Supporting Information

TargetDBP+: Enhancing the Performance of Identifying DNA-Binding Proteins via Weightedly Convolutional Features

Jun Hu [†,‡,*], Liang Rao [†], Yi-Heng Zhu [¶], Gui-Jun Zhang [†,*] and Dong-Jun Yu [¶,*]

[†] College of Information Engineering, Zhejiang University of Technology, Hangzhou, 310023, P. R. China, [‡] Key Laboratory of Data Science and Intelligence Application, Fujian Province University, Zhangzhou, 363000, China, and [¶] School of Computer Science and Engineering, Nanjing University of Science and Technology, Xiaolingwei 200, Nanjing, 210094, P. R. China

*Email: junh_cs@126.com, zgj@zjut.edu.cn, or njyudj@njust.edu.cn

Brief summary

This Supporting Information contains two Supporting Texts, i.e., Text S1 and S2, six Supporting Tables, i.e., Table S1, S2, S3, S4, S5, and S6, and one Supporting Figure, i.e., Figure S1.

Supporting Texts

Text S1: *UniSwiss* dataset

The *UniSwiss* dataset contains two subsets, i.e., the training dataset (*UniSwiss-Tr*) and independent validation dataset (*UniSwiss-Tst*). Both *UniSwiss-Tr* and *UniSwiss-Tst* include many DBPs (DNA-binding proteins) and non-DBPs (non-DNA-binding proteins). Here, the set consisting of DBPs in *UniSwiss-Tr* is denoted as *Tr-P* and the set consisting of non-DBPs in *UniSwiss-Tr* is named as *Tr-N*. Similarity, the set consisting of DBPs and the set consisting of non-DBPs in *UniSwiss-Tst* are named as *Tst-P* and *Tst-N*, respectively.

To construct the union set (denoted as *P*) of *Tr-P* and *Tst-P*, the 32,890 DBPs in the UniProtKB/Swiss-Prot[1] (up to October 5, 2020) database are first downloaded at https://www.uniprot.org/keywords/KW-0238 freely. Then, the CD-hit software[2] is used to remove the redundant protein sequences, such that the sequence identity between any two remained sequences is below 25%. To ensure no fragment in the final dataset, any sequence with less than 50 residues in length is removed. Finally, a total of 4,881 non-redundant protein sequences are gained to form *P*. We randomly selected 4,500 sequences from *P* to compose *Tr-P*. The remaining 381 sequences in *P* are used to construct *Tst-P*.

Similarity, in order to construct *Tr-N* and *Tst-N*, all non-DBPs in UniProtKB/Swiss-Prot (up to October 5, 2020) are first collected. Then, the CD-hit software[2] is used to remove the redundant protein sequences, such that the sequence identity between any two remained sequences is below 25%. To ensure no fragment in the final dataset, any sequence with less than 50 residues in length is removed. We denoted the remaining non-redundant non-DBPs as *TotalNega*. Finally, we randomly selected 4,500 and 381 non-DBPs from *TotalNega* to compose *Tr-N* and *Tst-N*, respectively.

After *Tr-P*, *Tst-P*, *Tr-N*, and *Tst-N* are collected, UniSwiss, including *UniSwiss-Tr* and UniSwiss-Tst, could be easily constructed. The *UniSwiss* dataset is downloadable at https://github.com/jun-csbio/TargetDBPplus/.

Text S2: Empirically tune the parameter *k* of the convolutional sliding weight window

To date, there is no good way to obtain the optimal value of *k* of the convolutional sliding weight window **w** and the values of its elements. In addition, it is impractical to determine the parameters of **w** via exhaustive search. In this study, to simply and empirically tune the optimal size of **w**, for each potential value of *k*, the value of the middle element of **w** is $\frac{k+1}{2k+1}$ and the value of each other element of **w** is $\frac{1}{2(2k+1)}$. Table S6 demonstrates the results of *k*=0, *k*=1, *k*=2, and *k*=3 under the condition of *G*=1 over ten-fold cross-validation on the *UniSwiss-Tr* dataset. Note that, as described in the section of "Convolutional feature representation", *G* is another parameter of generating the convolutional feature representation.

From Table S6, we can easily find that *k*=1 achieves the highest *MCC* value (0.676). The values of *Spe*, *Acc*, *Pre*, *MCC*, and *AUC* of *k*=1 are 2.45%, 0.05%, 1.94%, 0.45%, and 0.22% higher than that of *k*=0, 1.98%, 0.16%, 1.61%, 0.60%, and 0.11% higher than that of *k*=2, and 1.78%, 0.43%, 1.54%, 1.20%, and 0.55% higher than that of *k*=3, respectively. Hence, in this study, we set *k* to be 1, which means the size of **w** is 3.

However, the optimal values of three elements of **w** should be further fine-tuned. Hence, in the section of "Comparison performance of different convolutional sliding weight windows", we have testified the performance of DBP identification on eight different convolutional sliding weight windows, i.e., $\mathbf{w}_{0.1.0}$=(0/1, 1/1, 0/1), $\mathbf{w}_{1.1.1}$=(1/3, 1/3, 1/3), $\mathbf{w}_{1.4.1}$=(1/6, 4/6, 1/6), $\mathbf{w}_{1.5.1}$=(1/7, 5/7, 1/7), $\mathbf{w}_{1.9.1}$=(1/11, 9/11, 1/11), $\mathbf{w}_{1.13.1}$=(1/15, 13/15, 1/15), $\mathbf{w}_{1.17.1}$=(1/19, 17/19, 1/19), and $\mathbf{w}_{1.21.1}$=(1/23, 21/23, 1/23), over ten-fold cross-validation and jackknife tests on the training dataset *UniSwiss-Tr* under the condition of *G*=1. Finally, we find that $\mathbf{w}_{1.9.1}$ is a suitable choice.

Supporting Tables

Table S1. The *p*-values in Wilcoxon signed rank *t*-test for the difference in the outputted probabilities of belonging to the class of DBPs between different convolutional sliding weight windows on *UniSwiss-Tr* over 10-fold cross-validation tests

#	$\mathbf{w}_{0.1.0}$	$\mathbf{w}_{1.1.1}$	$\mathbf{w}_{1.4.1}$	$\mathbf{w}_{1.5.1}$	$\mathbf{w}_{1.9.1}$	$\mathbf{w}_{1.13.1}$	$\mathbf{w}_{1.17.1}$	$\mathbf{w}_{1.21.1}$
$\mathbf{w}_{0.1.0}$		0.074812	0.000284	0.068093	0.957944	0.486446	0.428836	0.106718
$\mathbf{w}_{1.1.1}$			0.074812	0.118644	0.142869	0.460522	0.126363	0.060573
$\mathbf{w}_{1.4.1}$				0.042765	0.808313	0.767795	0.727749	0.128892
$\mathbf{w}_{1.5.1}$					0.136858	0.079936	0.054771	0.316478
$\mathbf{w}_{1.9.1}$						0.915907	0.035607	0.627702
$\mathbf{w}_{1.13.1}$							0.063894	0.29176
$\mathbf{w}_{1.17.1}$								0.435247
$\mathbf{w}_{1.21.1}$								

Table S2. The *p*-values in Wilcoxon signed rank *t*-test for the difference in the outputted probabilities of belonging to the class of DBPs between different convolutional sliding weight windows on *UniSwiss-Tr* over jackknife tests

#	$\mathbf{w}_{0.1.0}$	$\mathbf{w}_{1.1.1}$	$\mathbf{w}_{1.4.1}$	$\mathbf{w}_{1.5.1}$	$\mathbf{w}_{1.9.1}$	$\mathbf{w}_{1.13.1}$	$\mathbf{w}_{1.17.1}$	$\mathbf{w}_{1.21.1}$
$\mathbf{w}_{0.1.0}$		0.068154	0.941102	0.254655	0.398892	0.719946	0.857721	0.250467
$\mathbf{w}_{1.1.1}$			0.123748	0.221399	0.026848	0.460572	0.277605	0.13995
$\mathbf{w}_{1.4.1}$				0.049683	0.286667	0.364314	0.291492	0.005032
$\mathbf{w}_{1.5.1}$					0.711903	0.048381	0.392681	0.597978
$\mathbf{w}_{1.9.1}$						0.004206	0.004969	0.688666
$\mathbf{w}_{1.13.1}$							0.075322	0.47994
$\mathbf{w}_{1.17.1}$								0.486446
$\mathbf{w}_{1.21.1}$								

Table S3. Performance comparisons of different values of *G* on *UniSwiss-Tr* over ten-fold cross-validation test.

G	*Sen*	*Spe*	*Acc*	*Pre*	*MCC*	F_1	*AUC*
1	82.42	86.20	84.31	85.66	0.687	0.840	0.921
2	84.33	85.00	84.67	84.90	0.693	0.846	0.924
3	79.53	90.42	84.98	89.25	0.704	0.841	0.923
4	81.31	88.71	85.01	87.81	0.702	0.844	0.923
5	80.24	89.07	84.66	88.01	0.696	0.839	0.921
6	81.04	89.16	85.10	88.20	0.704	0.845	0.922
7	81.13	88.64	84.89	87.72	0.700	0.843	0.921
8	79.67	90.56	85.11	89.40	0.706	0.843	0.925
9	81.49	89.60	85.54	88.68	0.713	0.849	0.930
10	82.78	88.18	85.48	87.50	0.711	0.851	0.928
11	82.62	88.58	85.60	87.85	0.713	0.852	0.930
12	84.58	86.02	85.30	85.82	0.706	0.852	0.926
13	80.98	89.09	85.03	88.13	0.703	0.844	0.924
14	86.02	84.64	85.33	84.85	0.707	0.854	0.925
15	82.16	89.16	85.66	88.34	0.715	0.851	0.929
16	78.80	91.84	85.32	90.62	0.713	0.843	0.929
17	79.67	91.36	85.51	90.21	0.715	0.846	0.929
18	85.82	85.87	85.84	85.86	0.717	0.858	0.931
19	79.67	91.33	85.50	90.19	0.715	0.846	0.929
20	79.24	91.84	85.54	90.67	0.717	0.846	0.930
21	80.36	90.73	85.54	89.66	0.715	0.848	0.929

Table S4. Performance comparisons of different values of *G* on *UniSwiss-Tr* over jackknife test.

G	*Sen*	*Spe*	*Acc*	*Pre*	*MCC*	F_1	*AUC*
1	82.42	86.69	84.56	86.10	0.692	0.842	0.923
2	84.67	85.64	85.16	85.50	0.703	0.851	0.926
3	80.07	90.42	85.24	89.32	0.709	0.844	0.926
4	81.09	89.44	85.27	88.48	0.708	0.846	0.926
5	80.84	89.16	85.00	88.17	0.702	0.843	0.924
6	82.42	88.38	85.40	87.64	0.709	0.850	0.925
7	83.22	87.38	85.30	86.83	0.707	0.850	0.925
8	80.44	90.38	85.41	89.32	0.712	0.846	0.927
9	85.47	86.27	85.87	86.16	0.717	0.858	0.933
10	82.93	88.69	85.81	88.00	0.717	0.854	0.931
11	85.60	86.33	85.97	86.23	0.719	0.859	0.932
12	80.93	89.98	85.46	88.98	0.712	0.848	0.927
13	81.73	89.13	85.43	88.26	0.711	0.849	0.927
14	80.78	90.36	85.57	89.33	0.715	0.848	0.927
15	81.62	90.16	85.89	89.24	0.720	0.853	0.932
16	82.51	88.73	85.62	87.99	0.714	0.852	0.927
17	81.58	90.38	85.98	89.45	0.722	0.853	0.932
18	84.24	88.00	86.12	87.53	0.723	0.859	0.932
19	84.22	88.00	86.11	87.53	0.723	0.858	0.932
20	85.91	85.96	85.93	85.95	0.719	0.859	0.932
21	82.20	89.84	86.02	89.00	0.723	0.855	0.932

Table S5. The *p*-values in Wilcoxon signed rank *t*-test for the difference in the outputted probabilities of belonging to the class of DBPs between different feature types on *UniSwiss-Tr* over jackknife tests

Feature	PseConvAAOHM	PseConvPSSM	PseConvPSSPM	PseConvPSAPM	PseConvPPDBS	SerialCom	WSerialCom
PseConvAAOHM		6.65E-14	0.170563	0.416992	2.59E-09	1.18E-08	2.00E-13
PseConvPSSM			8.57E-22	2.26E-22	0.370264	7.45E-06	1.00E-24
PseConvPSSPM				2.38E-03	2.59E-09	1.31E-20	6.00E-29
PseConvPSAPM					7.07E-05	9.12E-27	5.00E-29
PseConvPPDBS						0.429195	0.070000
SerialCom							1.00E-65
WSerialCom							

Table S6. Performance comparisons of different values of *k* over ten-fold cross-validation on the *UniSwiss-Tr* dataset under the condition of *G*=1.

k	*Sen*	*Spe*	*Acc*	*Pre*	*MCC*	F_1	*AUC*
0	81.20	86.07	83.63	85.35	0.673	0.832	0.915
1	79.16	88.18	83.67	87.01	0.676	0.829	0.917
2	80.62	86.47	83.54	85.63	0.672	0.830	0.916
3	79.98	86.64	83.31	85.69	0.668	0.827	0.912

Supporting Figures

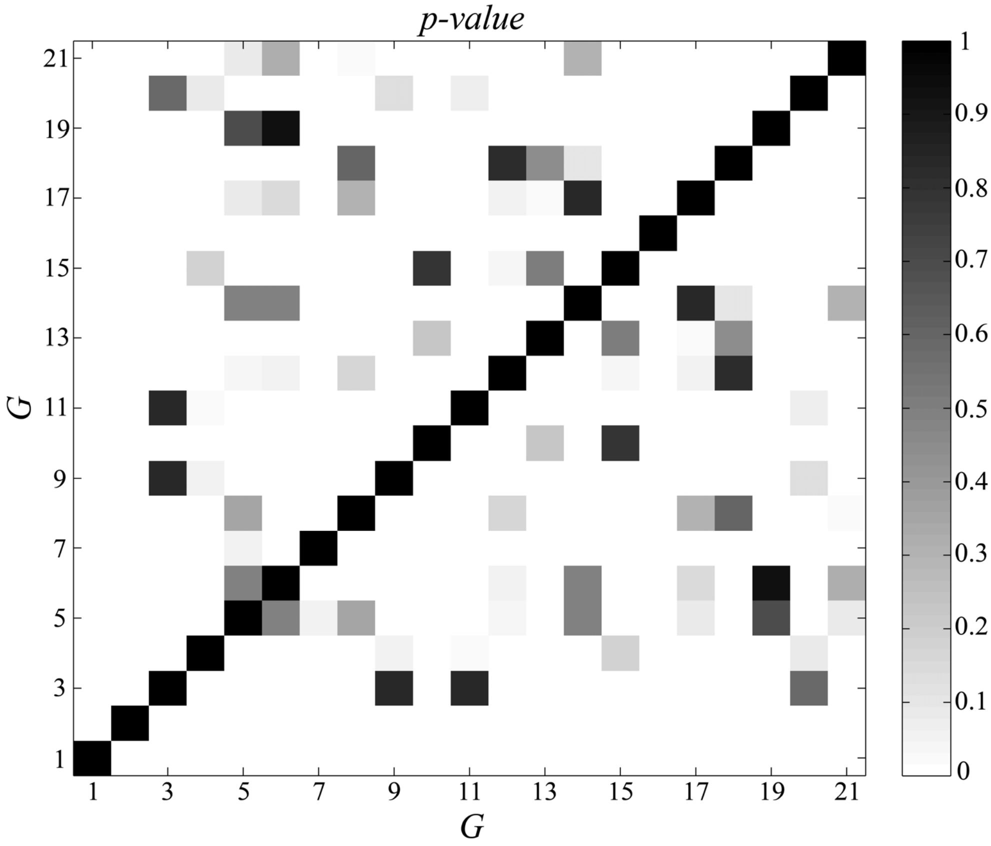

Figure S1. The *p*-values in Wilcoxon signed rank *t*-test for the difference in the outputted probabilities of belonging to the class of DBPs between different values of *G* on *UniSwiss-Tr* over jackknife tests

References

(1) Boutet, E.; Lieberherr, D.; Tognolli, M.; Schneider, M.; Bansal, P.; Bridge, A. J.; Poux, S.; Bougueleret, L.; Xenarios, I. UniProtKB/Swiss-Prot, the manually annotated section of the UniProt KnowledgeBase: how to use the entry view. *Plant bioinformatics: methods and protocols* **2016**, 23-54.

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