Bioinformatics, 37(4), 2021, 490–496
doi: 10.1093/bioinformatics/btaa810
Advance Access Publication Date: 21 September 2020
Original Paper

Structural bioinformatics

A novel sequence alignment algorithm based on deep learning of the protein folding code

Mu Gao * and Jeffrey Skolnick*

Center for the Study of Systems Biology, School of Biological Sciences, Georgia Institute of Technology, Atlanta, GA 30332, USA

*To whom correspondence should be addressed.

Associate Editor: Arne Elofsson

Received on June 24, 2020; revised on August 11, 2020; editorial decision on September 5, 2020; accepted on September 8, 2020

Abstract

Motivation: From evolutionary interference, function annotation to structural prediction, protein sequence comparison has provided crucial biological insights. While many sequence alignment algorithms have been developed, existing approaches often cannot detect hidden structural relationships in the 'twilight zone' of low sequence identity. To address this critical problem, we introduce a computational algorithm that performs protein *Sequence Alignments from deep-Learning of Structural Alignments* (SAdLSA, silent 'd'). The key idea is to implicitly learn the protein folding code from many thousands of structural alignments using experimentally determined protein structures.

Results: To demonstrate that the folding code was learned, we first show that SAdLSA trained on pure α-helical proteins successfully recognizes pairs of structurally related pure β-sheet protein domains. Subsequent training and benchmarking on larger, highly challenging datasets show significant improvement over established approaches. For challenging cases, SAdLSA is ∼150% better than HHsearch for generating pairwise alignments and ∼50% better for identifying the proteins with the best alignments in a sequence library. The time complexity of SAdLSA is O(N) thanks to GPU acceleration.

Availability and implementation: Datasets and source codes of SAdLSA are available free of charge for academic users at http://sites.gatech.edu/cssb/sadlsa/.

Contact: mu.gao@gatech.edu or skolnick@gatech.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Protein sequence comparison and alignment are an essential component of computational biology. While the classic BLAST algorithm is an efficient heuristic approach (Altschul, 1997), it often fails to detect subtle, yet significant, similarity for sequences within the 'twilight zone' of sequence identity, where <30% of aligned residues are identical (Muller *et al.*, 1999; Rost, 1999). To address this issue, profile–profile-based sequence comparison methods take the advantage of the fact that many sequence variations arise from divergent evolution within the same protein family (Eddy *et al.*, 1995; Sadreyev and Grishin, 2003; Yona and Levitt, 2002). Among the best are algorithms based on Hidden Markov Models (HMMs; Eddy *et al.*, 1995; Soding, 2005; Steinegger *et al.*, 2019), e.g. the widely adopted HHsearch approach (Steinegger *et al.*, 2019). Although they are more sensitive than BLAST, the issue of identifying sequence and structural similarity in the twilight zone remains.

It is well known that the three-dimensional structures of proteins are more conserved than their sequences (Chothia and Lesk, 1986). Often, the structural comparison of two protein domains reveals that they belong to the same structural fold, but their structural similarity is not obvious from their corresponding sequence comparison (Holm and Sander, 1996). Thus, compared to protein sequence space, protein fold space is far more compact. One main reason for this observation is that physical constraints limit the number of possible protein folds to about ∼1000 for individual domains (Zhang *et al.*, 2006). The underlying physical principles not only allow for maintenance of the same fold due to divergent evolution, where diminished sequence similarity is beyond the sensitivity of a sequence comparison algorithm, but also result in similar fold topologies being independently discovered by convergent evolution. Either way, it is highly desirable to develop a more sensitive sequence alignment algorithm that can provide insights about the structural relationships between proteins without prior knowledge of their corresponding structures and whether or not the proteins are in fact evolutionarily related.

As a step toward achieving this long-sought goal, we present SAdLSA (*Sequence Alignments from deep-Learning of Structural Alignments*), a novel deep-learning-based approach that performs very sensitive sequence alignments by deep learning (DL) of the protein folding code from tens of thousands of protein structural alignments. The goal of SAdLSA is to reproduce protein structural alignments when only the sequences (and not their corresponding native structures) are known. Although for benchmarking purposes, we used the experimental structures and our evaluation procedure is somewhat similar to 'threading' (e.g. Skolnick *et al.*, 2004), it is important to note that unlike threading, the application of SAdLSA does not require any structural information as input.

2 Materials and methods

As shown in Figure 1, the main component of SAdLSA is a fully convolutional neural network consisting of multiple residual blocks (He *et al.*, 2016). The network takes sequence profiles generated from the classic sequence search algorithms as input, and then learns the residue–residue distances obtained by optimal structural superposition (Zhang and Skolnick, 2005) on the two structures encoded by their respective sequences. The learning focuses on cross-sequence residue distances between pairs of proteins displayed in the structural alignment. This idea was inspired by, but is fundamentally different from, intra-sequence residue distance prediction designed for protein structural prediction (Senior *et al.*, 2020; Xu, 2019). The latter aims to predict the distance relationships between amino acids *within the same sequence*, whereas our objective is to infer structural relationships *between the pairs of sequences subjected to direct comparison*. For each pair of residues from the two respective sequences, the DL network outputs a probability distribution of their residue–residue distance in 21 bins ranging from 0 to 20 Å, at 1 Å spacing, with the last bin for pair distances >20 Å. Finally, a dynamic programming (DP) algorithm provides an optimal sequence alignment according to the predicted residue–residue distance matrix.

2.1 Deep-learning neural network model

The architecture of the DL neural network is shown in Figure 1. The inputs to the network are two sequence profiles generated from PSI-BLAST version 2.2.26 (Altschul, 1997) or HH-suite version 3.0 (Steinegger *et al.*, 2019), each of dimension $N_k \times 20$, where N_k is the length of the kth sequence ($k = 1, 2$), and the 20 columns represent 20 different amino acids at each residue position. The outer product of these two one-dimensional features is then performed to form a two-dimensional (2D) feature matrix, where at position (i, j) of the matrix the elements are a concatenation of the 20 columns formed from the ith residue of sequence 1 and the jth residue of sequence 2. Given these 2D features, the purpose of this neural network model is to predict a matrix of alignment distances for the two input sequences. To accomplish this objective, we employ a fully convolutional residual neural network consisting of up to 80 convolutional layers. Each residual block consists of two 2D convolutionary layers, two rectifier layers and an addition layer. In a residual block, the inputs x_i are transformed into $F(x_i) + x_i$ prior to the second activation (see Fig. 1), where $F(x_i)$ is the residual function (He *et al.*, 2016). If the new convolutional layers after x_i reduce the training error, $F(x_i)$ should deliver a meaningful value; otherwise, then it is approximated by zero. Thus, residual blocks provide an effective way to train a deep neural network.

In our implementation, each convolutional layer is composed of 64 filters with a kernel size of 3×3 or 5×5. If an input sequence is longer than 400 residues in length, to prevent memory overflow, we randomly sample up to 400 residue segments. After the convolutionary blocks, the outputs are transposed and averaged prior to the last 2D convolutionary layer that outputs 22 channels, representing 21 distance bins (1–20 at 1 Å each, >20 Å, and ignored pairs). The latter is reserved for missing labels caused by gap residues in a protein's structure. Finally, a softmax layer is employed for calculating the probability scores for each distance bin (Bishop, 2006). For training, we employed the cross-entropy as the loss function. Our network model takes proteins of variable lengths. The SAdLSA alignment distance prediction module is implemented using TensorFlow (Abadi *et al.*, 2016). The training distance labels are created from structural alignments by APoc (Gao and Skolnick, 2013). The global alignment algorithm of APoc is essentially an improved version of TM-align (Zhang and Skolnick, 2005).

The direct output from a deep-learning model is a matrix of probabilities of distance bins. For the purpose of DP, we convert this probability matrix into a mean distance matrix D, whose element $d_{ij} = \sum_{k=1}^{n} p_{ij}^k b_k$, where i and j are the target/template sequence positions, p_{ij}^k is the probability for bin k at position (i, j), b_k is distance

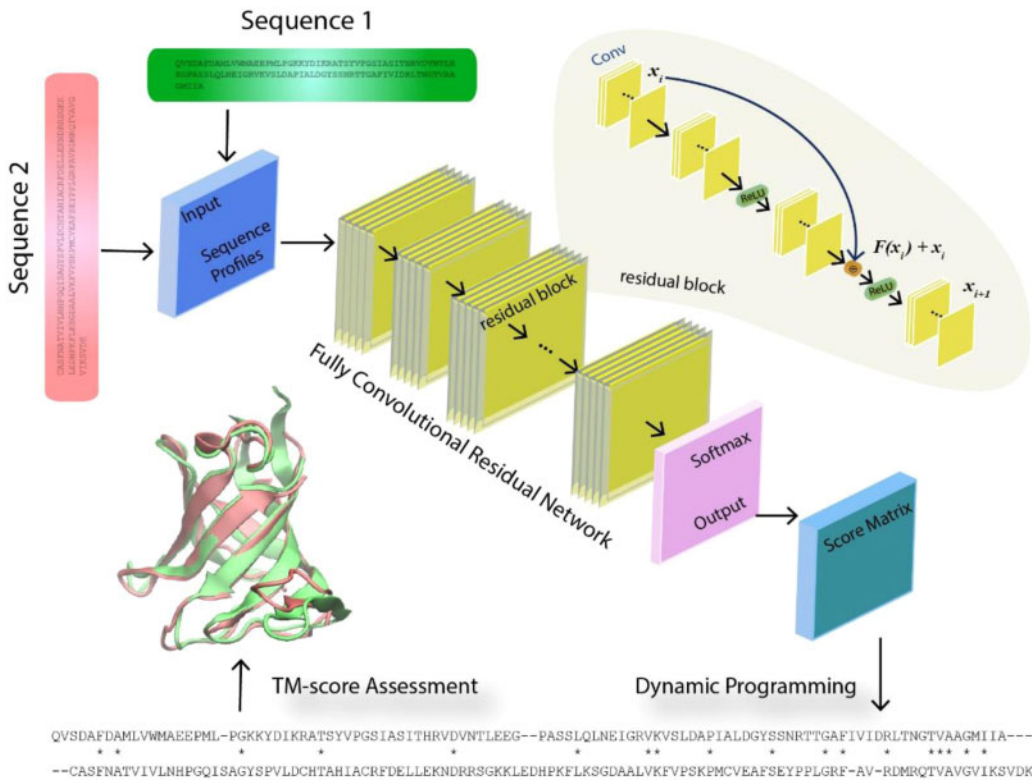

Fig. 1. Flowchart of SAdLSA. The inputs are two sequence profiles fed into a fully convolutional residual neural network. The output of the network is a matrix of predicted structural alignment distances, which are then employed by a DP procedure to obtain the optimal sequence alignment. In benchmark tests, the quality of the sequence alignment is assessed by the TM-score (Zhang and Skolnick, 2004)

labels from the sequence (1, 2, ... 20, 22). D is subsequently used as the scoring matrix to obtain the optimal alignment using a Smith–Waterman like DP algorithm (Smith and Waterman, 1981). Specifically, the scoring matrix for DP is $d_0 - D$, where d_0 is set at 16, and both gap-opening and extension penalties are set at −2. These parameters are empirically selected and have not been optimized. The distance matrix D is also used to calculate an estimated TM-score (Zhang and Skolnick, 2004) for the alignment (see Section 2.6). For each network model, we perform five training sessions and use the average of outputs of these five models to obtain a consensus distance matrix.

2.2 Pure α and pure β datasets

For proof-of-concept, two datasets of protein domains are curated from the SCOP database v2.07e composed of 5884 domains at 30% sequence identity (Fox *et al.*, 2014). The training set consists of 695 pure α-helical structures. The α-helical protein training set must satisfy two conditions: (i) each protein is categized as an all-α structure by SCOP and (ii) the protein cannot have any β-strand residues according to DSSP (Kabsch and Sander, 1983). Of these, we obtain 12 534 α-structure pairs that have a TM-score >0.4 as the training pairs. We use them to train our deep-learning models that predict the residue–residue distance at the optimal structural alignment. Similar criteria were applied to curate the pure β-sheet structures, except that we substitute β-sheet structures for α-helical structures. This yields a testing set of 121 pure β-sheet protein structures. In the test, for each β-sheet sequence, we perform a systematic scan with SAdLSA on the template structure library of 5884 SCOP domains, which includes ∼1000 SCOP folds and the maximum size of each fold family is limited to 50 entries. Note that the template library contains all fold classes.

2.3 SCOP10 testing datasets

Testing sets are curated from 2964 SCOP domain cluster representatives at 10% sequence identity. To prevent bias, we limit the maximum number of entries for each fold to 10. From these ∼3k entries, we randomly select 593 domain sequences as the main testing set. The remainder serves as the template library. From it, we derived two benchmark tests: (i) PAIR947 for assessing pairwise alignment quality given the sequence pairs and (ii) LIB593 for searching a sequence library to find the sequences with the closest matching pairs of structure. To derive PAIR947, for each of the 593 testing sequences, we randomly select up to two template entries to form pairs with the query, if they have a TM-score >0.45 in the structural alignment. This procedure gives the 947 target/template pairs as the pairwise alignment testing set. In the LIB593 test, we search the full SCOP10 sequence library for each of the 593 target sequences.

2.4 SCOP30 training/validation datasets

The 5884 SCOP domain representatives are employed to derive the training and validation datasets. First, we remove all entries in the 593 SCOP10 domain test set from the SCOP30 domain set and also remove all the homologs if they share a BLAST e-value <0.1 with any entry in the testing set. We perform structural alignments with APoc on the remaining entries in the SCOP30 set and retain all pairs with a TM-score >0.4. This procedure yields about 79k protein pairs. We then randomly split them into five pairs of training/validation sets, with each validation set containing 5k entries and remaining proteins comprise the training set. The validation cases are used primarily to prevent overfitting and to select appropriate hyper-parameters, including the learning rate, momentum and the weight decay parameter for the L2 regularization. We did not perform extensive brute-force search for the best hyper-parameters, but rather selected a set of reasonably good parameters from hundreds of training runs.

2.5 Sequence profiling

PSI-BLAST version 2.2.26 is employed to derive the sequence profile from the UniProt Ref90 sequence database released on April 25, 2018 (Wu, 2006). HH-suite version 3.0 is employed with a specially curated sequence database Uniclust30 released on October, 2017 (Steinegger *et al.*, 2019). We first run HHblits on this sequence library to derive the appropriate HMM model for each target and template sequence in our benchmark tests. Using these HMM models, we run HHsearch for searching hits in a sequence library or run HHalign for pairwise alignment. In all HHsearch or HHalign runs, we fully enabled the MAC algorithm by using mact = 0. This procedure yields the best TM-score for the domain's sequence alignment. In the benchmark test on α/β sets, we employed PSI-BLAST profiles as the training features to SAdLSA, and in the SCOP10/30 benchmark tests, we employed HHblits profiles as the training features.

2.6 Alignment assessment and ranking

Since we benchmark SAdLSA on protein sequences with experimentally determined structures, this allows us to build structural models using the sequence alignment and then assess the quality of the sequence alignment using the TM-score, ranging from 0 to 1, a protein length-independent metric where a TM-score >0.4 indicates a statistically significant alignment (Zhang and Skolnick, 2004). For evaluation, given an alignment between a target sequence and a template sequence, we build a structural model for the target sequence by copying coordinates from the experimental structure of the corresponding template. This structural model is then compared with the experimental (native) structure of the target by the program TM-score (Zhang and Skolnick, 2004). In order to evaluate the quality of the template, we also perform direct structural alignments using experimental structures of both the target and template with APoc (Gao and Skolnick, 2013), which yields a TM-score from an optimized structural alignment.

SAdLSA primarily ranks the sequence alignment by a predicted TM-score, which can be used for both pairwise comparison and library searches. Given an alignment, we employ the distance matrix D defined in Section 2.1 to predict the TM-score of the sequence alignment. Specifically, we use $t_i = d_i - c$, where i is the alignment position, d_i is taken from the matrix D along the optimal path from DP, and c is an offset constant set at 1. The matrix of mean distance immediately yields,

$$\text{Predicted TM-score} \equiv \left[\sum_{i=1}^{N} \frac{1}{\left(1+(t_i/d_0)^2\right)}\right]/L_T \qquad (1)$$

where N is the length of the sequence alignment, L_T is the length of the target sequence, t_i is the mean distance taken from the distance matrix and $d_0 = 1.24\sqrt[3]{L_T - 15} - 1.8$.

Alternatively, when searching a sequence library, the alignment score s from DP may be used for ranking hits. In addition, the statistical significance can be estimated from the Z-score $= (s - \bar{s})/\sigma$, where $\bar{s}$ and σ denote the mean and standard deviation, respectively.

3 Results

We describe three different tests. In the first test, we demonstrate that the deep-learning model is transferrable between different fold classes. The second test shows the improvement in pairwise sequence alignments. The third test examines the performance of SAdLSA in searching hits from a sequence library.

3.1 Aligning sequence encoding β-structures from learning sequences encoding α-structures

As with any machine-learning approach, one major concern is the transferability of the learned models to new cases. To address this, we first provide a proof-of-concept illustration using 12k pairwise structural alignments generated by comparing 695 α-helical protein domains as the training set (see Section 2.2). Our test set is the sequences of 121 β-sheet domains; for each target, we search for its best hit in ∼6000 SCOP sequences encoding ∼1000-folds including

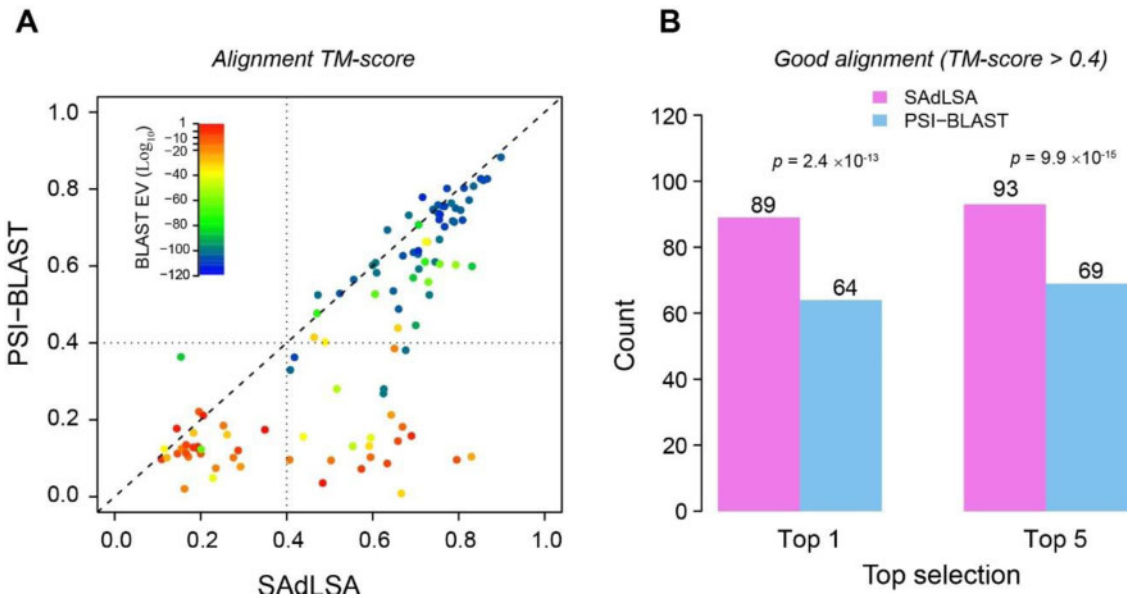

Fig. 2. A proof-of-concept demonstration of SAdLSA. (**A**) Head-to-head comparison between PSI-BLAST and SAdLSA on 121 sequences encoding β-sheet structures. The training set of SAdLSA is composed of 695 sequences encoded for α-helical structures only. Each circle represents the top-ranked sequence hit for each β-sheet protein obtained by applying each method to scan a sequence library of ∼6000 SCOP protein domain sequences that share no >30% sequence identity to the query sequences and that include ∼1000-fold types. The color scale is according to the PSI-BLAST e-value. (**B**) Statistics of good sequence alignment hits with a TM-score >0.4 for the top 1 and the best of top 5 ranked hits. The *P*-value is calculated on paired TM-scores using the Student *t*-test

all 4-fold classes (α, β, α/β and α + β) that share <30% sequence identity among themselves and the target sequences (Fox *et al.*, 2014). Thus, we examine the ability to identify sequences encoding β-sheet domains with DL models trained on structural alignments of α-helical proteins. The goal is to examine if the information learned by DL models trained on a protein fold classes is transferable to another completely different, fold class. If so, this would imply that SAdLSA has learned the protein folding code. The top hit ranked by its alignment score is subsequently evaluated and compared to the corresponding top hit by PSI-BLAST. In this test, SAdLSA uses the same sequence profile generated and employed by PSI-BLAST for its sequence alignments. As shown in the head-to-head comparison plot (Fig. 2A), SAdLSA exhibits a clear advantage over PSI-BLAST. For 83% of targets, SAdLSA achieves better alignments than PSI-BLAST in terms of the respective TM-score. The mean TM-score of SAdLSA is 0.558 versus 0.427 by PSI-BLAST. This difference is statistically highly significant (P-value $= 2.4 \times 10^{-13}$, paired two-tailed Student *t*-test, same below). The improvement is observed not only in cases difficult for PSI-BLAST at high e-values, but also for relatively easy ones. Perhaps, the most relevant metric is the count of cases where a good sequence alignment (TM-score >0.4) is obtained. Considering only the top-ranked hits, SAdLSA identifies good alignments for 89 cases whereas 64 cases are found by PSI-BLAST (Fig. 2C).

If we use sequence profiles generated by the well-established HMM approach HHsearch (Steinegger *et al.*, 2019) instead of the PSI-BLAST profiles as the training features and repeat the same training and testing procedures, we obtain good top-ranked hits for 91 cases and an improved mean TM-score at 0.587, which is slightly better than 0.576, the mean TM-score of top 1 hits by HHsearch using the same sequence profiles, though the statistical significance is not obvious at a *P*-value of 0.15. Note that we fully enable the Maximum Accuracy algorithm of HHsearch (setting the mact parameter at 0 throughout this study instead of the default value). With its default parameter, the mean TM-score of HHsearch top hits is significantly worse at 0.523. Nevertheless, the lack of a clear advantage over the optimal performance of HHsearch is largely due to the small training set generated from <700 α-helical structures and the fact that HH-suite already performs well on this relatively easy benchmark set for HHsearch (but not for SAdLSA whose parameters were learned from a limited number of α-helical proteins). We next demonstrate the advantage of SAdLSA on a bigger training dataset and on more challenging test sets.

3.1 Comprehensive benchmark tests on SCOP domains

Encouraged by the above results, which are strongly suggestive that SAdLSA has learned the underlying protein folding code, we further trained SAdLSA on a much larger set of 79k pairs from the SCOP30 library and tested it on an extrinsic test set of 593 protein domain sequences randomly selected from 391 SCOP folds (see Section 2.3). Homologs of the testing sequences at 30% sequence identity or higher, or with a BLAST e-value <0.1, are excluded from the training set.

3.2.1 Pairwise sequence alignment test

In the first test, PAIR947 that is designed to assess alignment quality, we tested the quality of the alignments for the 947 target/template pairs which have good structural alignments whose TM-scores >0.45. Each target sequence has up to two such pairs randomly selected from structural alignments across a template sequence library, SCOP10, which contains ∼3000 SCOP domains sharing <10% sequence identity with the target sequences. Thus, this test is very challenging. We compared the results of SAdLSA with HHsearch (Steinegger *et al.*, 2019). To make it a fair comparison, we employed the same sequence profiles as HHsearch in the training of SAdLSA. Overall, the performance of SAdLSA is significantly better than HHsearch (Fig. 3A–C). In 77% of cases, the alignments produced by SAdLSA are better than HHsearch. The mean TM-score of SAdLSA alignments is 0.339, which is 23% higher than the mean TM-score of 0.275 of HHsearch alignments (P-value $= 2.5 \times 10^{-106}$). Since this is a very challenging test set, we focus on good alignments with TM-scores >0.4. In this regard, SAdLSA produces 254 good alignments versus 168 by HHsearch. (Note that the default parameters of HHsearch produced only 106 good alignments.) Importantly, while both methods generally produce good alignments for relatively easy cases (blue circles in Fig. 3B), for the hard cases (red circles), SAdLSA found 127 good alignments, 154% more than the 50 cases found by HHsearch. This shows the highly significant advantage of SAdLSA (P-value $= 9.7 \times 10^{-106}$). Moreover, the probability scores estimated by SAdLSA for aligned residue–residue distances provide

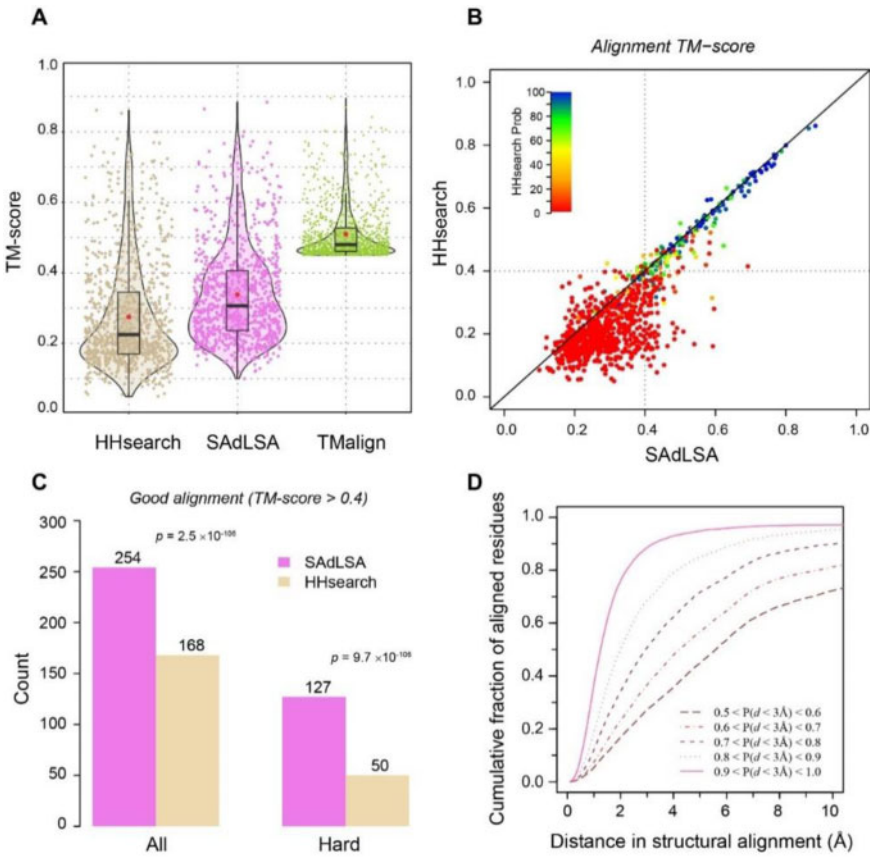

Fig. 3. Benchmark test PAIR947 on 947 protein sequence pairs compared using HHsearch and SAdLSA. (**A**) The violin plots show the TM-score given by the sequence alignment from HHsearch and SAdLSA, and the best structural alignment from TM-align. In each plot, the black box indicates the interquartile range from 25% to 75%; the median is represented by a black bar within the box, and the whisker extends up to 1.5 times the interquartile range. The red circle is positioned at the mean value. Individual data points from each method are shown as small circles in the same color code as the boxplot. The contour of the violin is proportional to the estimated density from the data point counts. (**B**) Head-to-head comparison between HHsearch and SAdLSA for all 947 cases. Each point is defined by the respective TM-scores assessed on the method's optimal sequence alignment. The color code indicates the level of difficulty according to the posterior probability scores given by HHsearch. Those with a probability score <0.95 are considered as hard cases. (**C**) Statistics of good sequence alignment hits with a TM-score >0.4 from the benchmark set. The *P*-value is calculated on paired TM-scores using the Student *t*-test. (**D**) Predicted alignment distance at different probability ranges estimated by SAdLSA versus the true distance by structural alignment according to TM-align

a reasonable confidence estimate for the alignments (Fig. 3D). About 88% of aligned residue pairs with a probability >90% within 3 Å are <3 Å in the actual optimal structural alignment of the native structures. This allows us to predict the TM-score using Equation (1) to estimate alignment quality.

3.2.2 Sequence library search test

Having demonstrated that SAdLSA indeed produces significant better sequence alignments than a classic HMM-based approach, we further show that its better sequence alignments could be utilized to identify better templates in the SCOP10 library consisting of ∼3000 sequences from ∼1000-folds, i.e. the LIB593 test. Figure 4A shows the results of the best of top 5 ranked hits by SAdLSA compared to HHsearch. Overall, in 77% of cases, SAdLSA predicts a better sequence alignment than HHsearch in terms of TM-score. The mean of TM-score by SAdLSA is 0.499, versus 0.454 by HHsearch (*P*-value $= 1.5 \times 10^{-38}$). Both methods dominate over the baseline method PSI-BLAST, which has a mean TM-score of 0.242 and produces poorer alignments in virtually all cases. Overall, SAdLSA found a good alignment for 390 cases, versus 338 cases by HHsearch. Among the hard cases, SAdLSA found good hits for 124 queries, 57% more than the 79 cases by HHsearch.

Moreover, to verify that SAdLSA indeed selects better templates than HHsearch, and does not just improve alignment quality for the same target/template pairs, we compare the quality of the template hits found above by performing the structural alignments using the experimental structures of the targets and the corresponding best template hits. As the results shown in Figure 4B indicate, in 297 (50%, mostly easy, in blue) cases, the two methods identify the same structural template; notably, in 218 (37%) cases, SAdLSA found a better template, which is ∼2.8 times the 78 (13%) cases where HHsearch finds a better template. Evidently, SAdLSA is superior to HHsearch in detecting significant structural relationships within a sequence library.

The quality of SAdLSA sequence alignment can be assessed from the predicted TM-score using Equation (1). Supplementary Figure S1 shows a high correlation coefficient of 0.89 between the predicted TM-score and the TM-score assessed on the structural model built from the SAdLSA sequence alignment, and a correlation coefficient of 0.85 between the predicted TM-score and the TM-score calculated using the experimental structures of both template and target.

Unlike the deep-learning-based contact predictions, SAdLSA is less dependent on sequence diversity of the multiple sequence alignment (MSA) employed for deriving input features. Supplementary Figure S2 plots the top models' TM-scores versus sequence diversity. It has a modest correlation at 0.40. This is not surprising because a significant alignment may be obtained if both sequences subjected to the alignment hit similar MSAs, even when there are very few sequences in the MSAs.

We also analyzed whether SAdLSA is biased toward certain fold classes. Supplementary Table S1 shows that there is no such bias in general. For example, a *t*-test on model TM-score between all α and

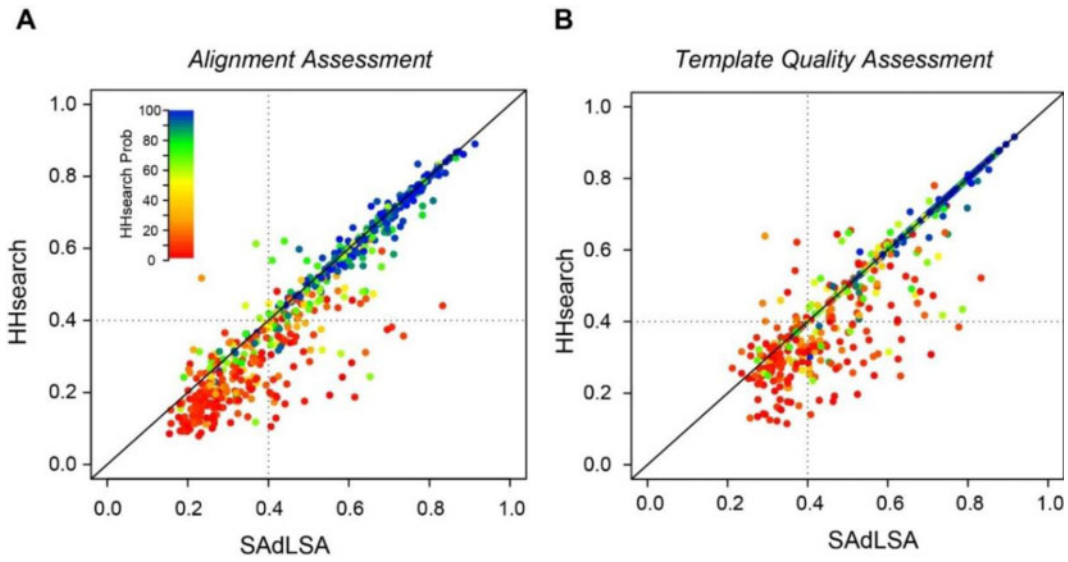

Fig. 4. TM-score comparison of HHsearch and SAdLSA for 593 test sequences by scanning SCOP10, a protein sequence library of ~3000 sequences. (**A**) Head-to-head comparison between HHsearch and SAdLSA for the best among top 5 ranked hits. Note that the selected template sequences can be different between two methods. Each point is the TM-score assessed using the optimal sequence alignment to the experimental structure. The color code indicates the level of difficulty as in Figure 2. (**B**) Head-to-head comparison for the identical target/template pairs as (A), but now assessed for the optimal structural alignment by TM-align on the experimental structures instead of the sequence alignment by SAdLSA or HHsearch. This provides a direct measure on the quality of the selected template, even though the sequence alignment may not be as good as the structural alignment

all β classes yields an insignificant *P*-value of 0.61. The mean TM-scores of all classes are around 0.50, with the exception that the α/β class is somewhat better at 0.55 and membrane proteins are somewhat worse at 0.42. The reason is likely due to the fact that the α/β class have better templates available than other classes, whereas the membrane class has fewer good template available, as reflected by the mean TM-score of the templates via structurally searching the best templates for each target (Supplementary Table S1).

Finally, the run time statistics of SAdLSA are shown in Supplementary Figure S3. Thanks to GPU acceleration, the time complexity is O(*N*) with respect to query sequence length, which is highly desirable for practical application. Using 4 Nvidia P100 GPUs, it took ~10 min to perform alignments against all 3000 sequences in the SCOP10 library for an input target sequence of ~400 residues in length.

4 Discussions

While these benchmark results demonstrate SAdLSA's significant advantage over previous methods, why is it better? Sequence alignment is essentially a comparison of the folding code encrypted within protein sequences. The classical substitution tables, such as the BLOSUM matrices (Henikoff and Henikoff, 1992), can be viewed as a mean-field approximation that compares individual amino acids. A better model, such as HMM, also considers immediate neighboring residues. While there are efforts to incorporate additional nonlocal interactions (Soding and Remmert, 2011), more accurate folding code comparison demands the consideration of nonlocal, long-range effects arising from correlations between residues that are distant in the sequence. Our deep-learning network with many layers learns the folding code with long-range effects by comparing entire sequences with their corresponding structural alignment. Indeed, here, a general deep-learning framework for improving sequence comparison is built by implicitly learning protein folding codes through structural comparison.

Although SAdLSA is ready for practical application, its linear time complexity is dependent on the memory capacity of GPUs. At present, proteins <1500 residues in length can be accommodated by 16 GB GPU memory. This may be overcome by parsing the input protein sequences into domains in advance. Another limitation is that the program relies on the generation of input sequence profiles by more speedy sequence library search tools.

One straightforward application of SAdLSA is to protein structure prediction. For this purpose, we note that there are different deep-learning-based approaches developed for predicting residue–residue contacts or even distances within the same sequence; the contact or distance information is then utilized for deriving the protein's structure (Gao *et al.*, 2019; Senior *et al.*, 2020; Xu, 2019). In contrast, by taking advantage of the observation that the number of distinct protein domain structures is rather small, we demonstrate that one can directly infer structural relationships to known folds. This provides a general-purpose sequence comparison approach whose potential applications go far beyond protein structure prediction, with the possibility of obtaining deeper functional or evolutional inferences.

Acknowledgements

We thank Hongyi Zhou for critical discussions, Bartosz Ilkowski and PACE at Georgia Tech for computing support and Jessica Forness for proof-reading the manuscript.

Funding

This work was supported in part by the Division of General Medical Sciences of the National Institute Health [NIH grant R35-118039].

Conflict of Interest: none declared.

References

Abadi,M. *et al.* (2016) TensorFlow: a system for large-scale machine learning. In *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, USENIX Association, Savannah, GA, USA, pp. 265–283.

Altschul,S.F. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Bishop,C.M. (2006) *Pattern Recognition and Machine Learning*. Springer, New York, NY.

Chothia,C. and Lesk,A.M. (1986) The relation between the divergence of sequence and structure in proteins. *EMBO J.*, **5**, 823–826.

Eddy,S.R. *et al.* (1995) Maximum discrimination hidden Markov models of sequence consensus. *J. Comput. Biol.*, **2**, 9–23.
Fox,N.K. *et al.* (2014) SCOPe: Structural Classification of Proteins—extended, integrating SCOP and ASTRAL data and classification of new structures. *Nucleic Acids Res.*, **42**, D304–D309.
Gao,M. and Skolnick,J. (2013) APoc: large-scale identification of similar protein pockets. *Bioinformatics*, **29**, 597–604.
Gao,M. *et al.* (2019) DESTINI: a deep-learning approach to contact-driven protein structure prediction. *Sci. Rep.*, **9**, 3514.
He,K. *et al.* (2016) Deep residual learning for image recognition. *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 770–778.
Henikoff,S. and Henikoff,J.G. (1992) Amino acid substitution matrices from protein blocks. *Proc. Natl. Acad. Sci. USA*, **89**, 10915–10919.
Holm,L. and Sander,C. (1996) Mapping the protein universe. *Science*, **273**, 595–603.
Kabsch,W. and Sander,C. (1983) Dictionary of protein secondary structure-pattern-recognition of hydrogen-bonded and geometrical features. *Biopolymers*, **22**, 2577–2637.
Muller,A. *et al.* (1999) Benchmarking PSI-BLAST in genome annotation. *J. Mol. Biol.*, **293**, 1257–1271.
Rost,B. (1999) Twilight zone of protein sequence alignments. *Protein Eng.*, **12**, 85–94.
Sadreyev,R. and Grishin,N. (2003) COMPASS: a tool for comparison of multiple protein alignments with assessment of statistical significance. *J. Mol. Biol.*, **326**, 317–336.
Senior,A.W. *et al.* (2020) Improved protein structure prediction using potentials from deep learning. *Nature*, **577**, 706–710.
Skolnick,J. *et al.* (2004) Development and large scale benchmark testing of the PROSPECTOR_3 threading algorithm. *Proteins: Struct. Funct. Bioinform.*, **56**, 502–518.
Smith,T.F. and Waterman,M.S. (1981) Identification of common molecular subsequences. *J. Mol. Biol.*, **147**, 195–197.
Soding,J. (2005) Protein homology detection by HMM-HMM comparison. *Bioinformatics*, **21**, 951–960.
Soding,J. and Remmert,M. (2011) Protein sequence comparison and fold recognition: progress and good-practice benchmarking. *Curr. Opin. Struct. Biol.*, **21**, 404–411.
Steinegger,M. *et al.* (2019) HH-suite3 for fast remote homology detection and deep protein annotation. *BMC Bioinform.*, **20**, 473.
Wu,C.H. (2006) The Universal Protein Resource (UniProt): an expanding universe of protein information. *Nucleic Acids Res.*, **34**, D187–D191.
Xu,J. (2019) Distance-based protein folding powered by deep learning. *Proc. Natl. Acad. Sci. USA*, **116**, 16856–16865.
Yona,G. and Levitt,M. (2002) Within the twilight zone: a sensitive profile-profile comparison tool based on information theory. *J. Mol. Biol.*, **315**, 1257–1275.
Zhang,Y. and Skolnick,J. (2004) Scoring function for automated assessment of protein structure template quality. *Proteins: Struct. Funct. Bioinform.*, **57**, 702–710.
Zhang,Y. and Skolnick,J. (2005) TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res.*, **33**, 2302–2309.
Zhang,Y. *et al.* (2006) On the origin and highly likely completeness of single-domain protein structures. *Proc. Natl. Acad. Sci. USA*, **103**, 2605–2610.