

RNA Flexibility Prediction With Sequence Profile and Predicted Solvent Accessibility

Hong Wei¹, Boling Wang, Jianyi Yang¹, and Jianzhao Gao¹

Abstract—Structural flexibility plays an essential role in many biological processes. B-factor is an important indicator to measure the flexibility of protein or RNA structures. Many methods were developed to predict protein B-factors, but few studies have been done for RNA B-factor prediction. In this paper, we proposed a new method RNAbval to predict RNA B-factors using random forest. The method was developed using a comprehensive set of features, including the sequence profile and predicted solvent accessibility. RNAbval achieved an improvement of 9.2-20.5 percent over the state-of-the-art method on two benchmark test datasets. The proposed method is available at <http://yanglab.nankai.edu.cn/RNAbval/>.

Index Terms—RNA B-factor, random forest, RNA solvent accessibility

1 INTRODUCTION

THE atomic mean squared displacement or uncertainty in the X-ray scattering structure was measured by B-factor. The B-factor is used as an indicator to reflect the structural fluctuation of a biomolecule's X-ray crystallography structure. The higher the B-factor, the more flexible the biomolecule is. B-factor was widely used in the analysis of protein folding rate [1], protein disorder region [2], protein active sites [3], and protein thermal stability [4]. It is useful in understanding the structure and functions to predict the B-factors.

It is a great challenge to revolve the protein dynamics for experimental studies [5]. Molecular dynamics was used to correlate temperature B-factors. However, it is time consuming to do that. Computation methods to predict B-factors were needed. These methods include: normal-mode analysis [6], graph theory [7], mean-field theory [8]. Recently, machine learning-based methods were proposed: support vector machine (SVM) [9], [10], [11], neural network (NN) [12]. And multiscale weighted colored graphs (MWCGs), which integrates machine learning and graph theory [13]. These methods achieved moderate correlation coefficients to predict protein B-factors.

Although there are many methods to predict the protein B-factors, only two methods were proposed to predict the RNA B-factor. The first method was proposed by Tian et. al. [14], which predicted ribosomal RNA B-factor using the sequence and structure information with least squares support vector machine (LSSVM). However, the dataset used in Tian et. al. [14] was small, containing only 13 crystal structures of ribosomal 50S subunits. In addition,

no webserver or software is available for this method. Recently, A new method, RNAflex was proposed [15]. RNAflex was built on a bigger dataset and performed normalization for consistency of structural flexibility across different structure. RNAflex was built with SVM using sequence profile. The sequence profile was got by querying RNA sequences against RNA sequence library with BLASTN [16]. RNAflex achieved *Pearson* correlation coefficient (PCC) of about 0.51.

In this paper, we want to improve the prediction of B-factor by combing more sequence-based information. We fused the nucleotide acid one hot vector, sequence profile, predicted secondary structure and solvent accessibility. Our new model achieved significant improvements on two benchmark datasets.

2 METHODS

2.1 Dataset

The training and test datasets introduced in RNAflex [15] were used to build the model and evaluate the performance of our method. All structures were downloaded from the Protein Data Bank (PDB) [17]. The resolutions of the structures are $< 3.0 \text{ \AA}$ and the RNA chains contain more than 32 bases. The sequence redundancy was removed by CD-HIT-EST [18] with the cutoff 80 percent. The B-factor for each nucleotide is represented by atom type C1. The training dataset consists of 108 RNA chains, and test dataset consists of 34 chains. We denote this test dataset by *Test1*.

To further evaluate the model, we build a new test dataset following the procedure in RNAflex [15]. (1) RNA structures were downloaded from PDB. (2) The structures were filtered with resolution $< 3 \text{ \AA}$ and RNA chains > 32 bases. (3) We removed the redundancy of RNA chains between the downloaded sequences and training dataset and *Test1* using CD-HIT-EST [18] with a cutoff at 80 percent. (4) We kept RNA structures which were released after year of 2015. Finally, we got 60 chains with resolution $< 3 \text{ \AA}$ (denoted by *Test2*).

• The authors are with the School of Mathematical Sciences and LPMC, Nankai University, Tianjin 300071, China. E-mail: {weihong96, wangboling}@mail.nankai.edu.cn, {yangjy, gaojz}@nankai.edu.cn.

Manuscript received 16 July 2019; revised 11 Oct. 2019; accepted 22 Nov. 2019. Date of publication 28 Nov. 2019; date of current version 7 Oct. 2021.

(Corresponding author: Hong Wei.)

Digital Object Identifier no. 10.1109/TCBB.2019.2956496

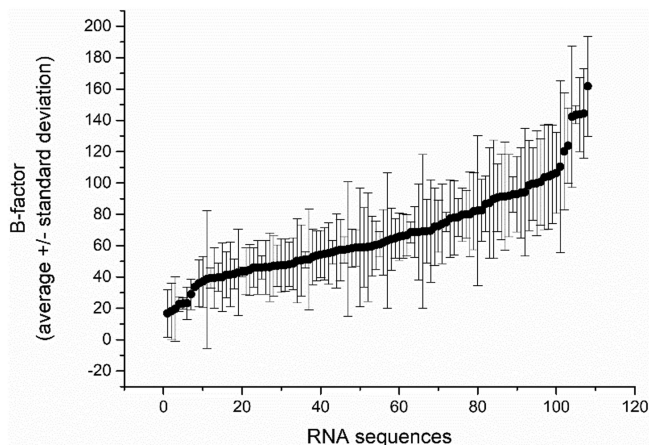


Fig. 1. The mean and standard deviation of B-factor values of each sequence in training dataset. The distributions of the B-factors are represented using the average and standard deviation (error bars).

2.2 Normalization of B-Factor

Normalization of B-factor is needed, since different temperatures and refinement procedures were used for structure determination. Fig. 1 shows that the averages and standard deviations of B-factors for each sequence in the training dataset. The maximal average B-factor (161.7) is 9.7 times the minimal average B-factor (16.7). It is necessary to normalize the B-factors for each structure before building model. First, we removed outliers using median-based approach [19] with the following formula:

$$M_i = 0.6745(x_i - x^*) / (\text{median of } |x_i - x^*|),$$

where x_i is the B-factor value of the i -th base in the chain. x^* is the median of x_i . The base is assigned as an outlier, if $|M_i|$ is greater than 3.5. We removed all outliers in the datasets. Second, all B-factors were normalized by the following formula:

$$B_i = (x_i - \mu) / \sigma,$$

where μ and σ are the mean and the standard deviation of the B-factor values, respectively. Table 1 shows the summary of the data sets.

2.3 Features

Different types of features were carefully designed. These features include:

One Hot Vector. Vector-based orthogonal codes were used. A, U, G, and C are represented by four-dimensional vectors: (1000), (0100), (0010) and (0001), respectively. These features were also used in RNAflex [15]. (4 values).

Sequence Profile. Sequence profile was obtained by querying the RNA sequences against NCBI's non-redundant nucleotide library (nt) library by BLASTN [16] with an E-value < 0.001 and a maximum of 50000 homologous sequences. To reduce the redundancy of alignment, the Henikoff weight [20] was used to weight each sequence in the multiple sequence alignment. The formula to calculate the weight of the k -th sequence is described by:

$$w_k = \frac{1}{L} * \sum_{i=1}^L w_{k,i} = \frac{1}{L} * \sum_{i=1}^L \frac{1}{n_k} * \frac{1}{f_{k,i}}, \quad (1)$$

TABLE 1
Summary of the Datasets Used in This Paper

Dataset	Resolution	#Chains	#Nucleotides	#Nucleotides after removing outliers
Training	$< 3.0 \text{ \AA}$	108	28059	26808
Test1	$< 3.0 \text{ \AA}$	34	6448	6015
Test2	$< 3.0 \text{ \AA}$	60	18599	10501

where L is the length of query RNA sequence; n_k is the number of nucleotide types at the i -th position of k -th sequence in multiple sequence alignment; $A_{k,i} = \{A, C, G, U\}$ is the nucleotide type in the k -th sequence at the i -th position; and $f_{k,i}$ is the number of occurrences for $A_{k,i}$. An example of how to compute Henikoff weight was provided in Table S1 in supplementary materials, which can be found on the Computer Society Digital Library at <http://doi.ieeeecomputersociety.org/10.1109/TCBB.2019.2956496>.

After that, the profile for each RNA sequence with dimension $L * 4$ was got. The frequency of the base type j at the position i , was defined by:

$$P_{ij} = -\log \left[\frac{(N_{ij} + s_{ij})}{\left(\sum_{i=1}^L (N_{ij} + s_{ij}) \right)} \right], \quad (2)$$

where N_{ij} is the number of observed based type j at position i ; the number N_{ij} is counted by the weight of each sequence; s_{ij} is a small number correction to avoid zero values, which is 9 if the base is the same type as the query base, and 0.3 otherwise; and L is the length of the RNA chain. (4 values).

RNA Accessible Surface Area (ASA). Accessible surface areas (ASA) is the surface area of a biomolecule that is accessible to the solvent. ASA values were predicted by the program RNAsnap [21] and RNAsol [22]. (2 values).

RNA Secondary Structure(SS). RNA secondary structure was predicted by RNAfold [23]. A nucleotide is encoded by 1 if it forms a base pair; and 0 otherwise. (1 value).

Normalized and Window-Based Features. Each feature was normalized to the range of $[-1, 1]$ using $[2(x - \min) / (\max - \min)] - 1$, where \max and \min are the maximum and minimum values of the original feature.

Sliding window was used to encode the RNA base to incorporate the effect of neighboring bases. If the base is at the terminal, the values were set to 0. The half of window size w was optimized on the training dataset with cross validation. The total number of features for each nucleotide is $(2w + 1) * 11$.

2.4 Machine Learning Methods and Metrics

Random Forest (RF). Random forest (RF) is an ensemble learning method. It builds multiple decision trees by combining the outputs of all trees. The number of decision trees were optimized. In this work, RF was implemented using the *scikit-learn* package [24].

Support Vector Machine (SVM). Support vector machine (SVM) was also considered for comparison. We implemented the model using LibSVM [25] with radial basis function (RBF) kernel. Support Vector Regression (SVR) was used to predict the real value of B-factor. The parameters C and γ were optimized by a grid search to achieve the best PCC with

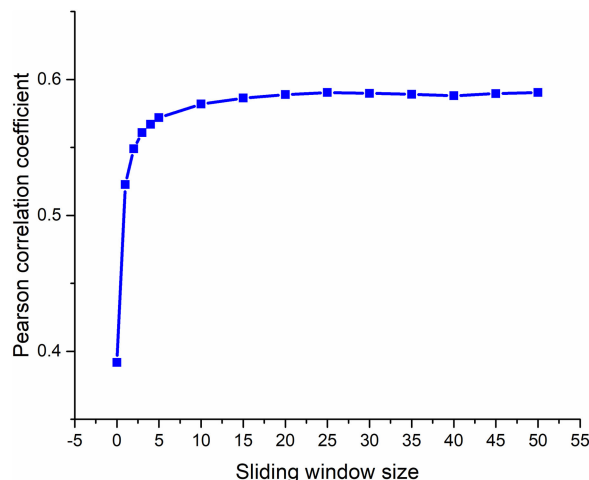


Fig. 2. The performance of RNAbval with different sizes of sliding window.

5-fold cross validation on training dataset. The optimized values for C and γ were 4, 0.00390625, respectively.

Ridge Regression (RR). Ridge regression (RR) is an effective regression method. The regularization parameter α is optimized. The final model is with regularization parameter α 1. We built it by the *scikit-learn* package [24].

Neural Network (NN). Neural network (NN) was implemented using the deep learning library Keras (<https://keras.io/>). Rectified Linear Unit (ReLU) was used as the activation function in each layer. We set the mean square error (MSE) as the loss function with the Adam optimizer. The parameters optimized include: the number of neurons in each fully connected layer and, batch size, dropout rate and epochs. The final model was a three-layer neural network. There are 512,256,128 neurons in each layer. The batch size is 50, the dropout rate is 0.3 and the epoch is 100.

Performance Evaluation. To measure the performance, the *Pearson* correlation coefficients (PCC) between the predicted values and the actual B-factor values were used.

3 RESULTS AND DISCUSSION

3.1 Optimized Sliding Window Size and Parameters

The final model was built using the random forest with the minimum number of samples in leaf nodes set as 1. The minimum number of samples in a node to stop the partition was 2 and the number of decision trees is 900. The size of the sliding windows is 25. We explored the hyperparameters in a wide range and found the above set of parameters yielded the highest PCC values. Fig. 2. shows the different performances of random forest with different sliding windows size. It shows that the model achieves the highest PCC value with the sliding window size of 25.

TABLE 2
The Performance of Different Algorithms on Training Dataset With Five-Fold Cross Validation

Model	PCC	sliding window size
Support vector machine	0.5402	50
Ridge regression	0.4331	50
Neural network	0.4799	50
RNAbval (Random forest)	0.5904	25

TABLE 3
The Feature Contribution in Five-Fold Cross Validation on the Training Set

Feature Group	PCC
All/ASA ^a	0.5186
All/profile	0.5508
All/Henikoff ^a	0.5764
All/ASA_RNAsol ^a	0.5775
All/ASA_RNAsnap ^a	0.5856
All/onehot	0.5889
All/SS	0.5897
All	0.5904

^a All/ASA means that all features were used to build the model except the ASA features from RNAsol and RNAsnap predictions; All/ASA_RNAsol (All/ASA_RNAsnap) means that all features were used to build the model except the ASA features from RNAsol (RNAsnap) predictions. All/SS means all features were used to build the model except the RNA secondary structure; All/Henikoff means the model was build using all features but without Henikoff weights.

We also try to build the model using other machine learning methods. Table 2 shows that results. Note that all the values reported in the table were got by methods with optimized parameters. The performances of SVM, Ridge regression, and neural network with different sliding window sizes were shown in Fig. S1 in supplementary materials available online. It shows that the random forest model achieved the highest PCC 0.5904 on the training dataset with the 5-fold cross validation. In the following sections, we refer our method as RNAbval.

3.2 Feature Contribution

To evaluate the contributions of each feature, we removed one group of features at one time, and calculated the PCC values on the training dataset with five-fold cross validation. Table 3 shows that the model with all features achieved the highest PCC value 0.5904. The removal of feature group decreases the PCC values. For example, the model without Henikoff weights achieved lower PCC than that the model with Henikoff weights. The biggest decrease was caused by removing the ASA feature group. It indicates that ASA is the most important feature in the model. The model without the ASA values from RNAsol achieved lower PCC than the model without the ASA values from RNAsnap. It shows that the RNAsol provides bigger contributions than the RNAsnap. The improvement of the proposed model is due to the new designed feature of predicted ASA values.

It may be arguing that the redundancy between the predicted ASA values from RNAsol and RNAsnap. We compute the PCC values between the predicted ASA values and actual ASA values for each RNA sequence in the training dataset. The actual ASA values were computed by POPs [26]. Fig. 3. shows the PCC values from RNAsnap and RNAsol. There are only 94 PCC values, because POPs can only outputs actual ASA values in PDB format. It shows that prediction from RNAsnap is quite different from the prediction of RNAsol. The points above the dash line means the RNA sequences where RNAsol achieved higher PCCs than that of RNAsnap. It indicates that the prediction of RNAsnap is complementary to the prediction of RNAsol. Our model get benefit from the combination of predictions from RNAsol and RNAsnap.

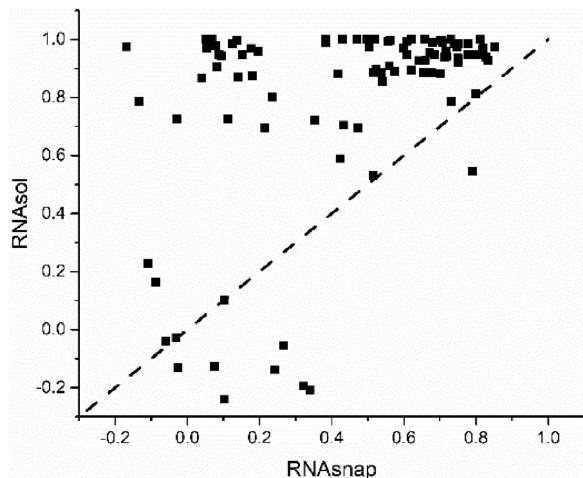


Fig. 3. The PCC values between the predicted ASA values by RNAsol/RNAsnap and actual ASA values. The actual ASA values were computed by POPs. The points above the dash line means that RNA sequences where RNAsol achieve higher PCC than that of RNAsnap. The points below the dash line means that RNA sequences where RNAsnap achieved higher PCC than that of RNAsol.

TABLE 4
Comparison With RNAflex on Training and Test Datasets

Methods	RNAflex	RNAbval (this work)	Improvement
Training ^a	0.5176 ^b	0.5904	14.1%
<i>Test1</i>	0.5028 ^b	0.6061	20.5%
<i>Test2</i>	0.4567 ^c	0.4986	9.2%

^aResults with 5-fold cross validation.

^bResults from RNAflex [15].

^cResults from the RNAflex webserver.

To further evaluate the ASA-based features, we compute PCC values between actual B-factors and predicted ASA values on the training dataset. The PCC between the predicted ASA values by RNAsnap and the actual B-factors is 0.3282, while the PCC between the predicted ASA values by RNAsol and actual B-factor is 0.3460. This suggests that RNAsol achieved higher correlation coefficient than that of RNAsnap. It is worth noting that the final model which combined all different types of features achieved higher PCC than just using part of features.

3.3 Comparison With Other Method

To compare the performance of proposed method with RNAflex, we tested our method on the training and test datasets (Table 4). On the training dataset, the proposed method achieved PCC 0.5904 with five-fold cross validation, while RNAflex achieved 0.5176. On the independent test dataset *Test1*, our method obtained higher PCC than that of RNAflex (0.6061 versus 0.5028). It indicates that the proposed method got improvement in both the training and the test datasets. It also shows that our method is robust that performances of proposed method in training set is similar to the test set. Fig. 4 shows the predicted B-factor values and actual B-factor values on the dataset *Test1*.

To further evaluate our method, we compared our method with RNAflex on the new test dataset *Test2*. Table 4 shows that our method achieved PCC 0.4986, which is higher than that of RNAflex (0.4567). RNAbval achieved an

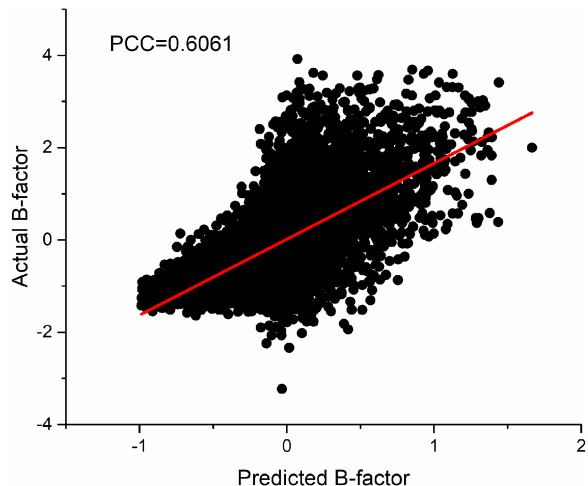


Fig. 4. The predicted and the real B-factors on the test dataset *Test1*.

TABLE 5
Comparison Results on Protein-Bound RNAs and Protein-Free RNAs

Complex	Protein-bound	Protein-free
RNAflex	0.56 ^a	0.11 ^a
RNAbval(this work)	0.6246	0.2416

^aResults from [15].

improvement of 9.2-20.5 percent over RNAflex on two test datasets. We further compare the average PCC in *Test2*. Our method achieved average PCC 0.2597 with standard deviation of 0.3185, while RNAflex achieved average PCC 0.2036 with standard deviation 0.2564. It shows that the advantage of the proposed method.

We further removed the redundancy between training dataset and test dataset using BLASTCLUST with cutoff 30 percent. There is no change in *Test1*. The number of RNA chains was decreased from 60 to 54 in *Test2*. The overall PCC value was changed from 0.4986 to 0.5052 on *Test2*. It shows that RNAbval achieved quite similar correlation when the redundancy cutoff was changed.

3.4 Results on the Protein-Bound RNAs and Protein-Free RNAs

To compare with RNAflex, we combined the cross validation and *Test1* results as [15] to get a larger dataset to analyze. There are two types of RNA structures in PDB: protein-bound and protein-free RNA. As shown in [21], [22], ASA prediction for protein-bound RNAs are significantly more accurate than and protein-free RNAs, probably because the former is more stable than the latter. Motivated by this observation, we also investigate the performance of our method on these two types of RNAs. In the combined dataset, 93 and 49 chains are protein-bound and protein-free RNAs, respectively. Table 5 shows the results on two different datasets. RNAflex achieved an overall PCC value 0.56 for the protein-bound RNAs and 0.11 for protein-free RNAs. While the proposed method RNAbval achieved 0.6246 and 0.2416 for protein-bound RNAs and protein-free RNAs. RNAbval obtained higher PCC values on the protein-bound RNAs than that of protein-free RNAs, consistent with the prediction of RNA ASA in [21], [22].

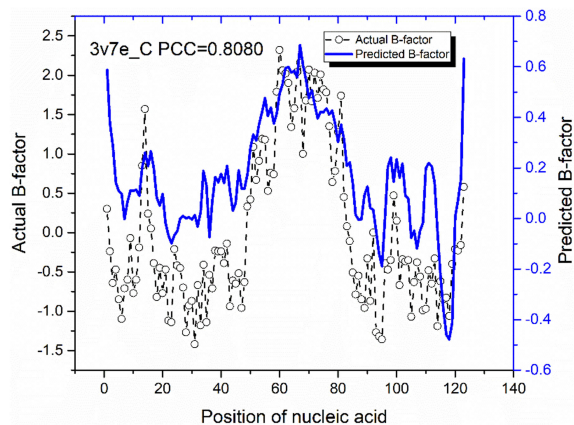


Fig. 5. Comparison of predicted and actual normalized B-factor profile of RNA (PDB 3V7E Chain C). Dashed line the actual B-factor, and solid line represents the predicted B-factor. The PCC value between the predicted and actual B-factor is 0.8080. The left y-axis gives actual B-factor and the right y-axis for predicted B-factor.

3.5 Case Study

Fig. 5. shows one example RNA from *Test1*. This is ribosomal RNA (PDB ID 3V7E chain C). The peak and valleys of the thermal fluctuations were reproduced quite accurately with PCC 0.8080.

4 CONCLUSION

RNA flexibility is important to study RNA function. Although there are many computational methods to predict protein B-factor, few methods were proposed to predict RNA B-factor. To address this issue, we proposed a new method, RNAbval to predict the RNA B-factor.

RNAbval used more RNA information. It combined RNA sequence profile, predicted RNA secondary structure, and predicted solvent accessibility. Comparisons with other state-of-the-art method shows that RNAbval achieves 9.2–20.5 percent improvement on the two test datasets.

We noticed that both RNAbval and RNAflex achieved lower PCC on the *Test2* than that on the *Test1*. RNA structures in *Test2* are newer ones than RNAs in the *Test1*. And the redundancy between the training dataset and *Test2* is less than 80 percent. It indicates that *Test2* is a hard test dataset to evaluate. More accuracy models are needed to improve the performance.

The proposed method achieved better method due to the new designed features. We showed that the predicted solvent accessibility is the most important feature in the model. We built a webserver for RNAbval, which is available at: <http://yanglab.nankai.edu.cn/RNAbval>.

ACKNOWLEDGMENTS

The authors thank Saisai Sun, for providing useful advice for building the webserver. This work was supported by the National Natural Science Foundation of China (NSFC) [Grant number 11701296, 11871290], the Natural Science Foundation of Tianjin [grant number 18JCQNJC09600], the Key Laboratory for Medical Data Analysis and Statistical Research of Tianjin (KLMDASR), Fok Ying-Tong Education Foundation (161003), China Scholarship Council, and the Fundamental Research Funds for the Central Universities.

REFERENCES

- [1] J. Gao *et al.*, "Accurate prediction of protein folding rates from sequence and sequence-derived residue flexibility and solvent accessibility," *Proteins: Struct. Function Bioinf.*, vol. 78, no. 9, pp. 2114–2130, 2010.
- [2] P. Radivojac *et al.*, "Protein flexibility and intrinsic disorder," *Protein Sci.*, vol. 13, no. 1, pp. 71–80, 2004.
- [3] O. Carugo and P. J. P. S. Argos, "Function, Accessibility to internal cavities and ligand binding sites monitored by protein crystallographic thermal factors," *Proteins: Struct. Function Bioinf.*, vol. 31, no. 2, pp. 201–213, 1998.
- [4] S. Parthasarathy and M. Murthy, "Protein thermal stability: Insights from atomic displacement parameters (B values)," *Protein Eng.*, vol. 13, no. 1, pp. 9–13, 2000.
- [5] D. A. Kondrashov, Q. Cui, and G. N. Phillips Jr, "Optimization and evaluation of a coarse-grained model of protein motion using x-ray crystal data," *Biophysical J.*, vol. 91, no. 8, pp. 2760–2767, 2006.
- [6] I. Bahar and A. J. Rader, "Coarse-grained normal mode analysis in structural biology," *Current Opinion Struct. Biol.*, vol. 15, no. 5, pp. 586–592, 2005.
- [7] D. J. Jacobs *et al.*, "Protein flexibility predictions using graph theory," *Proteins: Struct. Function Bioinf.*, vol. 44, no. 2, pp. 150–165, 2001.
- [8] B. Pandey *et al.*, "Protein flexibility prediction by an all-atom mean-field statistical theory," *Protein Sci.*, vol. 14, no. 7, pp. 1772–1777, 2005.
- [9] Z. Yuan, T. L. Bailey, and R. D. Teasdale, "Prediction of protein B-factor profiles," *Proteins: Struct. Function Bioinf.*, vol. 58, no. 4, pp. 905–912, 2005.
- [10] A. G. D. Brevern *et al.*, "PredyFlexy: Flexibility and local structure prediction from sequence," *Nucl. Acids Res.*, vol. 40, no. Web Server issue, pp. W317–W322, 2012.
- [11] P. Xiao-Yong and S. J. P. Hong-Bin, "Robust prediction of B-factor profile from sequence using two-stage SVR based on random forest feature selection," *Protein Peptide Lett.*, vol. 16, no. 12, pp. 1447–1454, 2009.
- [12] A. Yaseen *et al.*, "FLEXc: Protein flexibility prediction using context-based statistics, predicted structural features, and sequence information," *BMC Bioinf.*, vol. 17, no. 8, 2016, Art. no. 281.
- [13] D. Bramer and G. W. Wei, "Blind prediction of protein B-factor and flexibility," *J. Chem. Phys.*, vol. 149, no. 13, 2018, Art. no. 134107.
- [14] F. Tian *et al.*, "Predicting the flexibility profile of ribosomal RNAs," *Mol. Inf.*, vol. 29, no. 10, pp. 707–715, 2010.
- [15] I. Guruge *et al.*, "B-factor profile prediction for RNA flexibility using support vector machines," *J. Comput. Chemistry*, vol. 39, no. 8, pp. 407–411, 2018.
- [16] D. J. Lipman *et al.*, "Gapped BLAST and PSI-BLAST: A new generation of protein database search programs," *Nucl. Acids Res.*, vol. 25, no. 17, pp. 3389–3402, 1997.
- [17] G. Gilliland *et al.*, "The protein data bank," *Nucl. Acids Res.*, vol. 28, no. 1, pp. 235–242, 2000.
- [18] A. Godzik and W. Li, "Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences," *Bioinf.*, vol. 22, no. 13, pp. 1658–1659, 2006.
- [19] D. K. Smith *et al.*, "Improved amino acid flexibility parameters," *Protein Sci.*, vol. 12, no. 5, pp. 1060–1072, 2003.
- [20] S. Henikoff and J. G. Henikoff, "Position-based sequence weights," *J. Mol. Biol.*, vol. 243, no. 4, pp. 574–578, 1994.
- [21] Y. Yang *et al.*, "Genome-scale characterization of RNA tertiary structures and their functional impact by RNA solvent accessibility prediction," *RNA*, vol. 23, no. 1, pp. 14–22, 2017.
- [22] S. Sun *et al.*, "Enhanced prediction of RNA solvent accessibility with long short-term memory neural networks and improved sequence profiles," *Bioinf.*, vol. 35, no. 10, pp. 1686–1691, 2018.
- [23] R. Lorenz *et al.*, "ViennaRNA package 2.0," *Algorithms Mol. Biol.*, vol. 6, no. 1, 2011, Art. no. 26.
- [24] F. Pedregosa *et al.*, "Scikit-Learn: Machine learning in python," *J. Mach. Learn. Res.* vol. 12, no. 10, pp. 2825–2830, 2011.
- [25] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, 2011, Art. no. 27.
- [26] C. Luigi, K. Jens, and F. Franca, "POPS: A fast algorithm for solvent accessible surface areas at atomic and residue level," *Nucl. Acids Res.*, vol. 31, no. 13, pp. 3364–3366, 2003.



Hong Wei received the BS degrees in mathematics and applied mathematics from Nankai University, Tianjin, China, in 2017. She is currently working toward the MS degree in bioinformatics at Nankai University. Her research interests include machine learning, bioinformatics, and quality assessment of protein structure model.



Boling Wang received the BS degrees in applied statistics from the Tianjin Polytechnic University, Tianjin, in 2018. She is currently working toward the MS degree in the School of Mathematical Sciences at Nankai University, Tianjin, China. Her research interests include protein structure prediction, machine learning, and deep learning applications in bioinformatics.



Jianyi Yang received the PhD degree from Nanyang Technological University and had his postdoctoral training in the University of Michigan. He is a professor with the School of Mathematical Sciences, Nankai University, Tianjin, China. His research interests include protein structure and function prediction, protein structure alignment, and RNA structure prediction. He has made significant contributions to the development of many widely used tools, including I-TASSER, COACH, COACH-D, BioLiP, mTM-align, and so on. More details can be found at his lab web site: <http://yanglab.nankai.edu.cn/>.



Jianzhao Gao received the PhD degree in bioinformatics from Nankai University, Tianjin, China, in 2010. From 2010 to 2016, he was an assistant professor with the School of Mathematical Sciences, Nankai University. Since 2017, he has been an associate professor of bioinformatics in the School of Mathematical Sciences, Nankai university. His research interests include bioinformatics, protein structure, and function prediction. More details can be found at his lab web site: <http://www.biomath.cn/>.

▷ For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/csdl.