Prediction of TF-binding site by inclusion of higher order position dependencies

Jiyun Zhou*†, Qin Lu†, Ruifeng Xu*‡, Lin Gui*, Hongpeng Wang*
*School of Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, China †Department of Computing, Hong Kong Polytechnic University, Hung Hom, Hong Kong
Email addresses: zhoujiyun2010@gmail.com, csluqin@comp.polyu.edu.hk, xuruifeng@hit.edu.cn, guilin.nlp@gmail.com, wanghp@hit.edu.cn
‡Corresponding author

***Abstract*—Most proposed methods for TF-binding site (TFBS) predictions only use low order dependencies for predictions due to the lack of efficient methods to extract higher order dependencies. In this work, We first propose a novel method to extract higher order dependencies by applying CNN on histone modification features. We then propose a novel TFBS prediction method, referred to as CNN_TF, by incorporating low order and higher order dependencies. CNN_TF is first evaluated on 13 TFs in the mES cell. Results show that using higher order dependencies outperforms low order dependencies significantly on 11 TFs. This indicates that higher order dependencies are indeed more effective for TFBS predictions than low order dependencies. Further experiments show that using both low order dependencies and higher order dependencies improves performance significantly on 12 TFs, indicating the two dependency types are complementary. To evaluate the influence of cell-types on prediction performances, CNN_TF was applied to five TFs in five cell-types of humans. Even though low order dependencies and higher order dependencies show different contributions in different cell-types, they are always complementary in predictions. When comparing to several state-of-the-art methods, CNN_TF outperforms them by at least 5.3% in AUPR.**

***Index Terms*—Protein-DNA interaction, TF-binding site, CNN, Transcription factor, low order dependency, higher order dependency.**

I. Introduction

Gene expression is mainly regulated by interactions between DNA and transcription factors (TFs) [1]. So predictions of TF-binding sites (TFBSs) are important for understanding transcriptional regulatory networks and crucial in understanding fundamental cellular processes [2]. Two experimental techniques have been developed for TFBS identifications: chromatin immunoprecipitation followed by high throughput sequencing (ChIP-Seq) [3] and chromatin immunoprecipitation followed by array hybridization (ChIP-chip) [3]. These two technologies have been successfully used to map TF binding locations for many organisms. However, the lacking of antibodies for many TFs and the high expense have made them be useful only for a limited number of TFs. Therefore, computational methods are urgently required for TFBS identifications.

‡ Harbin Institute of Technology Shenzhen Graduate School, HIT Campus Shenzhen University Town, Xili, Shenzhen, 518055, China; Phone: (+86) 0755-26033283.

TFBSs are generally short and often degenerate sequence motifs [4] such that they are computationally difficult to be predicted at a genomic scale. The TFBSs of a TF can be represented by a consensus sequence and a position weight matrix (PWM) [5]. The consensus sequence representation is easy to visually interpret TFBSs. However, variations of the nucleotide composition at each position in TFBSs make the consensus sequence representation an unsuitable approach for TFBS representations [6, 7]. So many classical computational methods used PWMs to represent TFBSs [5]. A PWM is often derived from a set of aligned and functionally related sequences. The basic assumption for PWM is that the positions within a TFBS are independent between each other. However, position dependencies within TFBSs are observed in many studies including crystal structure analyses [8] and a biochemical study [9]. Tomovic and Oakeley analyzed the number of TFBSs with dependent positions by using three statistical tests and attempted to extracted evidences of position dependencies from TF-DNA crystal structures [10]. Their conclusion is that some TFs indeed show evidences of position dependencies. Based on Tomovic and Oakeley's finding, Zare-Mirakabad et al. proposed a scoring function by including dependencies between all positions in TFBSs [11]. The joint information content and the mutual information are used to measure dependencies in the scoring function. Evaluations show that including position dependencies indeed obtains performance gains. Furthermore, Siddharthan proposed dinucleotide weight matrix (DWM) to extend PWM by including dependencies between neighbor positions within a TFBS [12]. In addition to DWM, TFFM proposed by Mathelier and Wasserman can also capture dependencies between neighbor positions for predictions [13], in which the state transition probabilities in a hidden markov (HMM) model [14] are used to model the dependencies between neighbor positions.

Although DWM [12] and TFFM [13] can capture pairwise dependencies between neighbor positions, they cannot capture multiple dependencies among positions. As we know, histone modification features are post-translational modification levels of histones on chromatin structures. Histone modification features are DNA fragment features and span multiple positions, so they can capture multiple dependencies among positions. Several studies [15–17] have shown that TFBSs are associated with different histone modification features from non-TFBSs,

so they proposed novel prediction methods by incorporating histone modification features. Talebzadeh and Zare-Mirakabad [18] developed a method by combining two sets of histone modification features. Won at al. [19] proposed a HMM based method called Chromia, in which both histone modification features and sequence features are used for learning feature representations. Recently, Tsai et al. [15] examined respective contributions of sequence features, histone modification features, and structure features for TFBS predictions by using a random forest model [20] and concluded that all the three feature types are useful. Recent studies also suggested that DNA shape features are another important type of features for TFBS predictions [21]. DNA shapes represent the 3D structure of DNA fragments and span multiple positions, so shape features can extract multiple dependencies among positions. Methelier [21] proposed a method by using DNA shape features and demonstrated that DNA shape features indeed play important roles in TFBS predictions.

In addition to the DNA shape based method, several deep learning methods have been proposed for TFBS predictions. DeepBind [22], DeepSEA [23] and DanQ [24] are three representative methods. DeepBind [22] was proposed by Alipanahi et al. (2015) by applying Convolutional Neural Network (CNN) to DNA sequence features. DeepSEA [23] was proposed by Zhou and Troyanskaya (2015) by combining CNN and a multi-task learning method to learn representations for putative TFBSs. DanQ [24] is an improved model of DeepSEA, which was proposed by Quang and Xie (2016) by applying combined use of CNN and Recurrent neural network (RNN) to sequence features to learning representations for TFBSs. The multi-task learning method in both DeepSEA and DanQ contains 919 prediction tasks including 690 TFBS prediction tasks, 104 histone modification peak prediction tasks, 125 DNase Ihypersensitive sites (DHSs) prediction tasks.

Position dependencies have been proposed by works of Tomovic and Oakeley [10] and Zare-Mirakabad et al. [11] and used by methods including DWM [12], TFFM [13], Chromia [17], and the methods based on DNA shapes [21]. Position dependencies proposed in these literatures are non-independent influence among positions that spans a few base pairs and are referred to as higher order dependencies. However, the higher order dependencies in these literatures only span a few base pairs, for example, the higher order dependencies extracted in Tomovic and Oakeley's work [10] span only 12 base pairs on average. DeepBind by Alipanahi et al. [22] and DeepSEA by Zhou and Troyanskaya [23] attempted to capture higher order dependencies using more convolution layers. However, as Zeng et al. [25] pointed out that the performance of deep convolutional neural networks tends to decline when more convolution layers are used. In other words, learning higher order dependencies through higher convolution layers have very limited power to improve performance for predictions. DanQ reported by Quang and Xie [24] extracts higher order dependencies by combining RNN and CNN. However, DanQ [24] requires a lot of computer resources and it cannot clear the captured higher order dependencies, which are very important to understand TF-DNA interactions.

Histone modification features are DNA fragment features and already contain position dependencies. In this work, we extract position dependencies spanning more base pairs by extracting dependencies among histone modification features. As CNN is quite suited in extracting dependencies from a sequence [26], we propose a novel method, referred to as CNN_TF, to extract higher order dependencies by applying CNN to histone modification features. **Higher order dependencies** are position dependencies that spans a number of base pairs. As histone modification features contain position dependencies spanning more base pairs than the position dependencies proposed in literatures [10–13], CNN_TF can extract higher order dependencies at a larger scale than the methods presented in the literature[10–13]. In addition, our proposed CNN_TF also contains a CNN to extract position dependencies from sequence features. In contrast to higher order dependencies, position dependencies extracted from sequence features span fewer base pairs. So, we refer to the position dependencies extracted by CNN_TF from sequence features as **low order dependencies**. The extracted higher order dependencies and low order dependencies are used in combination for predictions. The resource and executable code is freely available at http://www.hitsz-hlt.com:8080/CNNTF/ and http://hlt.hitsz.edu.cn/CNNTF/.

II. Method and materials

According to recently published works [27–29], a complete prediction model in bioinformatics should contain five basic components: a validation benchmark dataset(s), an effective feature extraction procedure, an efficient predicting algorithm, a set of fair evaluation criteria and fair comparisons with state-of-the-art methods. In this section, the definition of TFBSs for our prediction task will be introduced first. Then, details of the five components of our proposed CNN_TF will be described in sequence.

A. TF binding sites (TFBSs)

Most studies used the ChIP-seq experiments [3] to identify TFBSs. The ChIP-seq experiments provide a peak for each TFBS. The obtained peaks can be provided in one of two formats. One is called **narrow peak** and the other is called **broad peak**. Both formats provide the chromosome, the start position, the end position and the signal for every peak. The narrow peak format, which requires technically more sophisticated equipments to get, can provide more accurate positions for TFBSs than the broad peak format. However, some datasets are provided in only the broad peak format. In this work, the narrow peak format will be used whenever available. Otherwise, the broad peak format will be used. The peaks in both of the two formats can be used to locate TFBSs. Although TFBSs are short and often degenerate sequence motifs, their contexts related to their functions may still contain many base pairs. The context of TFBSs in promoters contains 100 to 1000 base pairs and that of TFBSs in enhancers contains 50 to 1500 base pairs, respectively. In order to incorporate context information into predictions, TFBSs should be defined as sequences containing both the peaks and their context. Based on the study completed by Won et al. [17], we define a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2019.2892124, IEEE/ACM Transactions on Computational Biology and Bioinformatics

TFBS as a 2000-bp DNA segment for each peak by taking the midpoint of the peak as the center of the observation window. For a peak with the midpoint at the position i in the genome, the TFBS can be defined as

$$T_i = N_{i-999}N_{i-998}\cdots N_{i-1}N_iN_{i+1}\cdots N_{i+999}N_{i+1000} \quad (1)$$

where N_i denotes the nucleotide at the position i. Contrast to TFBSs, non-TFBSs are defined as 2000 bp DNA fragments which cannot be bound by the target TF. Therefore, TFBS predictions are defined as a binary classification problem. The input of the problem are 2000-bp DNA sequences and the output are whether the input DNA sequences are TFBSs or non-TFBSs.

B. Datasets

Two sets of datasets are used to evaluate CNN_TF: 13 TFs in the mES cell and 5 TFs in 5 cell-types of humans.

13 TFs in the mES cell: 13 TFs in the mouse embryonic stem (mES) cell have been widely used by multiple TFBS prediction methods: *Zfx, CTCF, c-Myc, n-Myc, E2f1, Esrrb, Klf4, Tcfcp2l1, Nanog, Oct4, Smad1, Sox2*, and *STAT3* [30, 31]. For these TFs, the TFBSs are obtained by ChIP-seq experiments [30, 31] and the peaks can be obtained from a literature [32] and our web server freely. The 2000-bp sequences are non-TFBSs if and only if they are do not overlap with each other nor overlap with the known TFBSs. As we can only get the histone modification features for 18 autosomes and the X chromosome, the TFBSs and the non-TFBSs of only these 19 chromosome are used to evaluate our method. The number of TFBSs and non-TFBSs of these 19 chromosomes are listed in Table S1, which is made available on our website.

5 TFs in 5 cell types of humans: In order to evaluate the influence of different cell-types on the performance of our method, we evaluate CNN_TF by a recent set of datasets collected by the Gene Expression Omnibus (GEO) [33]. In this set, five dissimilar TFs are selected: *CTCF, JunD, REST, GABP* and *USF2* and five dissimilar cell-types are selected: *GM12878, H1-hESC, HeLa-S3, HepG2* and *K562*. These five cell-types are chosen because they represent diverse classes of cell-types and the ChIP-seq peaks of the five TFs in them are available. For these TFs, the TFBSs are obtained by ChIP-seq experiments [30, 31] and the peaks can be obtained from a literature [16] and our web server freely. Similarly, the 2000-bp sequences, which are nonoverlapping with each other and nonoverlapping with the known TFBSs, are considered as non-TFBSs. As most TFs do not have TFBSs on the Y chromosome, we use the TFBSs and the non-TFBSs on the 22 autosomes and the X chromosome to evaluate our method. The number of TFBSs and non-TFBSs of the 23 chromosomes are listed in Table S2, which is made available on our website.

C. Feature representation

Both sequence features and histone modification features are important features of TFBSs. **Sequence features** of a TFBS are defined by the nucleotides within it and can be calculated by concatenating the one-hot vector of these nucleotides. So sequence features of the TFBS T_i can be represented as a feature matrix with dimension of 4×2000 as follows

$$S_{T_i} = [O(N_{i-999}), \cdots, O(N_i), \cdots, O(N_{i+1000})] \quad (2)$$

where $O(N_i)$ denotes the one-hot vector of the nucleotide N_i. **Histone modification features** refer to the post-translational modification levels of histones in chromatin structures. In this study, 8 types of histone modification features are used for the TFs in the mES cell: *H3, H3K4me1, H3K4me2, H3K4me3, H3K9me3, H3K36me3, H3K20me3*, and *H3K27me3*. The ChIP-seq data for these histone modification features can be obtained from literatures [34, 35] and our web server freely. For the 5 TFs in 5 cell-types of humans, 7 types of histone modification features are used: *H3K4me2, H3K4me3, H4K20me1, H3K9ac, H3K27ac, H3K27me3* and *H3K36me3*. The ChIP-seq data for these histone modification features can be obtained from a work [16] and our web server freely. According to Won et al's study [17], as the resolution for the ChIP-seq experiments is 25-bp, 25-bp bin is used as the unit to measure histone modification features. First, the histone modification features for each 25-bp bin are estimated by calculating the number of ChIP reads overlapping the bin. And then, the histone modification features for each 100-bp bin are calculated by averaging the histone modification features of the four 25-bp bins within it. So the histone modification features for the TFBS T_i can be represented as

$$C_{T_i} = [H(N_{i-999}, \cdots, N_{i-898}), \cdots, H(N_{i-99}, \cdots, N_i), \cdots, H(N_{i+901}, \cdots, N_{i+1000})] \quad (3)$$

where $H(\cdot)$ denotes the histone modification features over 100-bp bins. The TFBSs of the TFs in mES cell can be represented as feature matrices with dimension of 8×20 while the TFBSs of the TFs in the five cell-types of humans can be represented as feature matrices with dimension of 7×20.

D. Convolutional neural network(CNN)

In this work, we propose to apply CNN to sequence features and histone modification features to extract low order dependencies and higher order dependencies, respectively . This is where the name of our method CNN_TF is derived from. Then the extracted low order dependencies and higher order dependencies are fed into a softmax classifier for TFBS predictions.

Our proposed framework of CNN_TF is shown in **Fig 1**. The CNN_TF model consists of two CNNs: one is used for extracting low order dependencies from sequence features and the other is used for extracting higher order dependencies from histone modification features. Both the two CNNs contain three layers: the convolution layer, the rectification layer, and the pooling layer. Finally, the feature representations learned by the two CNNs are fed into a softmax classifier for predictions. Training for CNN_TF includes three sets of parameters: (1) filters F_S and thresholds b_S in the CNN for sequence features S, (2) filters F_C and thresholds b_C in the CNN for histone modification features C, and (3) the weights W for classification, where W_0 and W_1 are weight vectors for TFBS and non-TFBS, respectively. For a TFBS T, CNN_TF

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2019.2892124, IEEE/ACM Transactions on Computational Biology and Bioinformatics

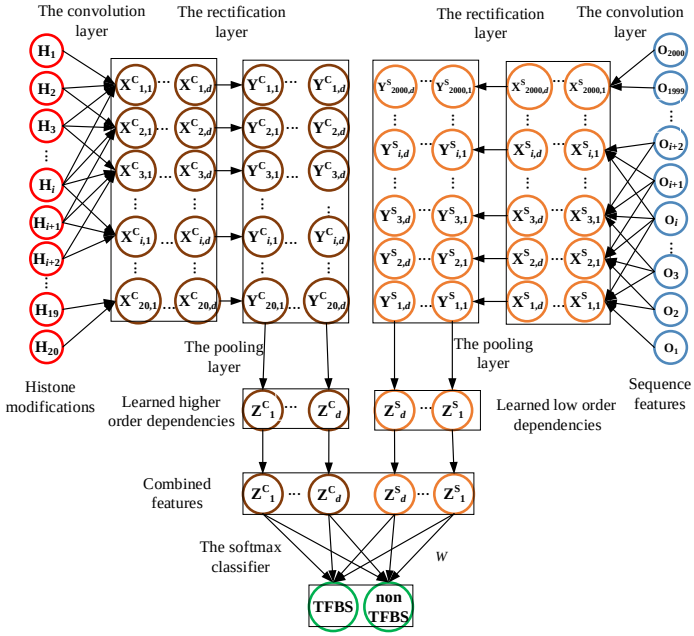


Fig. 1. The schematic graph of the architecture of CNN_TF.

provides a real-valued score $f(T)$ according to the following formula

$$f(T) = softmax_W(pool(rect_{b_S}(conv_{F_S}(S))) \bigoplus pool(rect_{b_C}(conv_{F_C}(C)))), \quad (4)$$

where $f(T)$ is defined by the softmax classifier through the concatenation operation $\bigoplus$ of two elements. The first element is the low order dependencies learned by the CNN from sequence features, where $conv_{F_S}()$, $rect_{b_S}$ and $pool()$ represent the three layers in the CNN for sequence features S. The second element is the high order dependencies learned by the CNN from histone modification features. Similarly $conv_{F_C}()$, $rect_{b_C}$ and $pool()$ denote the three layers in the CNN for histone modification features C. This real-valued softmax score is used for the prediction.

Details of the two CNNs will be explained by using the higher order dependency extraction as an instance. In the convolution layer, let us assume that there are d filters each has a length of m to convolve the raw input. Then, for a TFBS T with histone modification feature representation C, the convolution output is

$$\hat{X}^C = conv_{F_C}(C), \quad (5)$$

where $\hat{X}^C$ is an $(20 + m - 1) \times d$ matrix (20 denotes the length of the sequence of histone modification features). For the rectification layer, the input is $\hat{X}^C$ and the output $\hat{Y}^C = rect_{b_C}(\hat{X}^C)$ is get by the following formula

$$\hat{Y}^C_{i,k} = max(0, \hat{X}^C_{i,k} - b^C_k), \quad (6)$$

where b^C_k is the activation threshold for the filter k, learned in the training process. This layer is used to identify the important features by keeping only the scores larger than a specified threshold. The pooling layer takes the output matrix $\hat{Y}^C$ as input and outputs an vector $\vec{Z}^C$ with dimension of d. The element $\vec{Z}^C_k (1 \le k \le d)$ of vector $\vec{Z}^C$ is computed as

$$\vec{Z}^C_k = max(\hat{Y}^C_{1,k}, \cdots, \hat{Y}^C_{i,k}, \cdots, \hat{Y}^C_{(20+m-1),k}), \quad (7)$$

where $\hat{Y}^C_{i,k}$ is the element of the output matrix by the rectification layer.

Finally, the prediction for the TFBS T is completed by the following formula

$$f = softmax(W_{j,2\times d+1} + \sum_{k=1}^{d} W_{j,k}\vec{Z}^S_k + \sum_{k=1}^{d} W_{j,(d+k)}\vec{Z}^C_k), \quad (8)$$

where j denotes the prediction label ($j = 0$ and $j = 1$ denote the TFBS and the non-TFBS, respectively). $W_{0,*}$ is the weight vector for classifying input sequences as the TFBSs while $W_{1,*}$ is the weight vector for classifying input sequences as the non-TFBSs, $\vec{Z}^S$ represents the low order dependencies learned by the CNN from sequence features, $\vec{Z}^C$ represents the high order dependencies learned from the CNN from histone modification features, d denotes the dimension of these two representations. We trained CNN_TF using the following hyperparameters: both the two CNNs have 100 filters of length 10 and the dropout probability for both the two CNNs are 0.5.

III. Results

Four sets of evaluations are conducted here. The first experiment compares higher order dependencies with low order dependencies as well as their combined use on the 13 TFs in the mES cell. The second experiment compares CNN_TF with traditional classifiers which cannot extract dependencies by same features. The third experiment evaluates the influence of different cell-types on the performance of CNN_TF by the five TFs in the five cell-types of humans and the last experiment compares our proposed CNN_TF with state-of-the-art methods. Finally, based on the 13 TFs in the mES cell, we analyze the higher order dependencies learned by CNN_TF. In this section, definitions of metrics for our evaluation and parameter settings will be introduced first.

A. Evaluation Metrics and parameter settings

For TFBS predictions, since negative instances are far more than positive instances, we evaluate our proposed CNN_TF using the Area under the Precision-Recall curve (AUPR)[36] and the positive predictive value (PPV). The Precision-Recall curve plots the precision versus the recall of different thresholds on the importance score [37]. AUPR measures the similarity of the predictions to a known gold standard and is a more appropriate evaluation metric for extremely unbalanced datasets than AUC [37, 38]. The value of AUPR is between 0 and 1, indicating the lowest and highest performance, respectively. The positive predicative value (PPV) [17] is another useful evaluation metric for TFBS predictions, which can be calculated by following formula

$$PPV = TP/(TP + FP), \quad (9)$$

where TP denotes the number of true positives and FP denotes the number of false positives. The leave-one-chromosome-out cross-validation method is applied to evaluate

the performance of CNN_TF. In this validation method, one chromosome is used for test, one used for validation and the remaining chromosomes are used for training. The above test process is repeated for every chromosome and the performance is the average over all the chromosomes.

Two experiments are conducted to identify the optimal length for TFBSs and the optimal ratio of negative to positive instances in the training set by using CTCF in the mES cell as an instance. **Fig. 2(A)** shows the AUPR for TFBSs with different lengths when the ratio of negative to positive instances in the training set is set to 1. Results indicate that the length has very little effect on the performance of CNN_TF although TFBSs with length of 2000 achieve the highest AUPR. This is consistent with Won et al.'s work [17]. This is why we define TFBSs as DNA segments with 2000 nucleotides in our work. **Fig. 2(B)** shows the AUPR for training sets with different ratio of negative to positive instances. Results indicate that CNN_TF achieves the highest AUPR when the ratio of negative to positive instances in the training set is 3. So in this paper, the ratio of negative to positive instances in the training set is set to 3.

To analyze the performance on test sets with different ratio of negative to positive instances, we also conduct an experiment on CTCF in the mES cell. **Fig. 3** shows the AUPR of both low order and higher order dependencies as well as their combined use. As expected, AUPR declines gradually when the ratio increases. Due to the length limit of this paper, we measure the performance of CNN_TF in only two representative settings: (1) for the TFs in the mEs cell, the test set contains all the negative and positive instances; and (2) for the TFs in humans, the test set uses negative to positive ratio of 1.

B. Performances of low order and higher order dependencies

low order dependencies and higher order dependencies are learned by CNN from sequence features and histone modification features, respectively. To demonstrate the superiority of higher order dependencies over low order dependencies for TFBS predictions, we compare their predicting performance on the 13 TFs in the mES cell by the leave-one-chromosome-out cross-validation.

TABLE I shows the AUPRs of low order and higher order dependencies as well as their combined use on the 13 TFs in the mES cell. Among the 13 TFs in the mES cell, higher order dependencies outperforms low order dependencies significantly on 11 TFs while low order dependencies outperforms higher order dependencies on only 2 TFs. The AUPR of low order and higher order dependencies demonstrate the superiority of higher order dependencies clearly and there is no sign of overfitting. When these two dependency types are used in combination, it outperforms the two individual dependency types significantly on 12 TFs. This is a clear indication that low order dependencies and higher order dependencies are complimentary for TFBS predictions. The exception is Smad1 in which the combined use achieves lower performance than higher order dependencies.This is because the performance of the low order dependencies is too low compare to that of higher order dependencies. This further demonstrates the superior of higher order dependencies over low order dependencies. Note that two sets of p-values in this experiment are calculated by Wilcoxon rank sum test and all the p-values in the following text are calculated by Wilcoxon rank sum test. The p-values show that both performance improvements by higher order dependencies and the combined use are significant with p-value at no more than 7.17e-03.

TABLE I
AUPR OF LOW ORDER AND HIGH ORDER DEPENDENCIES ON THE 13 TFS IN MES CELL

TF	low order	higher order	combine	p-value[a]	p-value[b]
Zfx	0.511	0.654	**0.725**	3.25e-09	2.36e-4
CTCF	0.821	0.571	**0.894**	4.06e-22	2.74e-33
c-Myc	0.345	0.523	**0.535**	1.17e-08	1.01e-07
n-Myc	0.523	0.675	**0.702**	1.83e-08	9.79e-06
E2f1	0.363	0.802	**0.806**	5.54e-35	6.85e-03
Esrrb	0.683	0.572	**0.821**	1.86e-07	2.18e-19
Klf4	0.508	0.585	**0.696**	5.71e-03	2.64e-04
Tcfcp211	0.552	0.611	**0.748**	1.53e-04	1.46e-13
Nanog	0.214	0.250	**0.485**	1.10e-04	2.91e-26
Oct4	0.131	0.271	**0.338**	4.91e-12	2.40e-04
Smad1	0.012	**0.168**	0.167	1.25e-17	9.38e-01
Sox2	0.244	0.360	**0.526**	1.29e-10	9.16e-13
STAT3	0.102	0.205	**0.210**	1.01e-10	7.17e-03

[a] denotes the comparison between low order and higher order dependencies, [b] denotes the maximum p-value for the comparisons between the combine used and the two individual dependency types. The bold and underscore numbers denote the best and the second best performers, respectively.

C. Comparisons with typical bio-classifiers

The main advantage of our proposed CNN_TF is that it can extract both low order dependencies and higher order dependencies from sequence features and histone modification features, whereas typical classifiers including support vector machine (SVM) [39], random forest (RF) [20] and Multilayer perceptron (MLP) [40] can only use low order dependencies contained in histone modification features. To demonstrate that higher order dependencies are indeed useful for TFBS predictions, we compare CNN_TF with SVM, RF and MLP, all of which cannot capture higher order dependencies. SVM, RF and MLP require numeric features and cannot be fed with sequence data directly. Thus the input to these methods contains two integral parts: (1) normalized occurrence frequencies of nucleotides, dinucleotides and trinucleotides as well as (2) histone modification features. So the input features for instances can be represented as vectors with dimension of $(4 + 16 + 64 + 20 \times m)$, where m denote the number of used histone modification feature types.

The comparison among CNN_TF and the three typical classifiers is completed on the 13 TFs in the mES cell by the leave-one-chromosome-out cross-validation. The AUPRs of CNN_TF and the three traditional classifiers are shown in **TABLE II**. **TABLE II** shows that on 12 out of the 13 TFs, CNN_TF outperforms all the other classifiers by a larger margin with p-value of 8.34e-4 at least, to indicate that the improvements are very significant. More impressively, the improvements for CTCF, Esrrb and Nanog are more than 20%, the improvements for Zfx, Klf4, Tcfcp211 and Sox2 are more

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2019.2892124, IEEE/ACM Transactions on Computational Biology and Bioinformatics

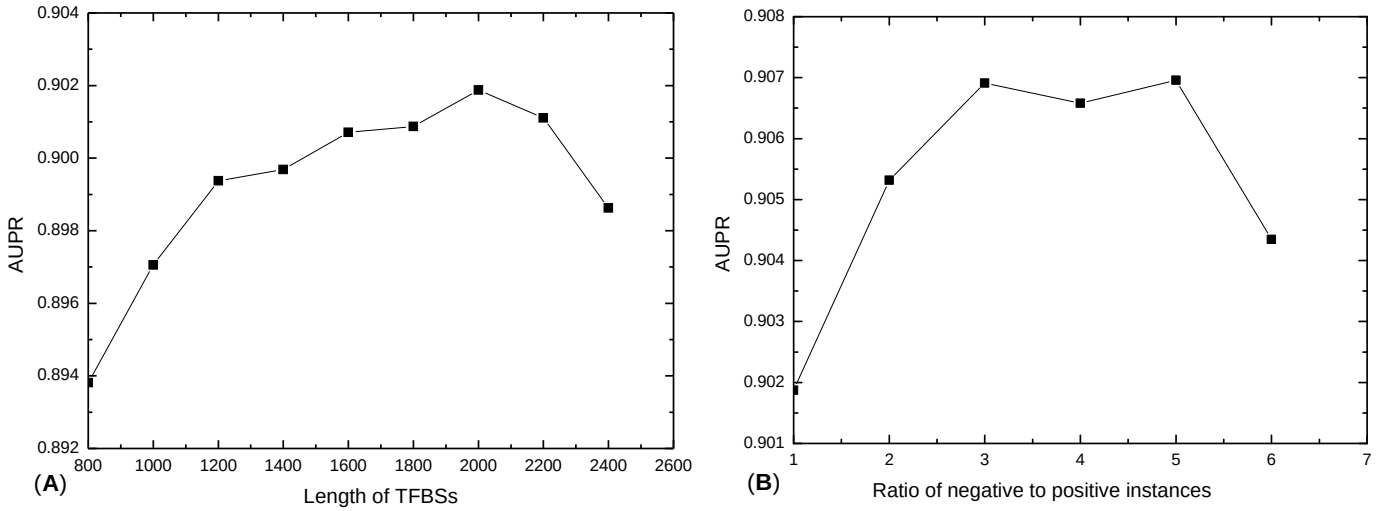


Fig. 2. (A) The AUPR for TFBSs with different length. (B) The AUPR for training sets with different ratio of negative to positive instances.

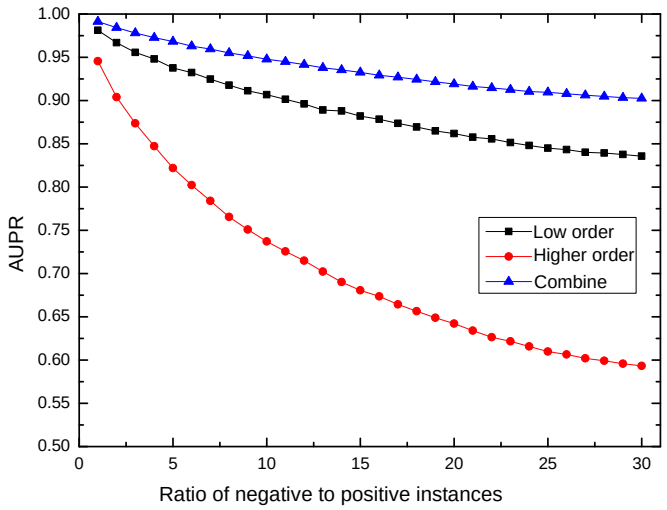


Fig. 3. The AUPR for test sets with different ratio of negative to positive instances.

TABLE II
AUPR OF CNN_TF AND THREE STATE-OF-THE-ART TRADITIONAL CLASSIFIERS ON THE 13 TFS IN THE MES CELL

TF	CNN_TF	SVM	RF	MLP	p-value[a]
Zfx	**0.725**	0.437	0.606	0.565	8.02e-09
CTCF	**0.894**	0.627	0.467	0.659	3.79e-16
c-Myc	**0.535**	0.323	0.453	0.425	8.34e-04
n-Myc	**0.702**	0.477	0.611	0.579	9.71e-07
E2f1	**0.806**	0.581	0.760	0.759	1.07e-04
Esrrb	**0.821**	0.523	0.419	0.539	1.27e-17
Klf4	**0.696**	0.465	0.492	0.534	8.72e-07
Tcfcp211	**0.748**	0.541	0.460	0.593	9.39e-12
Nanog	**0.485**	0.268	0.200	0.248	4.64e-24
Oct4	**0.338**	0.284	0.197	0.218	4.36e-08
Smad1	0.167	**0.196**	0.150	0.137	2.14e-01
Sox2	**0.526**	0.383	0.266	0.302	1.06e-11
STAT3	**0.210**	0.188	0.158	0.124	9.81e-05

[a] denotes the maximum p value of the comparisons between CNN_TF and the three state-of-the-art traditional classifiers. The bold and underscore numbers denote the best performers and the second best performers, respectively.

than 10%, the improvements for c-Myc, n-Myc and Oct4 are more than 5% and the improvments for the other 2 TFs are more than 2%. For Smad1, although SVM achieves higher AUPR than our method, the improvement is not significant as indicated by p-value of 2.14e-01. This evaluation validates that high order dependencies learned by CNN_TF indeed supply additional useful information for TFBS predictions.

D. Performances of CNN_TF on TFs in cell-types of humans

Several recent studies have reported that TF bindings are influenced by chromatin features such as DNA accessibilities, nucleosome occupancies, or the presence of some specific histone post-translational modifications. These chromatin features are different for different cell-types. So in this study, CNN_TF is applied to predict TFBSs for TFs in multiple different cell-types to analyze their influence on prediction performances. For each TF, five cell-types are considered: GM12878, H1-hESC, HeLa-S3, HepG2 and K562 in humans. We first evaluate the influence of different cell-types on the contributions of low order dependencies and higher order dependencies for TFBS predictions. The AUPRs of low order dependencies, higher order dependencies and their combined use are shown in **TABLE III**. Results show that for each TF, the performance of higher order dependencies for different cell-types are different. For GABP, higher order dependencies outperform low order dependencies significantly in all the five cell-types. When the two features are combined, the predicting performance is improved significantly in all the five cell-types. On the other hand, low order dependencies perform significantly better than higher order dependencies for REST in all the five cell-types. For REST too, the combined use still gains significant improvement in all the five cell-types. For the remaining three TFs, no single dependency type plays a dominant role. However, the performances for most cell-types are improved significantly when the two features are combined. This experiment clearly shows that low order dependencies and higher order dependencies have different contributions in the TFBS prediction for different cell-types. Furthermore, the two types of dependencies are complementary to each other and thus their combined use outperforms any single use irrespective of their dominance as a single dependency type

for different cell-types.

E. Comparison between CNN_TF and state-of-the-art methods on TFs in the mES cell

In this experiment, we compare our method with several state-of-the-art methods including Chromia [17], Cluster-Buster (CB) [41], MCAST [42], EEL [43] and Stubb [44] on the 13 TFs in the mES cell. Stubb has two versions: one is called Stubb-Single (SS) and the other is called Stubb-Multiple (SM). Chromia was proposed by Won at al. [19] based on a HMM model, in which both histone modification features and sequence features are used for learning feature representations. In Chromia, three HMM models including the promoter model, the enhancer model and the background model are trained and the log-odd score of the promoter model or the enhancer model to the background model is used for the prediction. Cluster-Buster [41] uses motifs documented in databases including JASPAR [45] and TRANSFAC [46] or predicted by de novo motif finding algorithms to search for TFBSs from test sequences. MCAST [42] uses a motif-based HMM model with several novel features to model TFBSs, for which a DNA database and a collection of known binding site motifs are used as inputs. In MCAST, motif-specific p-values are used to identify motif occurrences. EEL [43] uses motif conservation information and TFBS clusterings in the prediction model, which locates the enhancer elements according to a simplified biochemical and physical model of TF bindings [43]. In EEL, the binding score of a putative TFBS is calculated by aligning to the orthologous sequences. Stubb [44] uses a HMM framework to model enhancers by including motif conservation information and TFBS clusterings. In Stubb, the free energy calculated by Stubb is used for the prediction, where Stubb-Single uses correlations between binding sites to calculate the free energy while Stubb-Multiple incorporates phylogenetic comparisons among sequences from multiple species to calculate the free energy. For hyper parameters used in these methods, see Supplementary Note in our web server.

As both Stubb and EEL require pairwise alignments with other genomes and it is too time-consuming to evaluate their performance on the entire genome, only 20 chunks of genomic sequences (total513,846,568 bp) [19] that have pairwise alignments with the human genome were selected from the UCSC genome browser for test. The remaining genomic sequences are used for training. Note that the TFBSs for c-Myc and n-Myc have similar properties and Chromia combined them into a dataset labeled by Myc. So, we also incorporated them into a dataset. In the performance evaluation for all the 6 state-of-the-art methods and our CNN_TF method, only the top 600 predicted sites with larger prediction weights are used to estimate their performance. The PPV score of the top 600 predicted sites for each method is calculated and shown in **TABLE IV**, where the PPVs of the 6 state-of-the-art methods are referred from Won et al.'s work [17].

Results show that CNN_TF achieves obvious improvements for all the 12 TFs. For some TFs, the improvement achieved by CNN_FT is very promising. For example, the improvement for CTCF is over 60% PPV and the improvements for Esrrb, Tcfcp and Nanog are over or near 40% PPV. Note that some of the state-of-the-art methods cannot even provide any true TFBS for some TFs. For example, Stubb-Single and Stubb-Multiple cannot identify any true TFBS for Smad; Cluster-Buster cannot identify any true TFBS for E2f1, Nanog, Oct4, Smad1, Sox2 and STAT3. This comparison validates the usefulness of our proposed CNN_TF for TFBS predictions.

TABLE III
AUPR OF LOW ORDER DEPENDENCIES AND HIGH ORDER DEPENDENCIES ON TFS IN CELL-TYPES OF HUMANS

TF	CELL	low order	higher order	Combine	p-value[a]	p-value[b]
CTCF	GM12878	**0.954**	0.742	0.945	1.02e-02	2.46e-14
	H1-hESC	0.893	0.698	**0.933**	1.33e-14	2.50e-18
	HeLa-S3	0.895	0.729	**0.935**	1.00e-06	3.58e-07
	HepG2	0.932	0.781	**0.942**	2.86e-08	1.59e-12
	K562	0.892	0.750	**0.912**	8.73e-06	2.19e-09
GABP	GM12878	0.916	0.964	**0.984**	1.42e-06	5.61e-06
	H1-hESC	0.837	0.844	**0.877**	2.29e-01	1.81e-02
	HeLa-S3	0.963	0.970	**0.990**	5.60e-03	1.69e-02
	HepG2	0.939	0.954	**0.983**	2.08e-02	2.89e-04
	K562	0.890	0.954	**0.962**	1.00e-06	1.76e-02
JunD	GM12878	0.658	0.989	**0.990**	3.65e-10	7.78e-01
	H1-hESC	0.929	0.821	**0.961**	1.61e-04	8.76e-04
	HeLa-S3	0.954	0.969	**0.989**	8.04e-09	3.21e-02
	HepG2	**0.981**	0.730	0.972	3.08e-09	5.32e-02
	K562	0.879	0.860	**0.937**	5.12e-01	3.85e-03
REST	GM12878	0.822	0.791	**0.938**	1.10e-01	2.02e-06
	H1-hESC	0.853	0.731	**0.931**	6.11e-07	1.98e-10
	HeLa-S3	0.876	0.775	**0.946**	2.52e-03	2.49e-06
	HepG2	0.846	0.770	**0.947**	1.79e-04	1.52e-11
	K562	0.854	0.890	**0.947**	1.65e-03	6.93e-09
USF2	GM12878	0.947	0.898	**0.955**	1.91e-01	1.28e-02
	H1-hESC	0.880	0.831	**0.919**	5.17e-02	3.02e-04
	HeLa-S3	0.930	0.873	**0.975**	2.81e-04	1.16e-02
	HepG2	0.849	0.835	**0.930**	6.49e-01	2.79e-04
	K562	0.832	0.903	**0.945**	7.34e-03	6.22e-03

[a] denotes the comparison between low order dependencies and higher order dependencies, [b] denotes the maximum p-value of the comparisons between the combine use and the two individual dependency type. The bold and underscore numbers denote the best performers and the second best performers, respectively.

TABLE IV
PPV OF CNN_TF AND THREE STATE-OF-THE-ART METHODS ON THE 13 TFS IN THE MES CELL

TF	CNN-TF	Chromia	CB	MCAST	EEL	SS	SM
Zfx	**81.5%**	51.7%	5.6%	0.2%	24.8%	46.9%	26.0%
CTCF	**98.6%**	13.2%	51.3%	37.9%	44.0%	13.4%	3.9%
Myc	**82.8%**	57.8%	7.1%	0.4%	3.3%	20.2%	17.8%
E2f1	**98.1%**	85.3%	0.0%	1.3%	0.5%	12.0%	8.2%
Esrrb	**66.8%**	23.5%	9.7%	4.9%	16.2%	13.9%	5.1%
Klf4	**60.0%**	34.2%	5.7%	0.3%	12.5%	28.6%	9.5%
Tcfcp211	**77.3%**	33.8%	5.0%	11.5%	27.2%	12.7%	5.3%
Nanog	**47.3%**	7.8%	0.0%	0.4%	0.7%	1.4%	0.1%
Oct4	**25.0%**	15.0%	0.0%	2.8%	3.5%	0.5%	0.0%
Smad1	**10.6%**	1.0%	0.0%	0.4%	0.2%	0.0%	0.0%
Sox2	**35.8%**	4.2%	0.0%	2.4%	2.8%	0.2%	0.8%
STAT3	**17.1%**	1.0%	0.0%	0.2%	1.6%	2.9%	0.8%

CB denotes Cluster-Buster, SS denotes Stubb-Single and SM denotes Stubb-Multiple. The bold and underscore numbers denote the best performers and the second best performers, respectively.

F. Comparisons of CNN_TF with state-of-the-art methods on TFs in cell-types of humans

DNA shapes represent the 3D structures of DNA. Recently, Mathelier at al. [21] proposed a DNA shape based method for TFBS predictions in vivo. Four DNA shape features including the helix twist (HelT), the minor groove width (MGW), the propeller twist (ProT), and the Roll were used to represent putative TFBSs. These four DNA shape features were computed by a DNA shape method [47]. In Mathelier et al's work, four prediction models were developed: (1) one-hot+shape, which combines the one-hot encoding of nucleotides with DNA shape features; (2) PSSM+shape, which combines PSSM scores with DNA shape features; (3) TFFM_d+shape, which combines detailed TFFM scores and DNA shape features, and (4) TFFM_f+shape, which combines 1st-order TFFM scores and DNA shape features. The one-hot encoding, the PSSM scores and the TFFM scores [13] used in these DNA shape based models are representations of DNA sequence features. The difference between CNN_TF and these four models is that the four models only can extract low order dependencies by including DNA shape features whereas CNN_TF can extract both low order dependencies and higher order dependencies.

The evaluation is conducted on the five TFs in the five cell-types of humans by the leave-one-chromosome-out cross-validation. The results are listed in **TABLE V**. **TABLE V** shows that CNN_TF outperforms the four DNA shape based models significantly on all the 25 cell-type-TF pairs. A pair denotes the prediction task of a TF in a cell-type. The maximum improvement and the minimum improvement achieved by CNN_TF are 0.329 AUPR and 0.053 AUPR, respectively. The average improvement is 0.152 AUPR, which is a very large improvement. As the four shape feature based models can extract only low order dependencies by including DNA shape features while our proposed CNN_TF can extract both low order dependencies and higher order dependencies, the larger improvements achieved by CNN_TF are attributed the higher order dependencies learned by CNN_TF.

In addition to the DNA shape based method, deep learned methods including DeepSEA [23] and DanQ [24] also achieved state-of-the-art performances. In this section, we compare CNN_TF to DeepSEA [23] and DanQ [24] on the five TFs in the five cell-types of humans by the leave-one-chromosome-out cross-validation. As the hyper parameters of the three methods have been tuned by their authors, we use the same hyper parameters reported in the respective literature for these methods to make a fair comparison. DeepSEA contains three convolution layers and two max pooling layers in alternating order, followed by one fully connected layer and and a sigmoid output layer. The three convolution layers have 320, 480 and 960 kernels, respectively and the size of all the kernels is 8. Both the window size and the step size of the two max pooling layers are 4. The fully connected layer has 925 neurons. DanQ contains one convolution layer and one max pooling layer. The max pooling layer is followed by a bi-directional long short-term memory network (BLSTM), followed by a fully connected layer and a sigmoid output layer. The convolution layer contains 320 convolution kernels with size of 26. 13 is used as both the window size and the step size of the max pooling layer. The fully connected layer contains 925 neurons. For DanQ, Quang and Xie [24] proposed an alternative model, called DanQ-JASPAR, by initializing half of the kernels in the CNN with motifs from the JASPAR database [48]. The convolution layer in DanQ-JASPAR contains 1,024 kernels of size 30. Both the window size and step size of the max pooling are set to be 15. Detailed specifications of the architectures and hyper parameters used in these three methods are given in the Supplementary Note in our web server. The comparison among CNN_TF and these three methods is conducted on the five TFs in the five cell-types of humans as 7 types of histone modification features are available for these five cell-types. The tasks in DeepSEA, DanQ and DanQ-JASPAR contain 32 prediction tasks: the TFBS predictions of the five TFs in the five cell-types and the peak predictions of H3K4me2, H3K4me3, H4K20me1, H3K9ac, H3K27ac, H3K27me3 and H3K36me3. Finally, the TFBS predictions of the five TFs in the five cell-types are used for the comparative study.

The AUPRs of CNN_TF and the three state-of-the-art methods are listed in **TABLE VI**. Result shows that DanQ achieves higher AUPR than DanQ-JASPAR on 11 cell-type-TF pairs and achieves lower AUPR than DanQ-JASPAR on the remaining pairs. It indicates that DanQ and DanQ-JASPAR in practice have comparable performances. Result also shows that DanQ performs better than DeepSEA by 0.054 AUPR on average for the 25 cell-type-TF pairs, which is consistent with the work of Quang and Xie [24]. Note that CNN_TF performs better than DeepSEA by 0.238 AUPR on average for all the

TABLE V
AUPR OF THE FOUR DNA SHAPE BASED MODELS AND CNN_TF ON THE TFS IN CELL-TYPES OF HUMANS

TF	CELL	One-hot	PSSM	TFFM_d	TFFM_f	CNN-TF	*p*-value
CTCF	GM12878	0.777	0.775	0.758	0.762	**0.945**	4.98e-03
	H1-hESC	0.792	0.787	0.765	0.769	**0.933**	8.57e-03
	HeLa-S3	0.763	0.759	0.746	0.748	**0.935**	2.29e-02
	HepG2	0.788	0.785	0.769	0.774	**0.942**	8.12e-03
	K562	0.780	0.778	0.761	0.764	**0.912**	4.44e-02
GABP	GM12878	0.833	0.830	0.843	0.841	**0.984**	9.16e-03
	H1-hESC	0.824	0.821	0.815	0.815	**0.877**	1.96e-03
	HeLa-S3	0.804	0.805	0.794	0.791	**0.990**	7.67e-03
	HepG2	0.856	0.852	0.844	0.850	**0.983**	9.11e-03
	K562	0.833	0.829	0.822	0.826	**0.962**	2.32e-02
JunD	GM12878	0.711	0.711	0.683	0.692	**0.990**	1.83e-14
	H1-hESC	0.799	0.797	0.782	0.786	**0.961**	7.83e-10
	HeLa-S3	0.837	0.832	0.808	0.813	**0.989**	3.01e-15
	HepG2	0.815	0.810	0.790	0.794	**0.972**	4.09e-18
	K562	0.822	0.818	0.799	0.804	**0.937**	1.61e-09
REST	GM12878	0.785	0.782	0.764	0.774	**0.938**	2.07e-02
	H1-hESC	0.787	0.786	0.762	0.769	**0.931**	2.07e-02
	HeLa-S3	0.602	0.607	0.617	0.592	**0.946**	1.75e-19
	HepG2	0.791	0.791	0.777	0.777	**0.947**	2.23e-02
	K562	0.789	0.785	0.768	0.772	**0.947**	1.55e-02
USF2	GM12878	0.827	0.825	0.807	0.810	**0.955**	8.58e-10
	H1-hESC	0.839	0.835	0.818	0.822	**0.919**	4.65e-05
	HeLa-S3	0.819	0.815	0.800	0.804	**0.975**	5.76e-09
	HepG2	0.840	0.836	0.815	0.820	**0.930**	1.68e-05
	K562	0.822	0.821	0.800	0.802	**0.945**	1.00e-06

The bold and underscore numbers denote the best performers and the second best performers, respectively.

TABLE VI
AUPR OF CNN_TF AND FOUR STATE-OF-THE-ART METHODS ON THE TFS IN CELL-TYPES OF HUMANS

TF	CELL	DanQ	DanQ-JASPAR	DeepSEA	CNN-TF
CTCF	GM12878	0.773	0.737	0.684	**0.945**
	H1-hESC	0.802	0.762	0.692	**0.933**
	HeLa-S3	0.716	0.691	0.649	**0.935**
	HepG2	0.805	0.763	0.706	**0.942**
	K562	0.723	0.689	0.626	**0.912**
GABP	GM12878	0.776	0.836	0.746	**0.984**
	H1-hESC	0.748	0.762	0.721	**0.877**
	HeLa-S3	0.631	0.689	0.622	**0.990**
	HepG2	0.758	0.813	0.745	**0.983**
	K562	0.743	0.760	0.718	**0.962**
JunD	GM12878	0.698	0.721	0.681	**0.990**
	H1-hESC	0.674	0.670	0.625	**0.961**
	HeLa-S3	0.766	0.771	0.717	**0.989**
	HepG2	0.836	0.850	0.784	**0.972**
	K562	0.639	0.638	0.576	**0.937**
REST	GM12878	0.595	0.579	0.576	**0.938**
	H1-hESC	0.599	0.588	0.554	**0.931**
	HeLa-S3	0.581	0.582	0.542	**0.946**
	HepG2	0.612	0.584	0.593	**0.947**
	K562	0.628	0.631	0.613	**0.947**
USF2	GM12878	0.658	0.685	0.602	**0.955**
	H1-hESC	0.705	0.731	0.632	**0.919**
	HeLa-S3	0.629	0.640	0.546	**0.975**
	HepG2	0.687	0.755	0.572	**0.930**
	K562	0.658	0.717	0.565	**0.945**

The bold and underscore numbers denote the best performers and the second best performers, respectively.

pairs. As DeepSEA [23] can extract only low order dependencies while both DanQ and our proposed CNN_TF can extract both low order dependencies and higher order dependencies, the large improvements achieved by DanQ and CNN_TF over DeepSEA are contributed by higher order dependencies. When comparing CNN_TF with DanQ, result shows that CNN_TF performs better than DanQ for all the 25 cell-type-TF pairs. The least and the most improvement achieved by CNN_TF is 0.188 AUPR and 0.429 AUPR, respectively. The average improvement on the 25 cell-type-TF pairs is 0.313 AUPR, which is quite significant. Although DanQ [24] can extract higher order dependencies by incorporating RNN with CNN, our proposed CNN_TF performs better than DanQ by at least 0.188 AUPR for the 25 pairs. It indicates that the higher order dependencies learned by CNN_TF using histone modification features are more useful for predictions than that learned by DanQ which only made use of sequence features available at the time.

IV. DISCUSSION

Distinct histone modification features have been observed at different genomic loci. Won et al. [17] investigated the eight histone modification features for the TFBSs of the 13 TFs in the mES cell. They found that H3K4m1, H3K4m2 and H3K4m3 show strong signals around the TFBSs for all the 13 TFs while H3K27m3 shows much weaker signals around the TFBSs. More specifically, H3K4me1 and H3K4me2 present bimodal profiles for the TFBSs of all the 13 TFs; H3K4me3 shows strong peaks for the TFBSs of E2F1, c-Myc, n-Myc and Zfx, intermediate peaks for the TFBSs of Esrrb, Klf4, STAT3 and Tcfcp211, and weak signals for the TFBSs of CTCF, Nanog, Oct4, Smad1 and Sox2; H3K36me3 shows relatively strong signals for the TFBSs of E2f1, c-Myc, n-Myc and Zfx; and H3K9me3, H3K20me3 and H3K27me3 show low signals for the TFBSs of all the 13 TFs.

To validate the higher order dependencies learned by CNN_TF, we analyze the extracted higher order dependencies of a TF by using a summed filter of the TF, which is calculated by summing the d learned filters from histone modification features of the TF according to the following formula:

$$F = \sum_{k=1}^{d} W_{0,(d+k)} F_C^k, \tag{10}$$

where $W_{0,*}$ is the weight vector in the softmax classifier of CNN_TF, which denotes the contributions of individual filters for classifying inputs into TFBSs. F_C denotes the learned filters from histone modification features. For more details about W_0, please refer to Formula (8).

Due to the length limit of this paper, we only show the summed filters for c-Myc and Oct4. The summed filters for other TFs are listed in Figure S1 to S11 in the Additional file 1, which can be accessed from our website. **Fig. 4** and **Fig. 5** show the summed filters for Oct4 and c-Myc, respectively. The x-axis denotes the 10 positions in the summed filter and the y-axis denotes the weight for each position. **Fig. 4** shows that H3K4me1 and H3K4me2 indeed present a bimodal profile around the TFBSs of c-Myc and H3K4me3 shows strong signals for the TFBSs. In addition, H3K9me3, H3K20me3 and H3K27me3 show low signal for the TFBSs. This indicates that the learned higher order dependencies are consistent with the dependencies analyzed from ChIP-seq signals by a previous study [17]. **Fig. 5** also shows that H3K4me1 and H3K4me2 present bimodal profile around the TFBSs. H3K4me3 and the three repressive histone modification features including H3K9me3, H3K20me3 and H3K27me3 show weak signals for the TFBSs, which is also consistent with the conclusions of the previous study [17]. These results indicate that CNN_TF can indeed capture useful higher order dependencies for the prediction.

V. CONCLUSION

This paper presents the first study on TFBS predictions by using dependency information among histone modification features. Our proposed CNN_TF method captures low order dependencies as well as higher order dependencies by applying convolutional neural network to sequence features and histob ne modification features, respectively. Evaluations on both the 13 TFs in the mES cell and the 5 TFs in 5 different cell-types of humans show that higher order dependencies outperform low order dependencies significantly and the combine use performs better than individual dependency types significantly. This indicates that higher order dependencies are indeed more useful than low order dependencies for TFBS predictions. Our experiments also show that low order dependencies and higher order dependencies are complementary to each other in the prediction. Comparisons to state-of-the-art methods on both the 13 TFs in the mES cell the 5 TFs in 5 cell-types of humans

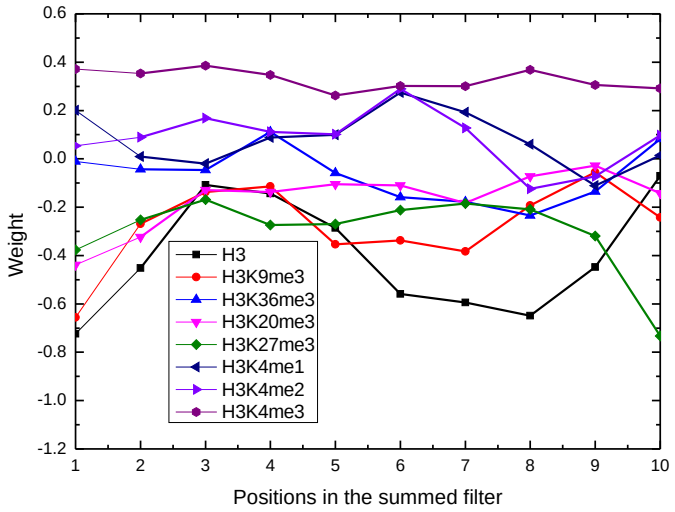


Fig. 4. The learned fragment dependency for c-Myc.

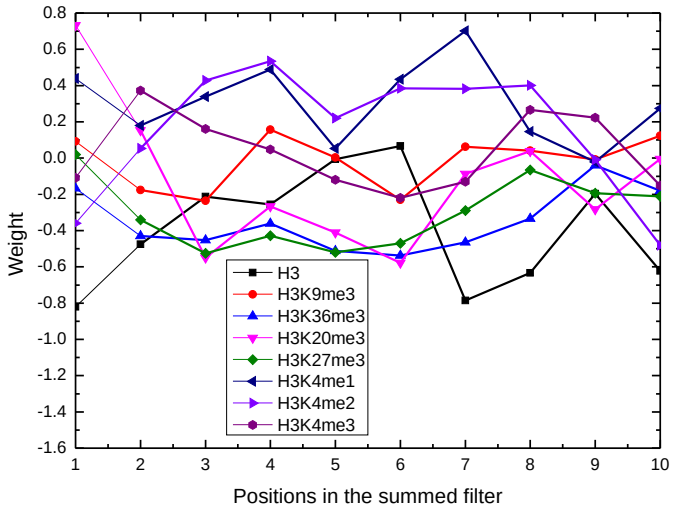


Fig. 5. The learned fragment dependency for Oct4.

show that CNN_TF outperforms the state-of-the-art methods with large improvements on all the TFs. Detailed examination of the higher order dependencies extracted by CNN_TF for all the 13 TFs in the mES cell shows that the learned higher order dependencies are consistent with the dependencies analyzed from ChIP-seq signals by a previous study. Our work on TFBS predictions indicates that the positions in the TFBSs for each TF indeed have higher order dependencies between each other. The positions in the TFBSs of TFs do not exist independently when interact with DNA. Our work is a further prove that high order dependencies do exist. As TFBSs are important integral components for gene transcriptions and translations, such as the TFBSs in promoters and enhancers, the extraction and analysis of high order dependencies contained in TFBSs can lead us to a deeper understanding of gene expression regulation and fundamental cellular processes of humans.

One direction of our future works is to investigate how to apply our proposed CNN_TF in cross-cell-type TFBS predictions. The second direction is to investigate how to use the Transformer model to extract dependencies between positions with even longer distances.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China U1636103, 61632011, Shenzhen Foundational Research Funding 20170307150024907, Key Technologies Research and Development Program of Shenzhen JSGG20170817140856618, HK Polytechnic Universitys graduate student grant: PolyU-RUDD.

COMPETING INTEREST

The authors declare that they have no conflicting interests.

REFERENCES

[1] F. Jacob and J. Monod, "Genetic regulatory mechanisms in the synthesis of proteins," *Journal of molecular biology*, vol. 3, no. 3, pp. 318–356, 1961.

[2] I. Dror, R. Rohs, and Y. Mandel-Gutfreund, "How motif environment influences transcription factor search dynamics: Finding a needle in a haystack," *BioEssays*, vol. 38, no. 7, pp. 605–612, 2016.

[3] V. R. Iyer, C. E. Horak, C. S. Scafe, D. Botstein, M. Snyder, and P. O. Brown, "Genomic binding sites of the yeast cell-cycle transcription factors SBF and mbf," *Nature*, vol. 409, no. 6819, pp. 533–538, 2001.

[4] M. L. Bulyk, "Computational prediction of transcription-factor binding site locations," *Genome biology*, vol. 5, no. 1, p. 201, 2003.

[5] G. D. Stormo, "DNA binding sites: representation and discovery," *Bioinformatics*, vol. 16, no. 1, pp. 16–23, 2000.

[6] B. Lenhard, A. Sandelin, L. Mendoza, P. Engström, N. Jareborg, and W. W. Wasserman, "Identification of conserved regulatory elements by comparative genome analysis," *Journal of biology*, vol. 2, no. 2, p. 13, 2003.

[7] D. T. Holloway, M. Kon, and C. DeLisi, "Integrating genomic data to predict transcription factor binding," *Genome informatics*, vol. 16, no. 1, pp. 83–94, 2005.

[8] N. M. Luscombe, R. A. Laskowski, and J. M. Thornton, "Amino acid-base interactions: a three-dimensional analysis of protein-DNA interactions at an atomic level," *Nucleic Acids Research*, vol. 29, no. 13, pp. 2860–2874, 2001.

[9] T. K. Man and G. D. Stormo, "Non-independence of mnt repressor-operator interaction determined by a new quantitative multiple fluorescence relative affinity (QuMFRA) assay," *Nucleic Acids Research*, vol. 29, no. 12, pp. 2471–2478, 2001.

[10] A. Tomovic and E. J. Oakeley, "Position dependencies in transcription factor binding sites," *Bioinformatics*, vol. 23, no. 8, pp. 933–941, 2007.

[11] F. Zare-Mirakabad, H. Ahrabian, M. Sadeghi, A. Nowzari-Dalini, and B. Goliaei, "New scoring schema for finding motifs in dna sequences," *Bmc Bioinformatics*, vol. 10, no. 1, pp. 1–21, 2009.

[12] R. Siddharthan, "Dinucleotide weight matrices for predicting transcription factor binding sites: generalizing the position weight matrix," *PloS one*, vol. 5, no. 3, p. e9722, 2010.

[13] A. Mathelier and W. W. Wasserman, "The next generation of transcription factor binding site prediction," *PLoS computational biology*, vol. 9, no. 9, p. e1003214, 2013.

[14] V. D. Marinescu, I. S. Kohane, and A. Riva, "MAPPER: a search engine for the computational identification of putative transcription factor binding sites in multiple genomes," *BMC bioinformatics*, vol. 6, no. 1, p. 79, 2005.

[15] Z. T. Y. Tsai, S. H. Shiu, and H. K. Tsai, "Contribution of sequence motif, chromatin state, and DNA structure features to predictive models of transcription factor binding in yeast," *PLoS computational biology*, vol. 11, no. 8, p. e1004418, 2015.

[16] S. Kumar and P. Bucher, "Predicting transcription factor site occupancy using DNA sequence intrinsic and cell-type specific chromatin features," *BMC bioinformatics*, vol. 17, no. 1, p. S4, 2016.

[17] K. J. Won, B. Ren, and W. Wang, "Genome-wide prediction of transcription factor binding sites using an integrated model," *Genome biology*, vol. 11, no. 1, p. R7, 2010.

[18] M. Talebzadeh and F. Zare-Mirakabad, "Transcription factor binding sites prediction based on modified nucleosomes," *PloS one*, vol. 9, no. 2, p. e89226, 2014.

[19] M. F. Berger, A. A. Philippakis, A. M. Qureshi, F. S. He, P. W. Estep, and M. L. Bulyk, "Compact, universal DNA microarrays to comprehensively determine transcription-factor binding site specificities," *Nature biotechnology*, vol. 24, no. 11, pp. 1429–1435, 2006.

[20] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[21] A. Mathelier, B. Xin, T. P. Chiu, L. Yang, R. Rohs, and W. W. Wasserman, "DNA shape features improve transcription factor binding site predictions in vivo," *Cell systems*, vol. 3, no. 3, pp. 278–286, 2016.

[22] B. Alipanahi, A. Delong, M. T. Weirauch, and B. J. Frey, "Predicting the sequence specificities of dna- and rna-binding proteins by deep learning," *Nature Biotechnology*, vol. 33, no. 8, p. 831, 2015.

[23] J. Zhou and O. G. Troyanskaya, "Predicting effects of noncoding variants with deep learningbased sequence model," *Nature Methods*, vol. 12, no. 10, pp. 931–934, 2015.

[24] D. Quang and X. Xie, "DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences," *Nucleic Acids Research*, vol. 44, no. 11, pp. e107–e107, 2016.

[25] H. Zeng, M. D. Edwards, G. Liu, and D. K. Gifford, "Convolutional neural network architectures for predicting dna-protein binding," *Bioinformatics*, vol. 32, no. 12, pp. i121–i127, 2016.

[26] J. Zhou, Q. Lu, R. Xu, L. Gui, and H. Wang, "CNNsite: Prediction of dna-binding residues in proteins using convolutional neural network with sequence features," in *Proceeding of the 2016 IEEE International Conference on Bioinformatics and Biomedicine*. IEEE, 2016, pp. 78–85.

[27] J. Zhou, Q. Lu, R. Xu, and L. Gui, "EL_LSTM: Prediction of DNA-binding residue from protein sequence by combining long short-term memory and ensemble learning," *IEEE/ACM Transactions on Computational Biology & Bioinformatics*, vol. PP, no. 99, pp. 1–12, 2018.

[28] R. Xu, J. Zhou, H. Wang, Y. He, X. Wang, and B. Liu, "Identifying dna-binding proteins by combining support vector machine and PSSM distance transformation," *BMC systems biology*, vol. 9, no. 1, p. S10, 2015.

[29] J. Zhou, R. Xu, Y. He, Q. Lu, H. Wang, and B. Kong, "Pdnasite: Identification of DNA-binding site from protein sequence by incorporating spatial and sequence context," *Scientific Reports*, vol. 6, p. 27653, 2016.

[30] N. D. Heintzman, R. K. Stuart, G. Hon, Y. Fu, C. W. Ching, R. D. Hawkins, L. O. Barrera, S. Van Calcar, C. Qu, K. A. Ching *et al.*, "Distinct and predictive chromatin signatures of transcriptional promoters and enhancers in the human genome," *Nature genetics*, vol. 39, no. 3, pp. 311–318, 2007.

[31] A. Barski, S. Cuddapah, K. Cui, T. Y. Roh, D. E. Schones, Z. Wang, G. Wei, I. Chepelev, and K. Zhao, "High-resolution profiling of histone methylations in the human genome," *Cell*, vol. 129, no. 4, pp. 823–837, 2007.

[32] X. Chen, H. Xu, P. Yuan, F. Fang, M. Huss, V. B. Vega, E. Wong, Y. L. Orlov, W. Zhang, J. Jiang *et al.*, "Integration of external signaling pathways with the core transcriptional network in embryonic stem cells," *Cell*, vol. 133, no. 6, pp. 1106–1117, 2008.

[33] T. Barrett, S. E. Wilhite, P. Ledoux, C. Evangelista, I. F. Kim, M. Tomashevsky, K. A. Marshall, K. H. Phillippy, P. M. Sherman, and M. Holko, "NCBI GEO: archive for functional genomics data setsupdate," *Nucleic Acids Research*, vol. 39, no. Database issue, pp. 1005–1010, 2011.

[34] T. S. Mikkelsen, M. Ku, D. B. Jaffe, B. Issac, E. Lieberman, G. Giannoukos, P. Alvarez, W. Brockman, T.-K. Kim, R. P. Koche *et al.*, "Genome-wide maps of chromatin state in pluripotent and lineage-committed cells," *Nature*, vol. 448, no. 7153, pp. 553–560, 2007.

[35] A. Meissner, T. S. Mikkelsen, H. Gu, M. Wernig, J. Hanna, A. Sivachenko, X. Zhang, B. E. Bernstein, C. Nusbaum, D. B. Jaffe *et al.*, "Genome-scale DNA methylation maps of pluripotent and differentiated cells," *Nature*, vol. 454, no. 7205, pp. 766–770, 2008.

[36] J. A. Swets, "Measuring the accuracy of diagnostic systems," *Science*, vol. 240, no. 4857, pp. 1285–1293, 1988.

[37] V. A. Huynhthu, A. Irrthum, L. Wehenkel, and P. Geurts, "Inferring regulatory networks from expression data using tree-based methods." *Plos One*, vol. 5, no. 9, pp. 4439–4451, 2010.

[38] A.-C. Haury, F. Mordelet, P. Vera-Licona, and J.-P. Vert, "TIGRESS: Trustful inference of gene regulation using stability selection," *BMC Systems Biology*, vol. 6, p. 145, 2012.

[39] V. N. Vapnik, "The nature of statistical learning theory," *IEEE Transactions on Neural Networks*, vol. 8, no. 6, pp. 1564–1564, 1997.

[40] C. M. Bishop, "Neural networks for pattern recognition," *Agricultural Engineering International the Cigr Journal of Scientific Research & Development Manuscript Pm*, vol. 12, no. 5, pp. 1235 – 1242, 2001.

[41] M. C. Frith, M. C. Li, and Z. Weng, "Cluster-Buster: Finding dense clusters of motifs in dna sequences." *Nucleic Acids Research*, vol. 31, no. 13, pp. 3666–3668, 2003.

[42] T. L. Bailey and W. S. Noble, "Searching for statistically significant regulatory modules," *Bioinformatics*, vol. 19 Suppl 2, no. suppl_2, p. ii16, 2003.

[43] K. Palin, J. Taipale, and E. Ukkonen, "Locating potential enhancer elements by comparative genomics using the eel software." *Nature Protocols*, vol. 1, no. 1, pp. 368–74, 2006.

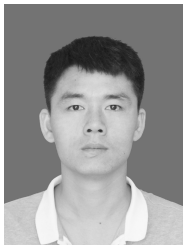
[44] S. Sinha, Y. Liang, and E. Siggia, "Stubb: a program for discovery and analysis of cis-regulatory modules," *Nucleic Acids Research*, vol. 34, no. Web Server issue, pp. 555–9, 2006.

[45] J. C. Bryne, E. Valen, M. H. Tang, T. Marstrand, O. Winther, P. I. Da, A. Krogh, B. Lenhard, and A. Sandelin, "JASPAR, the open access database of transcription factor-binding profiles: new content and tools in the 2008 update." *Nucleic Acids Research*, vol. 36, no. Database issue, pp. 102–106, 2008.

[46] V. Matys, O. V. Kel-Margoulis, E. Fricke, I. Liebich, S. Land, A. Barre-Dirrie, I. Reuter, D. Chekmenev, M. Krull, K. Hornischer *et al.*, "TRANSFAC® and its module TRANSCompel®: transcriptional gene regulation in eukaryotes," *Nucleic Acids Research*, vol. 34, no. Database issue, pp. D108–D110, 2006.

[47] T. Zhou, N. Shen, L. Yang, N. Abe, J. Horton, R. S. Mann, H. J. Bussemaker, R. Gordân, and R. Rohs, "Quantitative modeling of transcription factor binding specificities using DNA shape," *Proceedings of the National Academy of Sciences*, vol. 112, no. 15, pp. 4654–4659, 2015.

[48] A. Mathelier, O. Fornes, D. J. Arenillas, C. Y. Chen, G. Denay, J. Lee, W. Shi, C. Shyr, G. Tan, and R. Worsley-Hunt, "JASPAR 2016: a major expansion and update of the open-access database of transcription factor binding profiles," *Nucleic Acids Research*, vol. 44, no. Database issue, pp. D110–D115, 2016.



Jiyun Zhou obtained his B.Eng. degree from Northeast Forestry University, China, and M.Eng. degree from Harbin Institute of Technology, China, respectively. He is now a Ph.D. candidate in School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen) and Department of Computing, Hong Kong Polytechnic University. His main research interests are bioinformatics, natural language processing and machine learning

Qin Lu obtained her B.Eng. degree from Beijing Normal University, and M.Sc and Ph.D. degree from University of Illinois at Urbana-Champaign, respectively. She is now a Full Professor in Department of Computing, The Hong Kong Polytechnic University, Hong Kong. Her research interests are computational linguistics, ontology, text mining, knowledge discovery and bioinformatics.

Ruifeng Xu obtained his B.Eng. degree from Harbin Institute of Technology, China, and M.Phil. and Ph.D. degree from The Kong Polytechnic University, respectively. He is now a Full Professor and Ph.D. Supervisor in School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), China. His research interests are bioinformatics, natural language processing, emotion computing and text mining.

Lin Gui obtained his B.S. degree from Nankai University, China, and M.Eng. degree in from Harbin Institute of Technology, China, respectively. He is now a Ph.D. candidate in School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen). His research interests are natural language processing, machine learning and emotion computing.

Hongpeng Wang obtained his B.Eng., M. Eng. and Ph.D. degree from Harbin Institute of Technology, China. He is now a Full Professor and Ph.D. Supervisor in School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen). His research interests are intelligent robot, computer vision, artificial intelligence and bioinformatics.