and human proteins, based on the hypothesis that proteins with similar interface scaffolds share similar interaction partners. UniAlign's ability in detecting functionally important structural similarities is utilized in an application to discover interactions between HIV-1 ENV protein (gp41 and gp120) and human proteins. Structural compatibility of an HIV-human interaction pairs are evaluated via geometric, biochemical, and evolutionary features and a prediction model is developed using a Support Vector Machine. We used the trained and tuned SVM classifier to discover potential novel HIV-1 interacting partners for human proteins. Many predicted interactions had significant literature support, and we modeled the novel 3D interacting complex for HIV-1 envelope gp120 and gp41 proteins. This provides the first model for prediction of interactions that can also generate a protein-protein 3D complex. The results of the HIV-human interaction study have discovered novel virus-host interactions as well as potential clinical targets for therapeutic intervention.

We leave re-implementation of UniAlign in a lower-level programming language to future work. In addition, we could also predict the protein interaction maps between all HIV-1 proteins and human proteins.

LIST OF REFERENCES

1. Hasegawa, H. and L. Holm, *Advances and pitfalls of protein structural alignment.* Curr Opin Struct Biol, 2009. **19**(3): p. 341-8.

2. Thompson, J.D., F. Plewniak, and O. Poch, *BAliBASE: a benchmark alignment database for the evaluation of multiple alignment programs.* Bioinformatics, 1999. **15**(1): p. 87-8.

3. Orengo, C.A., et al., *CATH--a hierarchic classification of protein domain structures.* Structure, 1997. **5**(8): p. 1093-108.

4. Sauder, J.M., J.W. Arthur, and R.L. Dunbrack, Jr., *Large-scale comparison of protein sequence alignment algorithms with structure alignments.* Proteins, 2000. **40**(1): p. 6-22.

5. Kolodny, R., P. Koehl, and M. Levitt, *Comprehensive evaluation of protein structure alignment methods: scoring by geometric measures.* J Mol Biol, 2005. **346**(4): p. 1173-88.

6. Holm, L. and C. Sander, *Protein structure comparison by alignment of distance matrices.* J Mol Biol, 1993. **233**(1): p. 123-38.

7. Shindyalov, I.N. and P.E. Bourne, *Protein structure alignment by incremental combinatorial extension (CE) of the optimal path.* Protein Eng, 1998. **11**(9): p. 739-47.

8. Zhang, Y. and J. Skolnick, *TM-align: a protein structure alignment algorithm based on the TM-score.* Nucleic Acids Res, 2005. **33**(7): p. 2302-9.

9. Ye, Y. and A. Godzik, *Flexible structure alignment by chaining aligned fragment pairs allowing twists.* Bioinformatics, 2003. **19 Suppl 2**: p. ii246-55.

10. Levitt, M. and M. Gerstein, *A unified statistical framework for sequence comparison and structure comparison.* Proc Natl Acad Sci U S A, 1998. **95**(11): p. 5913-20.

11. Jung, J. and B. Lee, *Protein structure alignment using environmental profiles.* Protein Engineering, 2000. **13**(8): p. 535-543.

12. Holm, L., et al., *Searching protein structure databases with DaliLite v.3.* Bioinformatics, 2008. **24**(23): p. 2780-1.

13. Mayr, G., F.S. Domingues, and P. Lackner, *Comparative analysis of protein structure alignments.* Bmc Structural Biology, 2007. **7**.

14. Kim, C. and B. Lee, *Accuracy of structure-based sequence alignment of automatic methods.* BMC Bioinformatics, 2007. **8**: p. 355.

15. Pirovano, W., K.A. Feenstra, and J. Heringa, *The meaning of alignment: lessons from structural diversity.* Bmc Bioinformatics, 2008. **9**.

16. Gerstein, M. and M. Levitt, *Comprehensive assessment of automatic structural alignment against a manual standard, the scop classification of proteins.* Protein Science, 1998. **7**(2): p. 445-456.

17. 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.

18. Yan, R., et al., *A comparative assessment and analysis of 20 representative sequence alignment methods for protein structure prediction.* Sci Rep, 2013. **3**: p. 2619.

19. Rost, B. and C. Sander, *Combining evolutionary information and neural networks to predict protein secondary structure.* Proteins, 1994. **19**(1): p. 55-72.

20. Ortiz, A.R., C.E.M. Strauss, and O. Olmea, *MAMMOTH (Matching molecular models obtained from theory): An automated method for model comparison.* Protein Science, 2002. **11**(11): p. 2606-2621.

21. Frankel, A.D. and J.A. Young, *HIV-1: fifteen proteins and an RNA.* Annu Rev Biochem, 1998. **67**: p. 1-25.

22. Valencia, A. and F. Pazos, *Prediction of protein-protein interactions from evolutionary information.* Methods Biochem Anal, 2003. **44**: p. 411-26.

23. Tastan, O., et al., *Prediction of interactions between HIV-1 and human proteins by information integration.* Pac Symp Biocomput, 2009: p. 516-27.

24. Evans, P., et al., *Prediction of HIV-1 virus-host protein interactions using virus and host sequence motifs.* BMC Med Genomics, 2009. **2**: p. 27.

25. Berman, H., K. Henrick, and H. Nakamura, *Announcing the worldwide Protein Data Bank.* Nat Struct Biol, 2003. **10**(12): p. 980.

26. Doolittle, J.M. and S.M. Gomez, *Structural similarity-based predictions of protein interactions between HIV-1 and Homo sapiens.* Virol J, 2010. **7**: p. 82.

27. Zhang, Q.C., et al., *Structure-based prediction of protein-protein interactions on a genome-wide scale.* Nature, 2012. **490**(7421): p. 556-60.

28. Zhao, C. and A. Sacan. *Prediction of HIV-1 and human protein interactions based on a novel evolution-aware structure alignment method.* in *Proceedings of the International Conference on Bioinformatics & Computational Biology (BIOCOMP)*. 2013.

29. Zhang, Q.C., et al., *Protein interface conservation across structure space.* Proc Natl Acad Sci U S A, 2010. **107**(24): p. 10896-901.

30. Baspinar, A., et al., *PRISM: a web server and repository for prediction of protein-protein interactions and modeling their 3D complexes.* Nucleic Acids Res, 2014.

31. Caffrey, D.R., et al., *Are protein-protein interfaces more conserved in sequence than the rest of the protein surface?* Protein Science, 2004. **13**(1): p. 190-202.

32. Neuvirth, H., R. Raz, and G. Schreiber, *ProMate: A structure based prediction program to identify the location of protein-protein binding sites.* Journal of Molecular Biology, 2004. **338**(1): p. 181-199.

33. Keskin, O., B.Y. Ma, and R. Nussinov, *Hot regions in protein-protein interactions: The organization and contribution of structurally conserved hot spot residues.* Journal of Molecular Biology, 2005. **345**(5): p. 1281-1294.

34. Keskin, O. and R. Nussinov, *Protein-protein interactions: Similar binding sites; Different partners.* Biophysical Journal, 2007: p. 222a-222a.

35. Kihara, D. and J. Skolnick, *The PDB is a covering set of small protein structures.* Journal of Molecular Biology, 2003. **334**(4): p. 793-802.

36. Siew, N., et al., *MaxSub: an automated measure for the assessment of protein structure prediction quality.* Bioinformatics, 2000. **16**(9): p. 776-85.

37. Zhang, Y. and J. Skolnick, *Scoring function for automated assessment of protein structure template quality.* Proteins, 2004. **57**(4): p. 702-10.

38. Yang, Y., et al., *A new size-independent score for pairwise protein structure alignment and its application to structure classification and nucleic-acid binding prediction.* Proteins, 2012. **80**(8): p. 2080-8.

39. Mizuguchi, K. and N. Go, *Seeking significance in three-dimensional protein structure comparisons.* Curr Opin Struct Biol, 1995. **5**(3): p. 377-82.

40. Damm, K.L. and H.A. Carlson, *Gaussian-weighted RMSD superposition of proteins: A structural comparison for flexible proteins and predicted protein structures.* Biophysical Journal, 2006. **90**(12): p. 4558-4573.

41. Daniels, N.M., S. Nadimpalli, and L.J. Cowen, *Formatt: Correcting protein multiple structural alignments by incorporating sequence alignment.* Bmc Bioinformatics, 2012. **13**.

42. Shatsky, M., R. Nussinov, and H.J. Wolfson, *Optimization of multiple-sequence alignment based on multiple-structure alignment.* Proteins-Structure Function and Bioinformatics, 2006. **62**(1): p. 209-217.

43. Wang, S., et al., *Protein structure alignment beyond spatial proximity.* Scientific Reports, 2013. **3**.

44. Valdar, W.S.J., *Scoring residue conservation.* Proteins-Structure Function and Bioinformatics, 2002. **48**(2): p. 227-241.

45. Sunyaev, S.R., et al., *PSIC: profile extraction from sequence alignments with position-specific counts of independent observations.* Protein Engineering, 1999. **12**(5): p. 387-394.

46. 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.

47. Wang, G.L. and R.L. Dunbrack, *PISCES: recent improvements to a PDB sequence culling server.* Nucleic Acids Research, 2005. **33**: p. W94-W98.

48. Li, W., et al., *PSI-Search: iterative HOE-reduced profile SSEARCH searching.* Bioinformatics, 2012. **28**(12): p. 1650-1.

49. Soding, J., A. Biegert, and A.N. Lupas, *The HHpred interactive server for protein homology detection and structure prediction.* Nucleic Acids Research, 2005. **33**: p. W244-W248.

50. Edgar, R.C., *MUSCLE: multiple sequence alignment with high accuracy and high throughput.* Nucleic Acids Research, 2004. **32**(5): p. 1792-1797.

51. Sander, C. and R. Schneider, *The Hssp Database of Protein-Structure Sequence Alignments.* Nucleic Acids Research, 1994. **22**(17): p. 3597-3599.

52. Henikoff, S. and J.G. Henikoff, *Position-Based Sequence Weights.* Journal of Molecular Biology, 1994. **243**(4): p. 574-578.

53. Vingron, M. and P. Argos, *A fast and sensitive multiple sequence alignment algorithm.* Comput Appl Biosci, 1989. **5**(2): p. 115-21.

54. Henikoff, J.G. and S. Henikoff, *Using substitution probabilities to improve position-specific scoring matrices.* Comput Appl Biosci, 1996. **12**(2): p. 135-43.

55. Marti-Renom, M.A., M.S. Madhusudhan, and A. Sali, *Alignment of protein sequences by their profiles.* Protein Science, 2004. **13**(4): p. 1071-1087.

56. 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.

57. Eddy, S.R., *Where did the BLOSUM62 alignment score matrix come from?* Nature Biotechnology, 2004. **22**(8): p. 1035-1036.

58. 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.

59. Shannon, C.E., *The mathematical theory of communication. 1963.* MD Comput, 1997. **14**(4): p. 306-17.

60. Sander, C. and R. Schneider, *Database of Homology-Derived Protein Structures and the Structural Meaning of Sequence Alignment.* Proteins-Structure Function and Genetics, 1991. **9**(1): p. 56-68.

61. Edgar, R.C. and K. Sjolander, *A comparison of scoring functions for protein sequence profile alignment.* Bioinformatics, 2004. **20**(8): p. 1301-1308.

62. Ma, B., et al., *Protein-protein interactions: structurally conserved residues distinguish between binding sites and exposed protein surfaces.* Proc Natl Acad Sci U S A, 2003. **100**(10): p. 5772-7.

63. 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.

64. Sacan, A., et al., *LFM-Pro: a tool for detecting significant local structural sites in proteins.* Bioinformatics, 2007. **23**(6): p. 709-16.

65. Pandit, S.B. and J. Skolnick, *Fr-TM-align: a new protein structural alignment method based on fragment alignments and the TM-score.* BMC Bioinformatics, 2008. **9**: p. 531.

66. Zemla, A., *LGA: A method for finding 3D similarities in protein structures.* Nucleic Acids Res, 2003. **31**(13): p. 3370-4.

67. Khazanov, N.A., et al., *Overcoming sequence misalignments with weighted structural superposition.* Proteins, 2012. **80**(11): p. 2523-35.

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

69. Durbin, R., *Biological sequence analysis : probabalistic models of proteins and nucleic acids*. 1998, Cambridge, UK New York: Cambridge University Press. xi, 356 p.

70. Marchler-Bauer, A., et al., *CDD: conserved domains and protein three-dimensional structure.* Nucleic Acids Res, 2013. **41**(Database issue): p. D348-52.

71. Chandonia, J.M., et al., *The ASTRAL Compendium in 2004.* Nucleic Acids Res, 2004. **32**(Database issue): p. D189-92.

72. Stebbings, L.A. and K. Mizuguchi, *HOMSTRAD: recent developments of the Homologous Protein Structure Alignment Database.* Nucleic Acids Res, 2004. **32**(Database issue): p. D203-7.

73. Thompson, J.D., et al., *BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark.* Proteins, 2005. **61**(1): p. 127-36.

74. Marchler-Bauer, A., et al., *CDD: a conserved domain database for interactive domain family analysis.* Nucleic Acids Res, 2007. **35**(Database issue): p. D237-40.

75. Andreeva, A., et al., *SISYPHUS - structural alignments for proteins with non-trivial relationships.* Nucleic Acids Research, 2007. **35**: p. D253-D259.

76. Andreeva, A., et al., *Data growth and its impact on the SCOP database: new developments.* Nucleic Acids Research, 2008. **36**: p. D419-D425.

77. Cuff, A.L., et al., *The CATH classification revisited-architectures reviewed and new ways to characterize structural divergence in superfamilies.* Nucleic Acids Research, 2009. **37**: p. D310-D314.

78. Csaba, G., F. Birzele, and R. Zimmer, *Protein structure alignment considering phenotypic plasticity.* Bioinformatics, 2008. **24**(16): p. I98-I104.

79. Nayeem, A., D. Sitkoff, and S. Krystek, Jr., *A comparative study of available software for high-accuracy homology modeling: from sequence alignments to structural models.* Protein Sci, 2006. **15**(4): p. 808-24.

80. Hanson, R.M., *Jmol - a paradigm shift in crystallographic visualization.* Journal of Applied Crystallography, 2010. **43**: p. 1250-1260.

81. Holm, L. and P. Rosenstrom, *Dali server: conservation mapping in 3D.* Nucleic Acids Res, 2010. **38**(Web Server issue): p. W545-9.

82. Baspinar, A., et al., *PRISM: a web server and repository for prediction of protein-protein interactions and modeling their 3D complexes.* Nucleic Acids Research, 2014. **42**(W1): p. W285-W289.

83. Martin, A.C., *Mapping PDB chains to UniProtKB entries.* Bioinformatics, 2005. **21**(23): p. 4297-301.

84. Bairoch, A., et al., *The Universal Protein Resource (UniProt).* Nucleic Acids Res, 2005. **33**(Database issue): p. D154-9.

85. Fu, W., et al., *Human immunodeficiency virus type 1, human protein interaction database at NCBI.* Nucleic Acids Research, 2009. **37**: p. D417-D422.

86. Prasad, T.S.K., et al., *Human Protein Reference Database-2009 update.* Nucleic Acids Research, 2009. **37**: p. D767-D772.

87. Cukuroglu, E., et al., *Non-Redundant Unique Interface Structures as Templates for Modeling Protein Interactions.* Plos One, 2014. **9**(1).

88. Stark, C., et al., *BioGRID: a general repository for interaction datasets.* Nucleic Acids Res, 2006. **34**(Database issue): p. D535-9.

89. Hermjakob, H., et al., *IntAct: an open source molecular interaction database.* Nucleic Acids Res, 2004. **32**(Database issue): p. D452-5.

90. Salwinski, L., et al., *The Database of Interacting Proteins: 2004 update.* Nucleic Acids Res, 2004. **32**(Database issue): p. D449-51.

91. Hubbard, S. and J. Thornton, *NACCESS. 1993.* Department of Biochemistry and Molecular Biology, University College London, 1993.

92. Tuncbag, N., et al., *Architectures and functional coverage of protein-protein interfaces.* Journal of Molecular Biology, 2008. **381**(3): p. 785-802.

93. Keskin, O. and R. Nussinov, *Favorable scaffolds: proteins with different sequence, structure and function may associate in similar ways.* Protein Engineering Design & Selection, 2005. **18**(1): p. 11-24.

94. Cukuroglu, E., et al., *Non-redundant unique interface structures as templates for modeling protein interactions.* PLoS One, 2014. **9**(1): p. e86738.

95. Zhao, C. and A. Sacan, *UniAlign: protein structure alignment meets evolution.* Bioinformatics, 2015. **31**(19): p. 3139-46.

96. Gao, M. and J. Skolnick, *Structural space of protein-protein interfaces is degenerate, close to complete, and highly connected.* Proceedings of the National Academy of Sciences of the United States of America, 2010. **107**(52): p. 22517-22522.

97. Brown, G., et al., *Conditional Likelihood Maximisation: A Unifying Framework for Information Theoretic Feature Selection.* Journal of Machine Learning Research, 2012. **13**: p. 27-66.

98. Keskin, O., et al., *Principles of protein-protein interactions: What are the preferred ways for proteins to interact?* Chemical Reviews, 2008. **108**(4): p. 1225-1244.

99. Guendel, I., et al., *Role of Bruton's tyrosine kinase inhibitors in HIV-1-infected cells.* J Neurovirol, 2015. **21**(3): p. 257-75.

100. Osinusi, A., et al., *ITPA gene polymorphisms significantly affect hemoglobin decline and treatment outcomes in patients coinfected with HIV and HCV.* Journal of Medical Virology, 2012. **84**(7): p. 1106-1114.

101. Jelicic, K., et al., *The HIV-1 envelope protein gp120 impairs B cell proliferation by inducing TGF-beta1 production and FcRL4 expression.* Nat Immunol, 2013. **14**(12): p. 1256-65.

102. Anand, A.R. and R.K. Ganju, *HIV-1 gp120-mediated apoptosis of T cells is regulated by the membrane tyrosine phosphatase CD45.* Journal of Biological Chemistry, 2006. **281**(18): p. 12289-12299.

103. Joshi, A., K. Nagashima, and E.O. Freed, *Defects in cellular sorting and retroviral assembly induced by GGA overexpression.* Bmc Cell Biology, 2009. **10**.

104. Chu, H., J.J. Wang, and P. Spearman, *Human Immunodeficiency Virus Type-1 Gag and Host Vesicular Trafficking Pathways.* Hiv Interactions with Host Cell Proteins, 2009. **339**: p. 67-84.

105. Caly, L., et al., *Impaired nuclear import and viral incorporation of Vpr derived from a HIV long-term non-progressor.* Retrovirology, 2008. **5**.

Appendix A

Dependence of alignment accuracy on shift error, in BAliBASE AND HOMSTRAD

The results are similar to those shown in the manuscript for CDD dataset. For each method, there is a jump in accuracy when a shift error of 1 residue is allowed. DaliLite performs poorly compared to other methods in HOMSTRAD database.

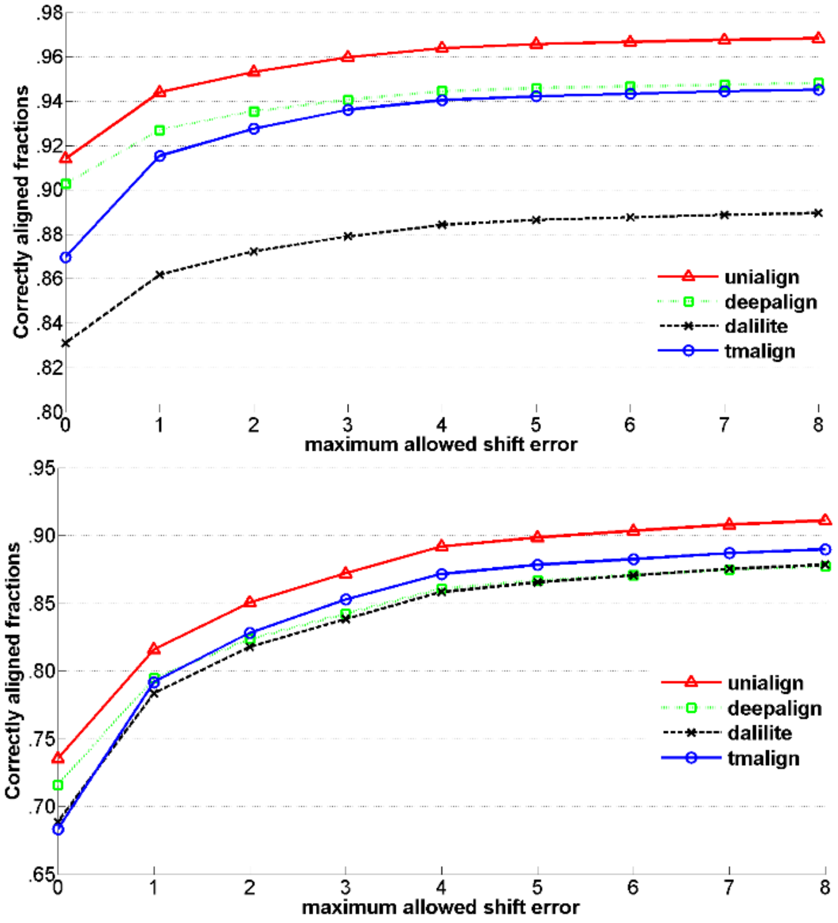


Figure 26 Top: Average *fcar* of HOMSTRAD alignments of different methods as a function of the shift error tolerance level δ. The y-axis starts from 0.80 to 0.98. Bottom: Average *fcar* of BAliBASE alignments of different methods as a function of the shift error tolerance level δ. The y-axis starts from 0.65 to 0.95.

Dependence of alignment accuracy on level of homology, in BAliBASE AND HOMSTRAD

For the HOMSTRAD database, the accuracy of the methods increase with increasing sequence identity level. This is similar to that observed for CDD database. For BAliBASE database, however, higher sequence identity level does not mean higher structure alignment accuracy. UniAlign provides the most robust results across different homology levels.

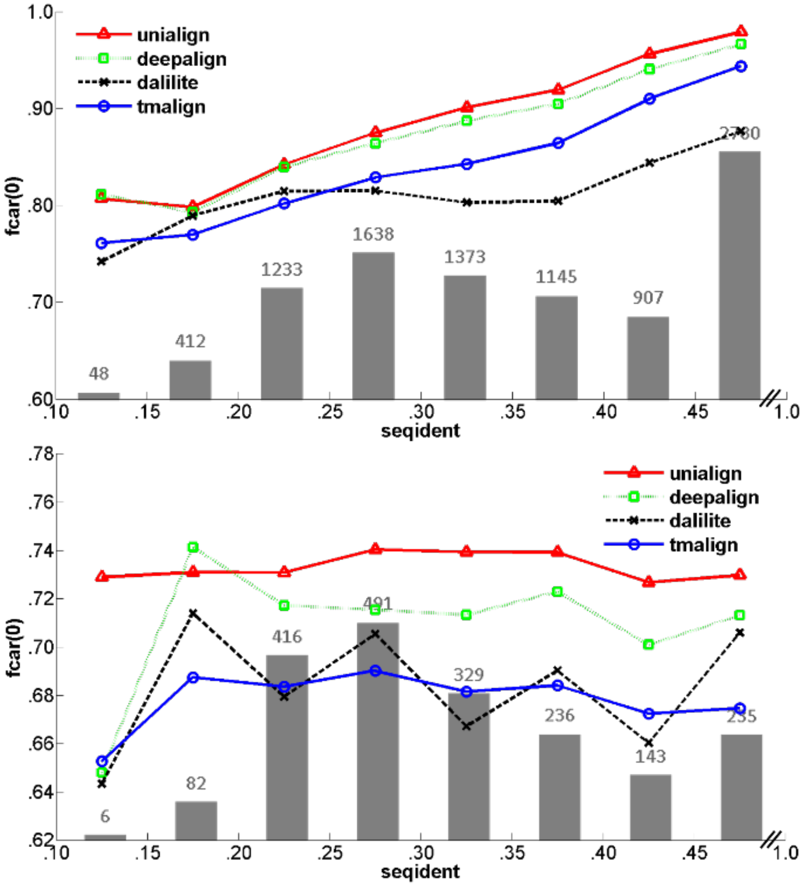


Figure 27 Dependence of alignment accuracy on the level of homology of the proteins from the HOMSTRAD and BAliBASE dataset. Alignments were grouped into sequence identity bins of 5% width. Line plots show the average *fcar*$_0$ values of various methods, whereas the histogram shows the number of alignments in each bin. Left: HOMSTRAD; Right: BAliBASE.

Dependence of alignment accuracy on structure similarity, in BAliBASE AND HOMSTRAD

UniAlign performs better than other methods across all geometric similarity levels. In BAliBASE, the geometric similarity (of the reference alignment) is not correlated with the performance of the structure alignment methods. Note that BAliBASE was developed for assessment of multiple sequence alignment methods whereas HOMSTRAD was designed for assessment of structure alignment methods.

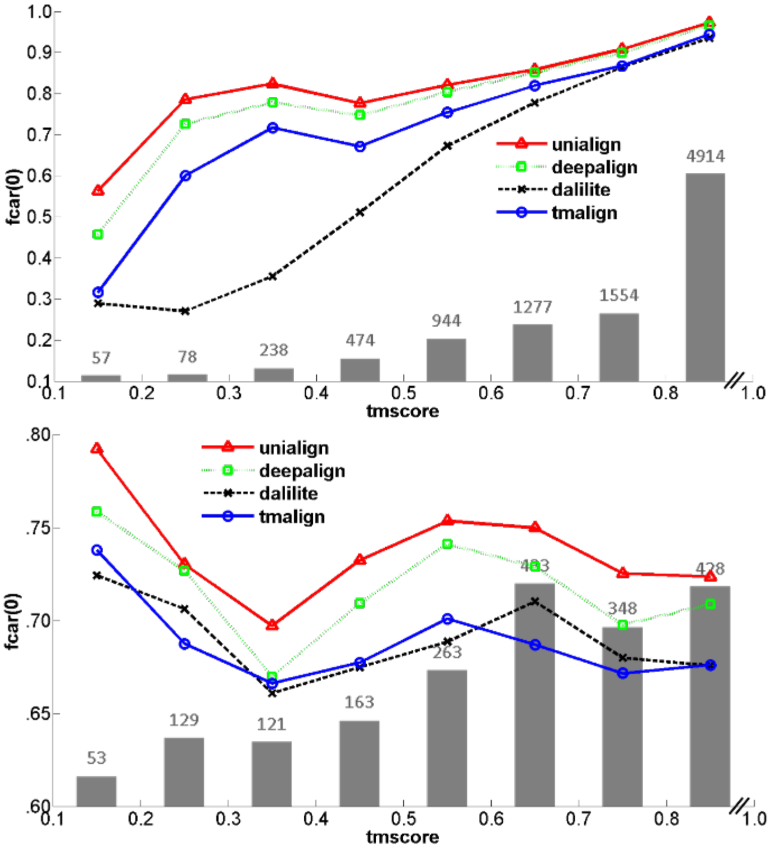


Figure 28 Dependence of alignment accuracy on the level of structural similarity of the proteins from the HOMSTRAD and BAliBASE dataset. Structural similarity is measured by the TMscore of the superposition generated from the reference alignments. Proteins are grouped into structural similarity bins of size 0.1. Line plots show the average *fcar*$_0$ values of different methods, whereas the histogram shows the number of alignments in each bin. Left: HOMSTRAD; Right: BAliBASE.

Appendix B

Performance on CDD dataset with the training data excluded.

The performance results on the CDD test dataset (excluding the pairs of proteins used in training UniAlign parameters) are shown below in Table 11. These results are nearly identical to those presented for the entire CDD dataset in the main manuscript.

Table 11 Comparison of the performance of four structure alignment methods on the CDD dataset, with the protein pairs used in training of UniAlign excluded. The best performance values are shown in bold. The scores of the reference database alignments are also shown in bold, when it is better than the best value from the structure alignment methods.

Method	$fcar_0$	UNI	GEO	SSE	SEQ	PRO	CON
UniAlign	**93.97%**	**2.097**	0.684	0.142	**0.288**	**0.168**	**0.054**
Deepalign	91.76%	0.996	0.654	**0.151**	0.270	0.127	0.045
DaliLite	92.34%	2.007	0.663	0.141	0.100	0.046	0.041
TMalign	85.49%	2.053	**0.685**	0.143	0.054	0.004	0.038
CDD core	**100.0%**	1.089	0.341	**0.232**	**0.695**	**0.572**	**0.071**

Appendix C

Training Dataset

A subset of the CDD dataset was used for empirical determination of the parameters involved in UniAlign. The training dataset contained the following 45 pairs of reference alignments:: 1a1v A:190-325 and 1d9x A:2-414; 1a1m A:182-278 and 1mfa L:1-111; 1a02 N:577-678 and 1cyg 492-574; 1a0f A:81-201 and 1rk4 B:92-234; 1a36 A:431-633; A:713-765 and 1aihA; 1a2kA and 1e3vB; 1a5z 22-163 and 1mld A:1-144; 12asA and 1jjcA; 1a4mA and 1k6w A:56-375; 1ad4A and 1f6yA; 1adj A:326-421 and 1b76 A:395-505; 1ami 2-528 and 1l5j A:373-862; 1aosB and 1c3uA; 1aoxA and 1jey A:34-253; 1apxA and 1bgp; 1ash and 1dlyA; 1attB and 1c5gA; 1ay7A and 1brsB; 1b0uA and 1e3m A:567-800; 1b34A and 1b34B; 1b55A and 1eazA; 1b57A and 1gvfA; 1b6a 110-374; 449-478 and 1chm B:157-402; 1b71 A:1-147 and 1bcfA; 1b7aA and 1fjjA; 1bfd 342-524 and 1jsc A:461-648; 1bk7A and 1bolA; 1bmo A:78-135 and 1cgjI; 1bp1 1-217 and 1bp1 218-456; 1bqg 144-422 and 1f9c B:131-372; 1brfA and 1ryt 148-191; 1bt0A and 1c1yB; 1bu8 A:337-449 and 1ca1 250-370; 1buc A:233-383 and 1ege A:242-396; 1bx4A and 1gc5A; 1c5fA and 1cwaA; 1c8u A:2-115 and 1iq6A; 1c9kC and 1g64A; 1ce8 A:936-1073 and 1eghC; 1cg1A and 1dai; 1cktA and 1gt0D; 1cm9A and 1f2lA; 1cnx and 1keqA; 1d1wA and 1m7vA; 1dcpA and 1usoB; 1dd8 A:1-253 and 1qfl A:4-268; 1deoA and 1esc; 1dfx 37-125 and 1dqiA; 1dgpA and 1ezfA; 1dj0A and 1k8w A:9-250; 1dk4A and 1ni9A; 1dmwA and 1ltzA; 1dp2 A:1-149 and 1gmxA; 1dqa A:587-703 and 1qax A:111-220; 1ds7A and 1nox; 1dt4A and 1e3p A:579-634; 1dypA and 1gbg; 1e2uA and 1jqkA; 1ef7A and 1gcb; 1efpB and 1gpm A:208-404; 1egwA and 1mnmB; 1eiy B:39-151 and 1gd7D;

1ekfA and 1et0A; 1ekjA and 1g5cA; 1ep2A and 1o95 B:1-340; 1eu1 A:626-780 and 1ogy A:682-801; 1eu1 A:4-625 and 1fdo 1-564; 1euaA and 1f05A; 1excA and 1k7kA; 1exi A:3-120 and 1jbgA; 1ezrA and 1hp0A; 1fc7 A:157-248 and 1kwaA; 1fftC and 1m56C; 1ft2B and 1ghqA; 1fxkA and 1fxkC; 1g13A and 1ktjA; 1g20B and 1mioA; 1g3uA and 1gky; 1h16A and 1hk8A; 1hh2 P:277-344 and 1hh2 P:199-276; 1im5A and 1nf8A; 1jey A:254-534 and 1jey B:242-545; 1jw9B and 1ngvA; 1k3c A:228-540 and 1ko7 A:130-298; 1kutA and 1obgA; 1kzhA and 1pfkA; 1l0vC and 1l0vD; 1nb8B and 1vjvA; 1o13A and 1p90A; 1qmh B:5-184; B:280-339 and 1uae; 1rybA and 2pth.

Appendix D

Running Time Analysis

The average running times on the CDD database for different methods is shown below in Table 12. Except for DaliLite, the methods were executed on a PC with Intel Xeon CPU @ 2.9 GHz, with 256GB memory. DaliLite was executed through the online service available at: http://www.ebi.ac.uk/Tools/services/rest/dalilite/run/.

Table 12 Average running times for different structure alignment methods, on the CDD database of pairwise alignments. The DaliLite running times were determined as the time it took for the web service to return results. These running times are dependent on the implementation details and the running environments and should not be used as a measure of computational efficiency.

Method	Running time (sec)
Deepalign	0.45
TMalign	0.57
DaliLite web service	13.76
UniAlign	942

Notice that UniAlign requires significantly more time than the other methods. As seen in the break-down times for different stages of UniAlign (Table 13), 87% of this time is due to the homology search using PSI-BLAST. The time requirements can be reduced significantly, by 95%, if we pre-calculate per-protein evolutionary and other information.

The main UniAlign algorithm takes 49 seconds, which is still significantly higher than Deepalign and TMalign. We attribute the higher running time requirements to the programming language used in the current implementation of UniAlign. UniAlign was implemented in the Matlab whereas TMalign and Deepalign were written in Fortran and C++, respectively. Matlab provides a rapid prototyping environment but is known to require more computing time than Fortran and C++. We leave re-implementation of UniAlign in a lower-level programming language to future work.

Table 13 Average running times for different stages of UniAlign

Computation	Running time (sec)
PSI-BLAST search	819.828
Construction of MSA, calculation of evolutionary scores	58.194
Calculation of other sequence and secondary structure information	15.352
UniAlign initial alignment and iterative optimization steps, with cached per-protein information	49.3

Curriculum Vitae

EDUCATION

Drexel University, Philadelphia, PA
Ph.D. Candidate in Biomedical Engineering
2011 - 2016

Dalian University of Technology, China
M.S. in Biomedical Engineering
2009 - 2011

Dalian University of Technology, China
B.S. in Electrical Engineering
2005 - 2009

JOURNAL PUBLICATIONS

1. **Zhao C**, Saçan A. UniAlign: protein structure alignment meets evolution. **Bioinformatics** (5-yr impact factor: 8.136). 2015 Jun 9. pii: btv354.
2. Yang H, **Zhao C**, Saçan A. Unfolding the Protein Surface for Pattern Matching. Bioinformatics (submitted).
3. Bell F, **Zhao C**, Saçan A. PDBCirclePlot: a novel visualization method for protein structures. preprint arXiv:1402.5323

CONFERENCE PRESENTATION

1. **Zhao C**, Saçan A. Prediction of HIV-1 and human protein interactions based on a novel evolution-aware structure alignment method. *International Conference on Bioinformatics & Computational Biology (BIOCOMP)*. 2013.
2. **Zhao C**, Qiu T. An automatic ocular artifacts removal method based on wavelet-enhanced canonical correlation analysis. *International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 2011.

CONFERENCE POSTER

1. **Zhao C**, Saçan A. Protein Structure Alignment meets Evolution. *International Conference on Intelligent Systems for Molecular Biology (ISMB).* 2014.
2. **Zhao C**, Song B, Yang H, Saçan A. Prediction of dihedral angles and 3D structure of proteins using neural network. *International Symposium on Bioinformatics Research and Applications (ISBRA)*. 2013.

WORKSHOP

1. University of Pennsylvania RNA-Seq Workshop for the Bioinformatician. 2015 Jun 23rd
2. Teaching Metagenomics. Drexel. 2015 Aug 14th

EXPERIENCE

Graduate Research Assistant, Drexel University, 09/2011 - present

- ***UniAlign: protein structure alignment meets evolution**, 04/2012 - 04/2015*
  - ✓ Designed and implemented a novel protein structural alignment algorithm UniAlign (available on: http://sacan.biomed.drexel.edu/unialign), systematically incorporating evolutionary information.
  - ✓ Published on Bioinformatics (5-Yr impact factor: 8.136). Primarily a *Matlab*-based implementation
- ***Prediction of protein-protein interaction between HIV-1 and human**, 01/2013 - present*
  - ✓ Invented the workflow of PPI prediction based on structural similarity and further constructed the interaction network between HIV-1 and human. Technology was mostly *Python* and *SQL*.
- ***Feature selection for 16S rRNA metagenomics analysis***, 09/2014 - 01/2015
  - ✓ Performed 16S NGS analysis to explore human gut microbiome composition using *QIIME* package.
  - ✓ Analyzed the most informative OTUs related to the phenotype DIET. Work was done in *Python*, *Bash* and high performance cluster *Proteus*.

Graduate Teaching Assistant, Drexel University, 09/2011 - present

- Calhoun Fellowship
- Led the lab sections for Matlab programming and bioinformatics classes, including bio-computational languages, computational bioengineering, quantitative system biology and advanced bioinformatics.

Genomic Data Science Specialization Certification, Johns Hopkins University, 09/ 2015 - 01/2016

- ***Accomplished the specialization with Distinction, certifications available on LinkedIn: https:www.linkedin.com/in/chunyuzhao.***

Bioinformatics Specialization Certification, University of California, San Diego, 08/ 2015 - present

- ***Accomplished five classes (hacker track) with Distinction, certification available on LinkedIn: https:www.linkedin.com/in/chunyuzhao.***
  - ✓ Implemented algorithms and data structures to solve various bioinformatics problems, such as frequency array, greedy motif search, de Bruijn graph, brute force algorithms for sequencing antibiotics, dynamic programming, breakpoint graphs, evolutionary tree reconstruction, computational proteomics, approximate pattern matching and hidden markov model.
- ***Python*** **implementation available on Github: https://github.com/aprilchunyuzhao/BioinformaticsFromCoursera**

TECHNICAL SKILLS

- **Programming Languages**: Python, Java, Matlab, R, Bash shell scripting, MySQL.
- **Expertise in** utilizing bioinformatics tools: Galaxy, Bioconductor, and bioinformatics resources: NCBI and UCSC genome browser.
- **Experienced in** building and maintaining relational databases and utilizing HPC cluster.
- **Experienced in** microarray, DNA-Seq, RNA-Seq, Chip-Seq, metagenomics, and comparative genomic analysis.