Published online 13 May 2019 *Nucleic Acids Research, 2019, Vol. 47, Web Server issue* ***W429–W436***
doi: 10.1093/nar/gkz384

LOMETS2: improved meta-threading server for fold-recognition and structure-based function annotation for distant-homology proteins

Wei Zheng[1], Chengxin Zhang[1], Qiqige Wuyun[2], Robin Pearce[1], Yang Li[1,3] and Yang Zhang[1,4,*]

[1]Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor, MI 48109, USA, [2]Computer Science and Engineering Department, Michigan State University, East Lansing, MI 48824, USA, [3]School of Computer Science and Engineering, Nanjing University of Science and Technology, Xiaolingwei 200, Nanjing 210094, China and [4]Department of Biological Chemistry, University of Michigan, Ann Arbor, MI 48109, USA

Received February 27, 2019; Revised April 19, 2019; Editorial Decision April 30, 2019; Accepted April 30, 2019

ABSTRACT

The LOMETS2 server (https://zhanglab.ccmb.med.umich.edu/LOMETS/) is an online meta-threading server system for template-based protein structure prediction. Although the server has been widely used by the community over the last decade, the previous LOMETS server no longer represents the state-of-the-art due to aging of the algorithms and unsatisfactory performance on distant-homology template identification. An extension of the server built on cutting-edge methods, especially techniques developed since the recent CASP experiments, is urgently needed. In this work, we report the recent advancements of the LOMETS2 server, which comprise a number of major new developments, including (i) new state-of-the-art threading programs, including contact-map-based threading approaches, (ii) deep sequence search-based sequence profile construction and (iii) a new web interface design that incorporates structure-based function annotations. Large-scale benchmark tests demonstrated that the integration of the deep profiles and new threading approaches into LOMETS2 significantly improve its structure modeling quality and template detection, where LOMETS2 detected 176% more templates with TM-scores >0.5 than the previous LOMETS server for Hard targets that lacked homologous templates. Meanwhile, the newly incorporated structure-based function prediction helps extend the usefulness of the online server to the broader biological community.

INTRODUCTION

Template-based modeling (TBM) represents the most accurate and reliable approach to protein structure prediction. The core procedure of TBM is fold-recognition, also known as threading, which aims to identify correct templates from the PDB with similar folds to the target protein and correctly align the target and template sequences. There is evidence that indicates the structural space of the current PDB library is complete and that TBM can be used to solve the protein structure prediction problem, at least for single-domain proteins (1). In reality, however, identification of ideal templates with correct alignments remains a major challenge, especially for protein targets that have no closely homologous templates in the PDB (2,3). To address this challenge, a large variety of threading methods have been developed over the last two decades based on various scoring functions and alignment algorithms (4–12). Because each threading algorithm has its own inherent advantages and disadvantages, there is no method that outperforms all others for every target protein (13,14). Thus, to improve the quality and reliability of TBM, several meta-server approaches were developed to combine threading results from multiple complementary algorithms (15–17). Early attempts at developing meta-threading-webservers, such as 3D-jury (15), submitted prediction jobs to different online servers and then combined the output results from each server after-the-fact. Such dependency on external online servers made these meta-servers difficult to maintain, especially when the external servers updated their APIs or stopped operating.

To improve the reliability of the meta-threading approach, we proposed LOMETS (16) which combined 9 locally-installed threading programs for protein structure prediction. Due to the robustness of the server's operation and the advantages associated with combining the results of multiple complementary algorithms over individual thread-

[*]To whom correspondence should be addressed. Tel: +1 734 647 1549; Fax: +1 734 615 6443; Email: zhng@umich.edu

ing programs for fold-recognition, LOMETS has been one of the most widely used meta-threading servers for protein structure prediction. Since its initial release in 2007, LOMETS has generated fold-recognition and full-length structure modeling for more than 30,000 protein sequences submitted by 15,201 users. With rapid developments in the field, however, the previous LOMETS server no longer represents the state-of-the-art in protein threading. In particular, the recent CASP experiments (3,18) have witnessed significant new progress in TBM, where contact-maps predicted through deep neural-network training, in conjunction with metagenomic sequencing, have been found to be vitally useful for modeling distant- and non-homologous protein sequences. Thus, integration of the new threading methods and sequence and structural data resources is urgently needed to further improve the quality of the meta-threading approach. This is the major motivation for the development of the second generation meta-threading server, LOMETS2, in this work.

Compared to the previous LOMETS server, LOMETS2 introduces three major new extensions: (i) the server now integrates a new and more comprehensive set of state-of-the-art threading programs, including contact-guided threading approaches; (ii) it utilizes deep profiles which are generated using a novel deep Multiple Sequence Alignment (MSA) construction method to improve the profile-based alignment accuracy; (iii) it adopts a re-designed, user-friendly interface and adds structure-based function annotations for a more comprehensive results report. The benchmark tests demonstrated that the LOMETS2 structure model quality is significantly better than any of its component threading programs as well as the previous meta-threading approach, LOMETS (16).

MATERIALS AND METHODS

Overview of the LOMETS2 pipeline

The LOMETS2 server pipeline consists of four consecutive steps: generation of deep sequence profiles, fold-recognition through its component threading programs, template alignment selection, and full-length model construction (Figure 1). Starting from a target protein sequence provided by the server user, a deep MSA is generated by iterative sequence homology searches against multiple sequence databases. The resulting deep MSA is used to calculate deep profiles in the form of sequence profiles (or position-specific scoring matrices) and profile Hidden Markov Models (HMMs). The deep profiles are then used by the 11 LOMETS2 threading programs, which are all locally installed on our computer cluster, to identify template structures from the template library, where deep profiles are pre-built for each template. Next, 220 templates (the top 20 templates per program) are combined to give the final LOMETS2 threading results. These 220 templates are ranked by a scoring function that integrates Z-score, confidence score for each method, and sequence identity between the identified templates and query sequence (see the detailed description in Text S1 in the Supporting Information, SI). Meanwhile, the function annotations associated with each template are collected from the BioLiP structure-function database (19). Finally, full-length models are constructed by MODELLER (20) utilizing the multiple distance restraints collected from the top-ranked templates identified by LOMETS2.

New developments in LOMETS2

DeepMSA method. The profiles (sequence profiles or profile HMMs), which are derived from MSAs for the query/template sequences, are one of the most important features for template recognition and threading-based sequence alignment. General MSA generation methods, such as PSI-BLAST (21) and HHblits (22), sometimes cannot collect a sufficient number of homologs in an MSA, especially for Hard targets, which can result in poor alignment quality when using profile-based threading algorithms. LOMETS2 employs a new composite approach, called DeepMSA, to create sensitive MSAs with higher numbers of diverse sequences than general MSA generation methods.

Starting from a query sequence, a deep MSA is generated by a three stage procedure (Supplementary Figure S1). In Stage 1, HHblits from the HH-Suite package is used to search the query against UniClust30 (23) to generate the first-level MSA. If the number of effective sequences (*Neff*, defined in Text S2) generated by Stage 1 is <128, Stage 2 will be performed. In Stage 2, Jackhmmer from the HMMER package (24) is used to search the query sequence against UniRef90 (25) to extract full-length sequences (hits). These hits are then converted into a custom HHblits-formatted database. Jump-starting it from the first-level MSA, HHblits is again applied to search this custom database to generate the second-level MSA. If the *Neff* of the second-level MSA is still <128, Stage 3 will be performed, where the second-level MSA is converted into a profile HMM by HMMbuild from the HMMER package. This HMM is then searched against the Metaclust (26) metagenome sequence database by HMMsearch from HMMER to extract full-length hits. Similar to Stage 2, hits from HMMsearch are built into a custom HHblits database. The second-level MSA is used to jump-start an HHblits search against this new custom HHblits database to get the third-level MSA. Using these three Stages, we generate deep MSAs, which in turn are utilized to build the final sequence profiles used by each of the LOMETS2 threading methods.

New threading methods. The LOMETS2 server integrates predictions from 11 different state-of-the-art threading algorithms, which can be classified into three types: the contact-based method CEthreader; sequence profile-based methods, including FFAS3D (10), MUSTER (9), Neff-MUSTER, PPAS (16), PROSPECT2 (6), SP3 (27) and SparksX (8); and profile HMM-based methods, consisting of HHsearch (11), HHpred (28) and PRC (29). Among them, CEthreader, FFAS3D, HHpred, MUSTER, Neff-MUSTER and SparksX are newly added to the LOMETS2 server. HHpred is a profile HMM-based threading method extended from HHsearch. SparksX and FFAS3D are both based on profile-profile comparison and use similar features, but FFAS3D employs a template re-ranking strategy. MUSTER, Neff-MUSTER and CEthreader are our in-house threading programs. MUSTER and Neff-MUSTER are profile-profile alignment algorithms that use dynamic

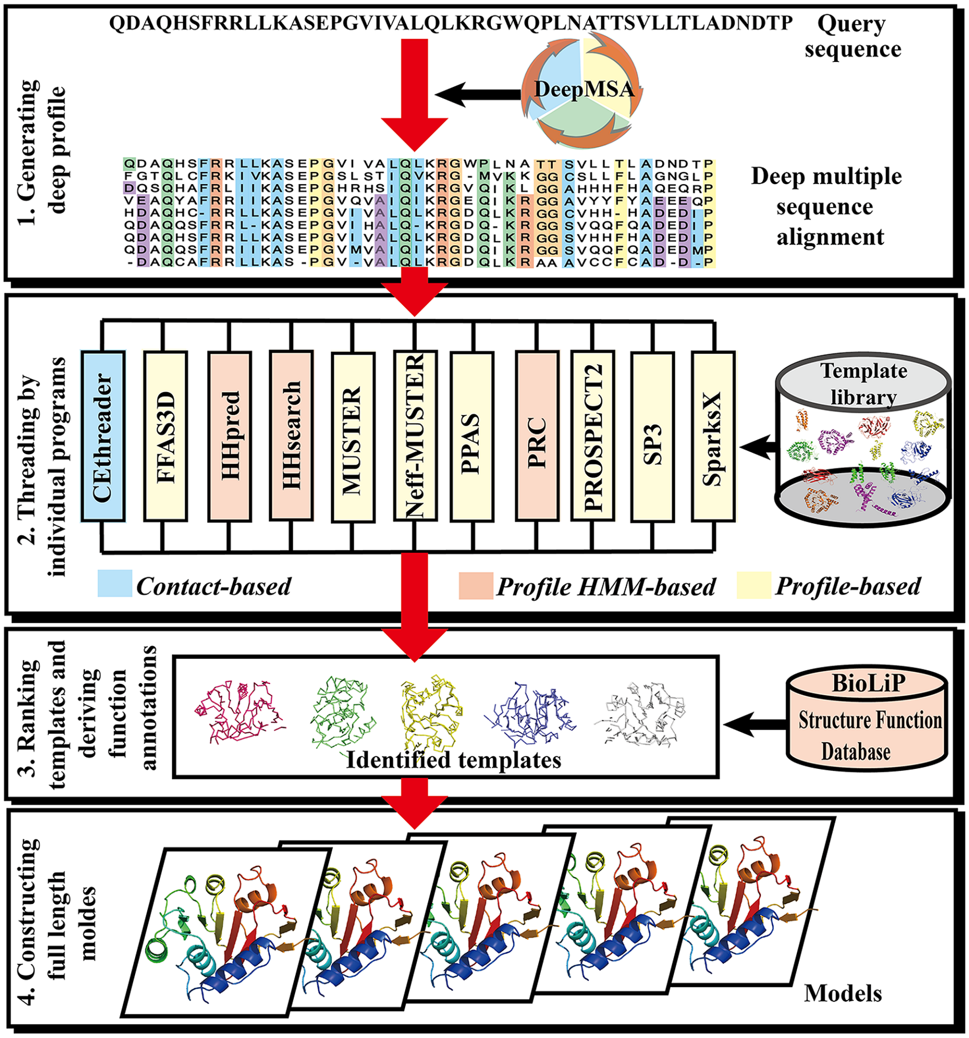

QDAQHSFRRLLKASEPGVIVALQLKRGWQPLNATTSVLLTLADNDTP
Query sequence
1. Generating deep profile
DeepMSA
QDAQHSFRRLLKASEPGVI VALQLKRGWPLNA TTSVLLTLADNDTP
FGTQLCFRKI VKASEPGSLSTIQVKRG-MVKK GGCSLLFLAGNGLP
DQSQHAFRLIIKASEPGHRHSIQIKRGVQIKL GGAHHHFHAQEQRP
VEAQYAFRRIIKASEPGVQVAIQIKRGEQIKR GGAVYYFYAEEEQP
HDAQHC-RRLLKASEPGVI VALQLKRGDQLKR GGCVHH-HADEDIP
QDAQQSFRRL-KASEPGVI HALQ-KRGDQ-KR GGSVQQFQADEDIP
QDAQHSFRRLLKASEPGVI VALQLKRGDQLKR GGSVHHFHADEDIP
QDAQQSFRRIIKASEPGVMVAIQIKRGDQIKR GGSVQQFQADEDMP
-DAQCAFRRLLKAS-PGV-VALQLKRGDQLKR AAAVCCFCAD-D-P
Deep multiple sequence alignment
2. Threading by individual programs
CEthreader
FFAS3D
HHpred
HHsearch
MUSTER
Neff-MUSTER
PPAS
PRC
PROSPECT2
SP3
SparksX
Template library
Contact-based
Profile HMM-based
Profile-based
3. Ranking templates and deriving function annotations
Identified templates
BioLiP
Structure Function Database
4. Constructing full length moldes
Models

Figure 1. The flowchart of the LOMETS2 server pipeline consists of four steps: generation of deep profiles, fold-recognition by 11 individual threading programs, template selection and full-length model construction.

programming together with a myriad of features, including sequence profiles, secondary structures, structure fragment profiles, solvent accessibility, dihedral torsional angles, and hydrophobic scoring matrices for fold-recognition. The major difference between MUSTER and Neff-MUSTER is that MUSTER uses a set of fixed weights for the sequence profile term while Neff-MUSTER employs dynamic weights derived from the *Neff* of the MSA used to build the sequence profile. Lastly, CEthreader is a contact-driven threading method, which first predicts the contact-map for a query sequence using our in-house, deep learning-based contact-map prediction method, ResPRE (30) (see Text S3). From the predicted contact-map, CEthreader generates single-body contact eigenvectors through the eigen-decomposition technique, and finally searches and aligns the query to templates using dynamic programming based on the contact eigenvectors as well as secondary structure and sequence profile comparison (see Text S4). A more detailed description for each of the 11 LOMETS2 threading methods is provided in Text S5.

A unified deep profile-based template library. In the previous version of the LOMETS server, the template libraries were collected from the original developers of each individual threading program, which negatively impacted its performance since some libraries were not comprehensive or were not updated regularly. In LOMETS2, a unified library of non-redundant protein structure templates is regularly collected for each of the threading methods. This library currently consists of 71 684 structures with a pair-wise sequence identity <70%. In addition to the structures and structure-based feature files, the DeepMSA program is used to create MSAs and then pre-build the deep sequence profiles and deep profile HMMs for each template. The library is updated weekly in accordance with the update cycle of the PDB database and has been made publicly available at https://zhanglab.ccmb.med.umich.edu/library/.

Re-designed server output. The output page of the LOMETS2 server was re-designed to make the server more convenient to use and to incorporate more comprehensive function annotations. The 3Dmol (31) applet is used to visualize the modeled structures. Moreover, we added function annotations derived from threading templates, including Gene Ontology (GO) terms for Molecular Function (MF), Biological Process (BP) and Cellular Component (CC) as well as the Enzyme Commission (EC) numbers.

RESULTS

Dataset

To test the LOMETS2 server, we constructed a non-redundant benchmark set of 614 proteins from the PDB with pairwise sequence identities <30%. The proteins in the benchmark set had lengths ranging from 57 to 500 residues and included 141 α-, 187 β- and 286 $\alpha\beta$-proteins. To make it relevant to TBM, each target protein was required to have at least one template with sequence identity <30% and TM-score >0.5 to the target as identified by TM-align structural alignment searches through the template library (32). Based on LOMETS classification, which categorizes proteins into different classes based on the significance scores of multiple threading programs, this dataset contained 403 Easy targets (for which at least one threading program in LOMETS detected a significant template hit) and 211 Hard targets (for which no program in LOMETS identified a significant hit). To remove bias caused by homologous sequences, a uniform sequence identity cutoff of <30% between the target and template sequences was enforced when the threading tests were conducted for each of the programs.

Overall comparison between LOMETS and LOMETS2

Figure 2A presents a head-to-head comparison between the TM-scores of the first templates identified by LOMETS and those identified by LOMETS2. The average TM-scores for the first templates identified by LOMETS2 for All, Easy and Hard targets were 0.62, 0.71 and 0.46, respectively, which were 9%, 4% and 31% higher than those identified by LOMETS. Out of the 614 targets, LOMETS2 successfully detected correct templates with a TM-score >0.5 for 479 targets, which was 17% higher than the number of targets with correct templates identified by LOMETS. For Hard targets in particular, LOMETS2 detected correct templates for 94 targets, while LOMETS only successfully detected correct templates for 34 targets. In Figure 2B, we show a comparison between the average TM-scores for the full-length models constructed by MODELLER based on templates identified by either LOMETS or LOMETS2. The average TM-scores of the first models generated by LOMETS2 for All, Easy and Hard targets were 11%, 5% and 35% higher than those by LOMETS.

Why does LOMETS2 outperform LOMETS?

The data in Figure 2 demonstrates that LOMETS2 significantly outperforms LOMETS in terms of identifying higher quality templates and generating better full-length models, especially for Hard targets that lack homologous templates. This improvement can be mainly attributed to three new developments in LOMETS2. First, the newly developed DeepMSA program detects more homologous sequences compared to traditional methods, thereby improving the accuracy of profile-based alignment algorithms. In Supplementary Figure S2, we list a comparison between the number of effective sequences detected by DeepMSA for the 614 targets versus the numbers detected by PSI-BLAST and HHblits, where the latter (PSI-BLAST and HHblits) were used as the default profile construction methods for the previous threading programs in LOMETS. On average, DeepMSA detected 6,500 sequences per protein (or *Neff* = 307), which is 8 (or 5) times higher than the number detected by HHblits and 10 (or 9) fold higher than the number identified by PSI-BLAST. In Figure 3, we present a comparison of the performance of each individual threading program using either the deep or default profiles, where the performance is improved by deep profiles for all threading programs. The specific TM-score values for each threading program are listed in Supplementary Table S1 in the SI, which shows that the improvements are statistically significant (with P-values $< 10^{-5}$ in Student's t-test) for nearly all of the programs (except for FFAS3D for which the use of deep MSAs increased the TM-score by 1%, but the difference is not statistically significant). Since the only difference between the two sets of programs was the sequence and HHM profiles that were used, the data demonstrates that the deep MSA generation method has a remarkable impact on improving the performance of nearly all individual methods as well as the overall meta-threading approach.

Second, the introduction of the new threading programs improves the overall quality of the LOMETS2 server results. In total, six new programs (CEthreader, HHpred, SparksX, FFAS3D, MUSTER and Neff-MUSTER) were added to LOMETS2 and four programs were retired due to poor performance. As shown in Supplementary Table S1, all of the new programs performed better, in terms of average template TM-score, than the ones inherited from the LOMETS server. In particular, the new CEthreader program identified templates with the highest TM-scores and detected more correct templates than any other threading program. Due to the utilization of *de novo* contact-map prediction, CEthreader is especially useful for Hard targets that lack homologous templates. This is demonstrated by the fact that the average TM-score of the first templates identified by CEthreader for Hard targets was at least 19% higher than any other LOMETS2 threading program, including the best sequence profile-based method SparksX (average TM-score = 0.378) and the best profile HMM-based method HHpred (average TM-score = 0.374). Meanwhile, the number of targets for which CEthreader identified correct first templates (TM-score > 0.5) was 61% and 74% more than the number of targets for SparksX and HHpred, respectively. The integration of the contact-guided threading method is particularly important for improving the overall ability of LOMETS2 to recognize distant-homology templates.

Finally, the new template-ranking score, which combines Z-score, program-specific confidence scores and sequence identity (Eq. S2 in Text S1), helps to select correct templates. Although CEthreader significantly outperforms other individual programs, on average, the LOMETS2 meta-server strategy still selects templates with higher TM-scores than any of its individual programs, including CEthreader. In Supplementary Table S2, we list the average TM-scores for the first full-length models constructed by MODELLER using templates from different threading programs,

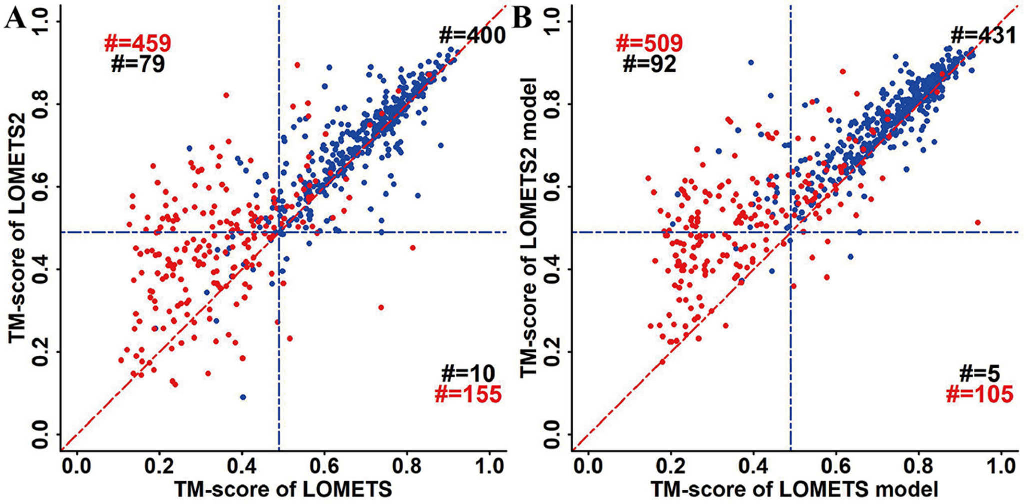


Figure 2. The TM-scores of the first templates (**A**) and full-length models (**B**) predicted by LOMETS and LOMETS2 for 614 test proteins. Red and blue points correspond to Hard and Easy targets, respectively. The red numbers represent the number of targets above or below the diagonal line, while the black numbers correspond to the number of targets in each sub-square.

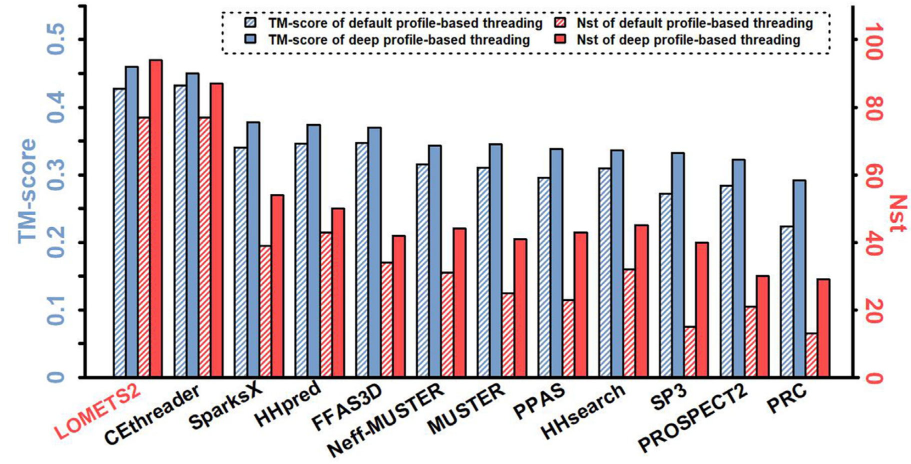


Figure 3. Average TM-scores of the first templates detected by LOMETS2 and its component threading programs based on default profiles versus those based on deep profiles for 211 Hard targets. N_{st} represents the number of targets whose first template has a TM-score >0.5.

where the average TM-score for LOMETS2 (TM-score = 0.6753) was 5.7% higher than its best threading program, CEtheader (TM-score = 0.6391), with a *P*-value of 2.9×10^{-61}. This demonstrates the efficacy of the composite ranking score for template selection. Here, we note that the difference in performance between LOMETS2 and its best component program is more pronounced for full-length model production (shown in Supplementary Table S2) than for raw template selection/alignment (Supplementary Table S1). This is because LOMETS2 ranks the templates from CEthreader as first in most cases due to the high confidence score given to the CEthreader program, which results in a modest difference between LOMETS2 and CEthreader when only considering the TM-scores of the first template alignments. For the full-length model construction, however, the consensus distance restraints collected from multiple complementary template alignments from different threading programs help MODELLER construct much more reliable full-length models than those produced by the restraints from individual threading programs; the use of such complementarity of different threading algorithms represents one of the major advantages of the meta-server approach (14,17).

Performance on recent Critical Assessment of Structure Prediction (CASP) experiments

The community-wide CASP experiments provide a unique opportunity to stringently test structure prediction methodologies (33). A preliminary version of LOMETS2 (without deep profiles) was used as the template identification component of the 'Zhang-Server' and 'QUARK' groups, which were ranked as the top two structure prediction algorithms in the automated Server Section in CASP13 in terms of the *Z*-score of the GDT-TS score (see http://www.predictioncenter.org/casp13/zscores_final.cgi?model_type=first&gr_type=server_only). However, LOMETS2 did not participate in the CASP experiment as an independent server.

To examine its performance, we ran the LOMETS2 program (with deep profiles) for the CASP13 targets, where all templates released after May 1, 2018 were excluded. Supplementary Figure S3 lists the average TM-score of the new LOMETS2 models for the 121 released domains, in comparison to the top server groups in CASP13. Overall, LOMETS2 ranked No. 4, following 'Zhang-Server' and 'QUARK', which used LOMETS2 templates as input templates, and 'RaptorX-DeepModeller'. 'RaptorX-DeepModeller' is an integration of two pipelines, namely, 'RaptorX-TBM', which generates models using a distance-guided threading program DeepThreader (34), and 'RaptorX-Contact', which is a distance-based *ab initio* folding method (35). In Supplementary Figure S4, we show a head-to-head comparison between models produced by LOMETS2, RaptorX-TBM and CEthreader for the 121 CASP13 domains. While LOMETS2 showed an overall comparable performance with RaptorX-TBM, RaptorX-TBM outperformed CEthreader; this latter difference probably reflects the advantage of distance over contact prediction for the improvement of threading performance. Additionally, RaptorX-TBM uses a combination of MODELLER and Rosetta for full-length model construction, while CEthreader and LOMETS2 only use MODELLER here.

In Supplementary Table S3, we further present a comparison of the models produced by LOMETS2 and its component threading programs for the 121 CASP13 domains. While CEthreader again outperformed other individual programs, LOMETS2 had a significantly higher TM-score than all of its component threading programs, with *P*-values $<10^{-6}$ for all the TM-score differences. This result is consistent with the large-scale benchmark tests shown above (Figure 3) and demonstrates the robustness and effectiveness of LOMETS2 for template identification and model construction.

WEB SERVER

Server input

The input to the LOMETS2 server is a single-chain amino acid sequence file in FASTA format. After submitting a job, a URL link with a random job ID is generated which allows the user to view the results while maintaining the user's data privacy. The user can optionally provide an email address and LOMETS2 will automatically send a notification email with a link to the results page upon job completion. This email address can also be used to retrieve the list of all previously submitted jobs by the user through https://zhanglab.ccmb.med.umich.edu/LOMETS2/check.html. Users will usually receive the prediction results in less than 24 h after submitting a sequence (protein size ≤ 1500 AA). However, the running time depends on the protein size, where a smaller protein takes less time than a larger protein. Additionally, if too many sequences are accumulated in the queue, the procedure may take a longer time (see Supplementary Figure S5 for the practical response time of the LOMETS2 server for 1,433 recently processed jobs). By default, LOMETS2 keeps all identified templates, while an advanced option is provided that enables users to remove templates sharing >30% sequence identity with the input sequence.

Server output

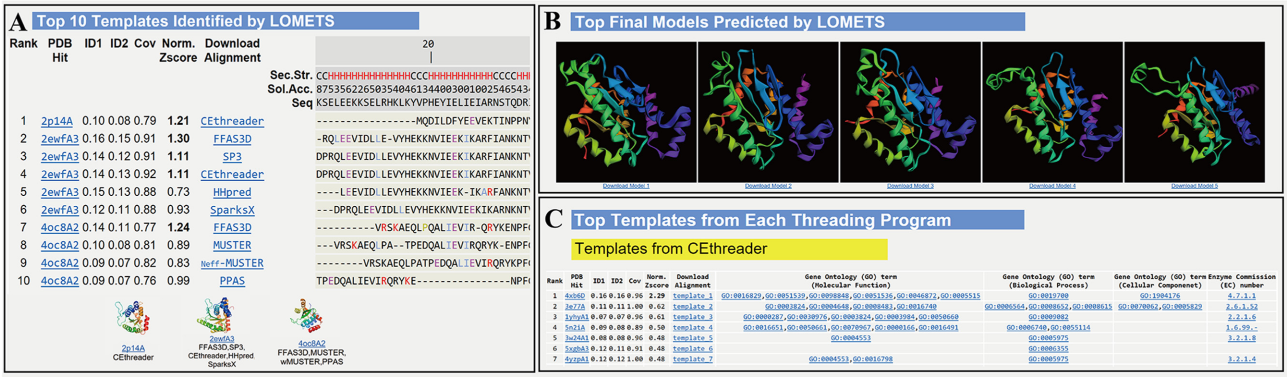
The LOMETS2 results page consists of six sections, including (i) the user input sequence, (ii) predicted secondary structure, (iii) predicted solvent accessibility, (iv) summary of the top 10 template alignments identified by LOMETS2, (v) full-length models built by MODELLER using the top five templates and (vi) function annotations associated with the top templates from the component threading programs. As an illustration, Figure 4 presents an example from the FokI restriction endonuclease (PDB ID: 2fokA) to explain the major sections (iv–vi) of the results page. When we ran this example, we removed templates with >30% sequence identity to the input sequence.

Section iv lists the top 10 templates identified by LOMETS2 (Figure 4A). For each template, the left panel displays the template's PDB ID, sequence identity to the query, alignment coverage, normalized *Z*-score, and the threading method that detected the template. Here, the normalized *Z*-score is equal to Z/Z_0, where Z_0 is a program specific cut-off (see Text S1). The normalized *Z*-score signifies the confidence of the template alignment, where a normalized *Z*-score ≥1 indicates a good alignment by the corresponding threading program. Users can download the alignment by clicking the threading program name. The right panel of this table shows the predicted secondary structure, predicted solvent accessibility and the query-template alignments by LOMETS2. The bottom panel displays 3D structures of the templates, together with the threading methods that detected them.

Section v (Figure 4B) shows the top five full-length models generated by LOMETS2. Users can drag, rotate or zoom in on the structure to check the model. PDB-formatted structure model files can be downloaded by clicking the link under the figures of the models. The last section (Figure 4C) displays up to 10 top templates from the component threading programs, where function annotations (Gene Ontology terms and Enzyme Commission numbers) curated from the BioLiP database are listed in the right panel, next to the corresponding template alignments in the left panel. Taking the CEthreader results as an example, the 8th–10th columns show the GO terms for the Molecular Function (MF), Biological Process (BP) and Cellular Component (CC) aspects. The four-digit EC number is shown in the 11th column, which indicates the type of enzyme to which the template belongs. The GO terms and EC number are hyperlinked to the QuickGO (36) Gene Ontology and the ExPASy (37) ENZYME databases.

CONCLUSION

In this work, we report a significantly extended version of the LOMETS server for template-based protein structure prediction and function annotation. The new LOMETS2 server contains 11 state-of-the-art threading programs, which are powered by a new deep MSA pipeline for enhanced profile- and HMM-based template alignment. To



A Top 10 Templates Identified by LOMETS

Rank	PDB Hit	ID1	ID2	Cov	Norm. Zscore	Download Alignment	20
						Sec.Str.	CCHHHHHHHHHHHHHHCCCHHHHHHHHHHHCCCCHH
						Sol.Acc.	87535622650354046134400300100254654
						Seq	KSELEEKKSELRHKLKYVPHEYIELIEIARNSTQDR
1	2p14A	0.10	0.08	0.79	**1.21**	CEthreader	-----------------MQDILDFYEEVEKTINPPN
2	2ewfA3	0.16	0.15	0.91	**1.30**	FFAS3D	-RQLEEVIDLLE-VYHEKKNVIEEKIKARFIANKNT
3	2ewfA3	0.14	0.12	0.91	**1.11**	SP3	DPRQLEEVIDLLEVYHEKKNVIEEKIKARFIANKNT
4	2ewfA3	0.14	0.13	0.92	**1.11**	CEthreader	DPRQLEEVIDLLEVYHEKKNVIEEKIKARFIANKNT
5	2ewfA3	0.15	0.13	0.88	0.73	HHpred	----LEEVIDLLEVYHEKKNVIEEK-IKARFANKNT
6	2ewfA3	0.12	0.11	0.88	0.93	SparksX	---DPRQLEEVIDLLEVYHEKKNVIEEKIKARNKNT
7	4oc8A2	0.14	0.11	0.77	**1.24**	FFAS3D	----------VRSKAEQLPQALIEVIR-QRYKENPF
8	4oc8A2	0.10	0.08	0.81	0.89	MUSTER	---VRSKAEQLPA--TPEDQALIEVIRQRYK-ENPF
9	4oc8A2	0.09	0.07	0.82	0.83	Neff-MUSTER	--------VRSKAEQLPATPEDQALIEVIRQRYKPF
10	4oc8A2	0.09	0.07	0.76	0.99	PPAS	TPEDQALIEVIRQRYKE---------------NPF

C Top Templates from Each Threading Program

Templates from CEthreader

Rank	PDB Hit	ID1	ID2	Cov	Norm. Zscore	Download Alignment	Gene Ontology (GO) term (Molecular Function)	Gene Ontology (GO) term (Biological Process)	Gene Ontology (GO) term (Cellular Componenet)	Enzyme Commission (EC) number
1	4xb6D	0.16	0.16	0.96	2.29	template_1	GO:0016829,GO:0051539,GO:0098848,GO:0051536,GO:0046872,GO:0005515	GO:0019700	GO:1904176	4.7.1.1
2	3e77A	0.11	0.11	1.00	0.62	template_2	GO:0003824,GO:0004648,GO:0008483,GO:0016740	GO:0006564,GO:0008652,GO:0008615	GO:0070062,GO:0005829	2.6.1.52
3	1yhyA1	0.07	0.07	0.96	0.61	template_3	GO:0000287,GO:0030976,GO:0003824,GO:0003984,GO:0050660	GO:0009082		2.2.1.6
4	5n2iA	0.09	0.08	0.89	0.50	template_4	GO:0016651,GO:0050661,GO:0070967,GO:0000166,GO:0016491	GO:0006740,GO:0055114		1.6.99.-
5	3w24A1	0.08	0.08	0.96	0.48	template_5	GO:0004553	GO:0005975		3.2.1.8
6	5xgbA3	0.12	0.11	0.91	0.48	template_6		GO:0006355		
7	4yzpA1	0.12	0.12	1.00	0.48	template_7	GO:0004553,GO:0016798	GO:0005975		3.2.1.4

Figure 4. Illustration of the LOMETS2 server output using FokI restriction endonuclease (PDB ID: 2fokA) as an example. The three main sections of the LOMETS2 output are (**A**) summary of the top 10 templates identified by LOMETS2, (**B**) five models predicted by LOMETS2 and (**C**) the top templates identified by each threading program.

provide a solid foundation for template-based modeling, a unified template library has been developed which helps facilitate the maintenance and upkeep of the library for all component threading methods. Meanwhile, the output interface was re-designed to enable interactive visualization of the structure models and function annotations.

The large-scale benchmark tests showed that the new developments significantly improve the accuracy of the meta-threading server, especially for Hard targets that lack homologous templates, where LOMETS2 detected 176% more templates with correct folds (TM-score > 0.5) (38) than LOMETS. Detailed data analyses showed these improvements are mainly due to the integration of the new cutting-edge threading algorithms, including those that incorporate deep convolutional network-based contact-map predictions. Meanwhile, the deep MSA-based profiles systematically improve the alignment accuracy for nearly all of the component threading algorithms.

The LOMETS2 server indirectly participated in the CASP experiment as it was used as the template identification component of the top-ranking 'Zhang-Server' and 'QUARK' groups in the Server Section of the most recent CASP13 experiment. Although LOMETS2 represents one of the most accurate servers for structure prediction, it is still outperformed by I-TASSER and QUARK, which implement far more intensive structure assembly simulations. Nevertheless, LOMETS2 has unique value to serve as a robust resource for entry-level template recognition and template-based function annotation with reasonable modeling accuracy; this is particularly important for large-scale structure and function modeling for which folding and refinement simulations are prohibitively expensive. Moreover, based on the feedback from our server system users, many biological users are interested in knowing the original templates used to construct the final structural models. This knowledge is often useful in order to better interpret and understand the characteristics and functional implications of the structural models. The detailed template information is typically lost in the composite models produced by I-TASSER and QUARK, but is well-preserved in the LOMETS server models and the output data, which include the rich functional annotations for the templates and multiple template alignments from the different threading programs. Overall, with constant effort on the development and extension of the methodology, we expect LOMETS2 will continue to serve as an important, unique and reliable structure and function modeling resource for the broader biological community.

DATA AVAILABILITY

The webserver and benchmark dataset are available at https://zhanglab.ccmb.med.umich.edu/LOMETS/.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENTS

This work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by the National Science Foundation [ACI-1548562].

FUNDING

National Institutes of Health [GM083107, GM116960, AI134678]; National Science Foundation [DBI1564756]. Funding for open access charge: National Institutes of Health.

Conflict of interest statement. None declared.

REFERENCES

1. Zhang,Y. and Skolnick,J. (2005) The protein structure prediction problem could be solved using the current PDB library. *PNAS*, **102**, 1029–1034.
2. Dunbrack,R. (2014) *11th Community Wide Experiment on the Critical Assessment of Techniques for Protein Structure Prediction.* Riviera Maya.
3. Kryshtafovych,A., Monastyrskyy,B., Fidelis,K., Moult,J., Schwede,T. and Tramontano,A. (2018) Evaluation of the template-based modeling in CASP12. *Proteins*, **86**(Suppl. 1), 321–334.
4. Bowie,J.U. and Eisenberg,D. (1994) An evolutionary approach to folding small alpha-helical proteins that uses sequence information and an empirical guiding fitness function. *PNAS*, **91**, 4436–4440.
5. Jones,D.T., Taylor,W.R. and Thornton,J.M. (1992) A new approach to protein fold recognition. *Nature*, **358**, 86–89.

6. Xu,Y. and Xu,D. (2000) Protein threading using PROSPECT: design and evaluation. *Proteins*, **40**, 343–354.
7. Skolnick,J., Kihara,D. and Zhang,Y. (2004) Development and large scale benchmark testing of the PROSPECTOR_3 threading algorithm. *Proteins*, **56**, 502–518.
8. Yang,Y., Faraggi,E., Zhao,H. and Zhou,Y. (2011) Improving protein fold recognition and template-based modeling by employing probabilistic-based matching between predicted one-dimensional structural properties of query and corresponding native properties of templates. *Bioinformatics*, **27**, 2076–2082.
9. Wu,S. and Zhang,Y. (2008) MUSTER: Improving protein sequence profile-profile alignments by using multiple sources of structure information. *Proteins*, **72**, 547–556.
10. Xu,D., Jaroszewski,L., Li,Z. and Godzik,A. (2014) FFAS-3D: improving fold recognition by including optimized structural features and template re-ranking. *Bioinformatics*, **30**, 660–667.
11. Soding,J. (2005) Protein homology detection by HMM-HMM comparison. *Bioinformatics*, **21**, 951–960.
12. Kelley,L.A., Mezulis,S., Yates,C.M., Wass,M.N. and Sternberg,M.J. (2015) The Phyre2 web portal for protein modeling, prediction and analysis. *Nat. Protoc.*, **10**, 845–858.
13. Fischer,D. (2006) Servers for protein structure prediction. *Curr. Opin. Struct. Biol.*, **16**, 178–182.
14. Zhang,Y. (2008) Progress and challenges in protein structure prediction. *Curr. Opin. Struct. Biol.*, **18**, 342–348.
15. Ginalski,K., Elofsson,A., Fischer,D. and Rychlewski,L. (2003) 3D-Jury: a simple approach to improve protein structure predictions. *Bioinformatics*, **19**, 1015–1018.
16. Wu,S. and Zhang,Y. (2007) LOMETS: a local meta-threading-server for protein structure prediction. *Nucleic Acids Res.*, **35**, 3375–3382.
17. Fischer,D. (2003) 3D-SHOTGUN: a novel, cooperative, fold-recognition meta-predictor. *Proteins*, **51**, 434–441.
18. Croll,T., Sammito,M., Bunkoczi,G. and Read,R.J. (2018) Assessing template-based models. *13th Community Wide Experiment on the Critical Assessment of Techniques for Protein Structure Prediction*.
19. Yang,J., Roy,A. and Zhang,Y. (2013) BioLiP: a semi-manually curated database for biologically relevant ligand-protein interactions. *Nucleic Acids Res.*, **41**, D1096–D1103.
20. Sali,A. and Blundell,T.L. (1993) Comparative protein modelling by satisfaction of spatial restraints. *J. Mol. Biol.*, **234**, 779–815.
21. Altschul,S.F., Madden,T.L., Schaffer,A.A., Zhang,J., Zhang,Z., Miller,W. and Lipman,D.J. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.
22. Remmert,M., Biegert,A., Hauser,A. and Soding,J. (2011) HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nat. Methods*, **9**, 173–175.
23. Mirdita,M., von den Driesch,L., Galiez,C., Martin,M.J., Soding,J. and Steinegger,M. (2017) Uniclust databases of clustered and deeply annotated protein sequences and alignments. *Nucleic Acids Res.*, **45**, D170–D176.
24. Johnson,L.S., Eddy,S.R. and Portugaly,E. (2010) Hidden Markov model speed heuristic and iterative HMM search procedure. *BMC Bioinformatics*, **11**, 431.
25. Suzek,B.E., Wang,Y., Huang,H., McGarvey,P.B. and Wu,C.H. (2015) UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*, **31**, 926–932.
26. Steinegger,M. and Soding,J. (2018) Clustering huge protein sequence sets in linear time. *Nat. Commun.*, **9**, 2542.
27. Zhou,H. and Zhou,Y. (2005) Fold recognition by combining sequence profiles derived from evolution and from depth-dependent structural alignment of fragments. *Proteins*, **58**, 321–328.
28. Meier,A. and Soding,J. (2015) Automatic prediction of protein 3D structures by probabilistic multi-template homology modeling. *PLoS Comput. Biol.*, **11**, e1004343.
29. Madera,M. (2008) Profile Comparer: a program for scoring and aligning profile hidden Markov models. *Bioinformatics*, **24**, 2630–2631.
30. Li,Y., Hu,J., Zhang,C., Yu,D. and Zhang,Y. (2019) ResPRE: high-accuracy protein contact prediction by coupling precision matrix with deep residual neural networks. *Bioinformatics*, **in press**.
31. Rego,N. and Koes,D. (2015) 3Dmol.js: molecular visualization with WebGL. *Bioinformatics*, **31**, 1322–1324.
32. Zhang,Y. and Skolnick,J. (2005) TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res.*, **33**, 2302–2309.
33. Moult,J., Fidelis,K., Kryshtafovych,A., Schwede,T. and Tramontano,A. (2014) Critical assessment of methods of protein structure prediction (CASP)–round x. *Proteins*, **82**(Suppl. 2), 1–6.
34. Zhu,J., Wang,S., Bu,D. and Xu,J. (2018) Protein threading using residue co-variation and deep learning. *Bioinformatics*, **34**, i263–i273.
35. Xu,J. (2018) Protein structure modeling by predicted distance instead of contacts. *CASP13 Abstract*, 146–147.
36. Binns,D., Dimmer,E., Huntley,R., Barrell,D., O'Donovan,C. and Apweiler,R. (2009) QuickGO: a web-based tool for Gene Ontology searching. *Bioinformatics*, **25**, 3045–3046.
37. Bairoch,A. (2000) The ENZYME database in 2000. *Nucleic Acids Res.*, **28**, 304–305.
38. Xu,J. and Zhang,Y. (2010) How significant is a protein structure similarity with TM-score = 0.5? *Bioinformatics*, **26**, 889–895.