Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Original Paper

Genome analysis

MTTFsite: Cross-cell-type TF Binding Site Prediction by using Multi-task Learning

Jiyun Zhou[1,2], Qin Lu[2], Lin Gui[3], Ruifeng Xu[1,*], Yunfei Long[2]and Hongpeng Wang[1]

[1]Schoolof Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, 518055, China
[2]Department of Computing, the Hong Kong Polytechnic University, Hung Hom, 999077, Hong Kong
[3]Department of Computer Science, University of Warwick, Coventry, cv4 4al, UK.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: The prediction of transcription factor binding sites (TFBSs) is crucial for gene expression analysis. Supervised learning approaches for TFBS predictions require large amounts of labeled data. However, many TFs of certain cell-types either do not have sufficient labeled data or do not have any labeled data.
Results: In this paper, a multi-task learning framework (called MTTFsite) is proposed to address the lack of labeled data problem by leveraging on labeled data available in cross-cell-types. The proposed MTTFsite contains a shared CNN to learn common features for all cell-types and a private CNN for each cell-type to learn private features. The common features are aimed to help predicting TFBSs for all cell-types especially those cell-types that lack labeled data. MTTFsite is evaluated on 241 cell-type TF pairs and compared to a baseline method without using any multi-task learning model and a fully-shared multi-task model which uses only a shared CNN and do not use private CNNs. For cell-types with insufficient labeled data, results show that MTTFsite performs better than the baseline method and the fully-shared model on more than 89% pairs. For cell-types without any labeled data, MTTFsite outperforms the baseline method and the fully-shared model by more than 80% and 93% pairs, respectively. A novel gene expression prediction method (called TFChrome) using both MTTFsite and histone modification features is also presented. Results show that TFBSs predicted by MTTFsite alone can achieve good performance. When MTTFsite is combined with histone modification features, a significant 5.7% performance improvement is obtained.
Availability: The resource and executable code are freely available at http://hlt.hitsz.edu.cn/MTTFsite/ and http://www.hitsz-hlt.com:8080/MTTFsite/.
Contact: xuruifeng@hit.edu.cn
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Transcription factor (TF) binding sites (TFBSs) are important for understanding transcriptional regulatory networks and fundamental cellular processes, such as growth controls, cell-cycle progressions and developments, as well as differentiated cellular functions (Wasserman and Sandelin, 2004; Dror *et al.*, 2016; Zambelli *et al.*, 2012). TFBSs are short and often degenerate sequence motifs (Bulyk, 2003), which makes them computationally difficult to predict at genomic scale. TFBSs can be represented by consensus sequences and position weight matrices (PWMs) (Stormo, 2000, 2013). The consensus sequence representation provides a convenient way for visual interpretation of TFBSs. But, nucleotide variations at each position make the consensus sequence representation unsuited to represent TFBSs (Lenhard *et al.*, 2003; Holloway *et al.*, 2005). To overcome this problem, the PWM representation was proposed to represent TFBSs (Stormo, 2000, 2013). PWMs are derived from a set of aligned functionally related sequences and assume that the positions within each TFBS are independent of each other. However, some studies have shown that position dependencies do exist in TFBSs, such as crystal structure analyses (Luscombe *et al.*, 2001), biochemical studies (Man and Stormo, 2001; Bulyk *et al.*, 2002; Berger *et al.*, 2006), and statistical analyses of large collections of TFBSs (Barash *et al.*, 2003; Tomovic and Oakeley, 2007; Zhou and Liu, 2004). In order to integrate position dependencies in predictions, a new approach, called dinucleotide weight matrix (DWM), was proposed recently (Siddharthan, 2010). DWM extends PWM by taking into account dependencies between any two positions (Siddharthan, 2010). TFFM proposed by Mathelier and Wasserman (Mathelier and Wasserman, 2013a) further captures position dependencies for predictions. In TFFM, state transition probabilities in a Hidden Markov Model (HMM) (Marinescu *et al.*, 2005) were used to model position dependencies. Although the above four representation methods can represent TFBSs, they capture only sequence features.

Recent approaches attempted to use histones modification features to improve the accuracy of TFBS predictions (Tsai *et al.*, 2015; Kumar and Bucher, 2016; Won *et al.*, 2010). Histone modification features refer to the post-translational modification levels of various histones in chromatin structures, which are closely related to the formation of TFBSs. Won at al. (Won *et al.*, 2010) proposed a HMM based method called Chromia by combined use of histone modification features and sequence features. Tsai et al. (Tsai *et al.*, 2015) examined the contributions of sequence features, histone modification features, and structure features in TFBS predictions (Breiman, 2001). They conclude that all these three feature types were significant in TFBS predictions.

Recent studies suggested that DNA shape features are another important type of features for TFBS predictions (Mathelier *et al.*, 2016a). Mathelier (Mathelier *et al.*, 2016a) proposed a method by using DNA shape features predicted by DNAshape (Zhou *et al.*, 2013) and achieved a very good prediction performance. Andrabi et al. (Andrabi *et al.*, 2017) proposed DynaSeq to predict molecular dynamics-derived ensembles of a more exhaustive set of DNA shape features and than used them to predict TFBSs. In addition to these DNA shape based methods, several deep learning methods were proposed for TFBS predictions. DeepBind (Alipanahi *et al.*, 2015), DeepSEA (Zhou and Troyanskaya, 2015) and DanQ (Quang and Xie, 2016) are three representative methods. DeepBind, proposed by Alipanahi, applies Convolutional Neural Network (CNN) to DNA sequence features. DeepSEA, proposed by Zhou and Troyanskaya, combines CNN and a multi-task learning method to learn representations. DanQ, an improved model of DeepSEA proposed by Quang and Xie, combines the use of CNN and Recurrent neural network (RNN). All these three deep learning based methods achieved very good predicting performance and are considered the state-of-the art works.

When there exists large amount of labeled data, supervised computational methods can achieve very good performance. However, TFBSs for most TFs can only be identified by ChIP-Seq (Iyer *et al.*, 2001; Harbison *et al.*, 2004; Kim *et al.*, 2005) or ChIP-chip (Ren *et al.*, 2000), which are experimental techniques and are very labor-intensive and costly to run. TFs of many cell-types do not have sufficient labeled data and some do not have any labeled data. It remains quite challenging to train predictors for TFs of cell-types that lack labeled data. Nevertheless, several studies (Tsai *et al.*, 2015; Kumar and Bucher, 2016; Won *et al.*, 2010) have shown that TFBSs of a TF in different cell-types have some common histone modification features. A TF may also have common binding motifs in different cell-types (Matys *et al.*, 2006; Bryne *et al.*, 2007). So computational methods can leverage on the labeled data available in other cell-types to predict TFBSs for cell-types lacking labeled data. In this paper, we propose a multi-task learning framework, called **MTTFsite**, for TFBS predictions. MTTFsite contains a shared CNN to learn common features for all cell-types and a private CNN for each cell-type to learn private features. When the target cell-type has labeled data, its private features and the common features are combined to predict TFBSs. Thus, for a target cell-type with labeled data, MTTFsite amounts to a **data augmentation method** due to the fact that labeled data in the target cell-type is augmented by labeled data available in other cell-types. When a target cell-type does not have any labeled data, only the learned common features are used to predict TFBSs. Thus, for the target cell-type without labeled data, the term **cross-cell-type** refers to the fact that MTTFsite can use labeled data available in other cell-types to learn common features by the shared CNN.

Gene expression predictions provide a foundation for understanding the transcriptional controls of cell identities, diseases, and cell-based therapies. Many computational methods were proposed for gene expression predictions. DeepChrome (Singh *et al.*, 2016), TEPIC (Schmidt *et al.*, 2017) and Zhang's method (Zhang and Li, 2017) are three state-of-the-art methods. DeepChrome (Singh *et al.*, 2016) is a unified end-to-end architecture constructed by using Convolutional Neural Network (CNN). The main advantage of DeepChrome is that it can capture both pairwise interactions between neighboring bins and between different histone modification features. However, DeepChrome does not use TFBSs of any TF in predictions. TEPIC is a segmentation-based method which first predicts TFBSs by applying PWMs to open-chromatin regions (Schmidt *et al.*, 2017) and then uses predicted TFBSs in gene expression predictions. Although TEPIC can predict TFBSs by applying PWMs, only a small portion of TFs have known PWMs so far. Also, predicted TFBSs by PWMs usually have very high false positive rate due to the lack of position dependencies in PWM. Zhang's method combines 10 histone modification features, TFBSs of 15 TFs and one DNase-I hypersensitivity profile for gene expression predictions (Zhang and Li, 2017). As TFBSs of the 15 TFs are identified by experimental methods, this method is limited to only a very small number of cell-types.

The objective of this work is to predict gene expressions for cell-types without experimentally identified TFBSs for any TF. We propose a novel gene expression prediction method, referred to as **TFChrome**, by combined use of TFBSs predicted by MTTFsite and histone modification features. As MTTFsite can predict TFBSs for TFs in most cell-types by leveraging on labeled data available in cross-cell-types, TFChrome is capable of predicting gene expression for most cell-types with or without labeled data.

2 Methods

2.1 Datasets

TFs in five cell-types, including GM12878, H1-hESC, HeLa-S3, HepG2 and K562, are used to evaluate our proposed method. As MTTFsite needs to be evaluated by TFs with labeled data in at least two cell-types, where one is used for testing and the others for training, a total of 72 TFs are used to evaluate MTTFsite, where 17, 14, 18, 23 TFs have labeled data in all the five cell-types, four cell-types, three cell-types and two cell-types, respectively. The available TFBSs of these TFs in these five cell-types are identified by TF ChIP-seq experiments and their peaks can be downloaded from ENCODE (ENCODE Project Consortium, 2004) freely. The obtained peaks are usually provided in one of two formats: **narrow peak** and **broad peak**. Some TFs have well defined binding sites and can be modeled by narrow peaks while binding sites of other TFs are less well localized and would better be modeled by broader peaks. So the narrow peak format is used if available. Otherwise, the broad peak format is used. Based on works by Alipanahi et al. (Alipanahi *et al.*, 2015) and Zeng et al. (Zeng *et al.*, 2016), the TFBS at each peak is defined as a 101 bp sequence by taking the midpoint of the peak as the center. Contrast to TFBSs, the non-TFBSs of a TF are defined as 101 bp DNA regions which cannot be bound by the target TF. Many works (Won *et al.*, 2010; Kumar and Bucher, 2016) used a shuffle method to construct non-TFBSs. In the shuffle method, a

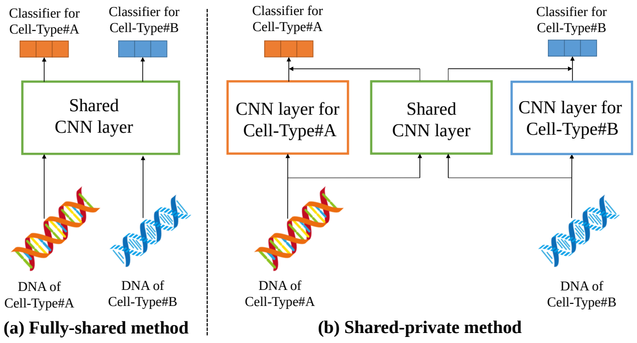
Classifier for Cell-Type#A
Classifier for Cell-Type#B
Shared CNN layer
DNA of Cell-Type#A
DNA of Cell-Type#B
(a) Fully-shared method
Classifier for Cell-Type#A
Classifier for Cell-Type#B
CNN layer for Cell-Type#A
Shared CNN layer
CNN layer for Cell-Type#B
DNA of Cell-Type#A
DNA of Cell-Type#B
(b) Shared-private method

Fig. 1. Architecture of multi-task learning for TFBS Prediction.

non-TFBS is constructed for each TFBS by shuffling the dinucleotides in the TFBS to keep the dinucleotide composition unchanged. In this study, however, as TFBSs need to be encoded by DNA sequences and histone modification features which need to be extracted from actual DNA sequences, we need to extract actual DNA fragments to construct non-TFBSs. So, we construct a non-TFBS for each TFBS by randomly selecting a 101 bp DNA fragment that has similar dinucleotide composition with the TFBS and is nonoverlapping with all TFBSs. This way, we can construct the same number of non-TFBSs as TFBSs for each TF. For each TF in each cell-type, the labeled data are divided into 3 separate, yet equal size folds: one fold for training, one fold for validation and one fold for test. The used TFs and its number of TFBSs in each cell-types are listed in Supplementary Table S1, which also can be accessed freely from our web-sever.

2.2 Feature Representation

Two types of features are used to represent TFBSs: sequence features and histone modification features. **Sequence features** of a TFBS are represented by the one-hot vectors of all its 101 nucleotides. For a TFBS T_i with the middle point at the position i in a genome, the sequence features can be represented by a feature matrix of dimension of 4×101 as follows

$$S_{T_i} = [O(N_{i-50}), \cdots, O(N_i), \cdots, O(N_{i+50})] \tag{1}$$

where $O(N_i)$ denotes the one-hot vector of nucleotide N_i. Seven types of **histone modification features** are used: *H3K4me2, H3K4me3, H4K20me1, H3K9ac, H3K27ac, H3K27me3* and *H3K36me3* as they are available for all the five considered cell-types. The ChIP-seq profiles for these histone modification features can be accessed freely from Kumar's work (Kumar and Bucher, 2016). Based on Won's work (Won *et al.*, 2010), we use the following scheme to apply histone modification features in MTTFsite: we first estimate the histone modification features for all nonoverlapping 25-bp bins and then estimate the histone modification features for each 100-bp bin by averaging the four 25-bp bins within it. Finally, histone modification features of the twenty 100-bp bins around a putative TFBS are concatenated to represent it. So the histone modification features for a TFBS T_i can be represented as

$$C_{T_i} = [H(N_{i-999}, \cdots, N_{i-898}), \cdots, H(N_{i-99}, \cdots, N_i), \cdots, H(N_{i+901}, \cdots, N_{i+1000})] \tag{2}$$

where $H(\cdot)$ denotes the histone modification features for a 100-bp bin. Since we use seven histone modification features, the histone modification features of a TFBS can be represented by a feature matrix with dimension of 7×20.

2.3 Convolutional Neural Network (CNN)

In recent years, CNN has been gradually introduced into bioinformatics to learn representations for protein sequences, DNA fragments and RNA fragments. For example, Alipanahi et al. (Alipanahi *et al.*, 2015) developed DeepBind to predict binding sites for DNA- and RNA-binding proteins by using CNN to learn representations for DNA fragments and RNA fragments. Wang et al. (Wang *et al.*, 2016) proposed a CNN based method to learn representations for proteins in protein secondary structure predictions. As the actual TFBSs of a TF often contain specific binding motifs, CNN is suitable to learn representations for TFBSs.

2.4 Multi-Task Learning for TFBS Prediction (MTTFsite)

Multi-task learning is an effective approach for improving the performance of a single task by leveraging on other related tasks (Liu *et al.*, 2017). Multi-task learning attempts to divide the features for multiple tasks into private and common features based on whether the features should be shared. Thus, in multi-task learning, each task contains both private features and common features. The private features of a task are the properties belonging to only this task while the common features are the characteristics shared by all the considered tasks. For TFBS predictions of a TF, the prediction in each cell-type can be defined as a task. Thus, TFBS predictions of a TF in multiple cell-types form a multi-task learning paradigm.

In multi-task learning, there can be two types of learning methods: the fully-shared model and the shared-private model (Liu *et al.*, 2017). The fully-shared model uses a single shared CNN to extract features for all cell-types, whose hypothesis is that features of individual cell-types are shared by all cell-types, as illustrated in Fig. 1(a). The feature space learned by the fully-shared model contains common features and also private features of each cell-type. Generally speaking, however, TFBSs of a TF in different cell-types may have common features and each cell-type may also has its own private features, not shared by other cell-types. Thus, private features of each cell-type will affect the prediction of other cell-types. A more serious issue is that, if some cell-types contain much more labeled data than others, the feature space learned by the fully-shared model may be dominated by private features of these cell-types, which will adversely affect the prediction of other cell-types with less labeled data, which is counter-productive to the goal of multi-task learning.

The shared-private model, on the other hand, contains a shared CNN to learn common features for all cell-types as well a private CNN for each cell-type to learn its private features. Features learned for every cell-type are separated into two subspace: the common feature space and the private feature space. In the prediction for each cell-type, its private features and the common features are integrated as the input. The separation of private features from common features makes sure that the private features of each cell-type will not affect the predictions of other cell-types. Thus, the shared-private model can leverage on labeled data available in other cell-types to learn solid information from common feature space, especially for cell-types with sparse or no labeled data. The shared-private model is illustrated in Fig. 1(b). Assuming for a TF in a cell-type (task) m, we have a dataset D_m with N_m instances, each instance is a pair of a putative TFBS x_i^m and its corresponding label y_i^m, that is:

$$D_m = \{(x_i^m, y_i^m)\}_{i=1}^{N_m} \tag{3}$$

As CNN is used to learn representations for all putative TFBSs, the private features h^m and the common features s^m of a putative TFBS x_i^m in the cell-type m learned by the shared-private model are formally formulated as:

$$h^m = \text{CNN}(x_i^m, \theta_m) \tag{4}$$

$$s^m = \text{CNN}(x_i^m, \theta_s) \tag{5}$$

where θ_m and θ_s are the parameters of the private CNN for the cell-type m and the shared CNN, respectively.

Our proposed MTTFsite follows the shared-private model. Thus, MTTFsite has the ability to separate private features of each cell-type from common features and can reduce the influence of private features of each cell-type to other cell-types. In MTTFsite, the network topology of the shared CNN and the private CNN for each cell-type contain two parallel CNN models: one is used to learn representations from sequence features and the other is used to learn representations from histone modification features. Then the common features and the private features of each cell-type are concatenated to represent instances and fed into a MLP for its prediction.

3 Experiments and Results

3.1 Experimental settings

In MTTFsite, the CNN models in both the shared CNN and private CNNs contain one convolution layer and each convolution layer consists of 200 convolution kernels of length 10. Each convolution layer is followed by a max pooling layer. A dropout regularization layer with dropout probability of 0.5 is used to avoid overfitting. The outputs of the shared CNN and the private CNN for the target cell-type are concatenated and inputted into the MLP of the target cell-type. The MLP consists of two fully connected layers of 200 neurons and a softmax classifier for predictions. We use Adagrad (Duchi *et al.*, 2011) with a batch size of 64 instances and default learning rate of 0.01. All these hyper-parameters are selected by carrying out experiments on validation set. During training, we train the model for 50 epochs. Once training is finished, we select the model with the highest accuracy on the validation set as our final model and evaluate its performance on the test set. All neural models are implemented in PyTorch.

To evaluate the performance of our proposed MTTFsite for TFBS prediction, we compare MTTFsite with two representative prediction methods: a baseline method and the fully-shared model. The baseline method is similar to the DeepBind method proposed by Alipanahi et al. (Alipanahi *et al.*, 2015) except that the baseline method also uses histone modification features as additional features. Both the baseline method and the fully-shared model contain two parallel CNN models: one is used to learn representations from sequence features and the other to learn representations from histone modification features. The two learned representations are concatenated and fed into a MLP for prediction. The hyperparameters of the baseline method and the fully-shared model have the same values as those used in MTTFsite.

3.2 Evaluation Metrics

AUC, F1-measure and Matthews Correlation Coefficient (MCC) are used as main metrics. AUC is the area under the Receiver Operating Characteristic curve. A ROC curve plots the true positive rate (sensitivity) versus the false negative rate (1-specificity) of different thresholds on the importance score. F1-measure is the harmonic average of the precision and recall. Precision is the fraction of true TFBSs among the predicted TFBSs, while recall is the fraction of true TFBSs that have been retrieved over the total amount of TFBSs. MCC is a correlation coefficient between the observed and predicted binary classifications. F1-measrue and MCC can be calculated by following formulae:

$$F1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{6}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{7}$$

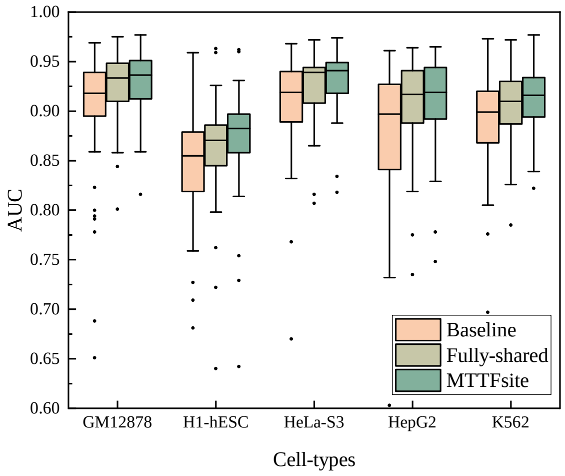

Fig. 2. Box plot depicting the AUC performance of data augmentation by the baseline method, the fully-shared model and MTTFsite on TFs in the five cell-types.

where TP, TN, FP and FN denote the number of true positives, the number of true negatives, the number of false positives and the number of false negatives.

3.3 Results of data augmentation using the fully-shared model

We first evaluate the performance of the fully-shared model on the TFs in the five cell-types and compare it with the baseline method. For each TF, the baseline method for each cell-type is trained by the training set of this cell-type, and is validated and tested by the validation set and the test set of this cell-type, respectively. By contrast, the fully-shared model of each cell-type is trained by the combined training data of all the cell-types and is validated and tested by the validation set and the test set of this cell-type, respectively.

The comparison between the fully-shared model and the baseline method in Supplementary Figure S1(A) and (B) shows that the fully-shared model performs better than the baseline method for most cell-type TF pairs. The box plot in Fig. 2 shows that the first quartile, the median and the third quartile of the AUC for the fully-shared model are higher than that of the baseline method for all the five cell-types. Details of AUC, F1-measure and MCC of the fully-shared model and the baseline method for each TF of the five cell-types are listed in Supplementary Table S2. Results show that the fully-shared model outperforms the baseline method for 49 TFs out of the 56 TFs in GM12878, 31 TFs out of the 42 TFs in H1-hESC, 33 TFs out of the 37 TFs in HeLa-S3, 42 TFs out of the 43 TFs in HepG2 and 60 TFs out of the 63 TFs in K562. These are the evidences that multi-task learning can indeed improve the performance of TFBS predictions in most cell-type TF pairs through labeled data available in cross-cell-types. Thus, we can come to a conclusion that the TFBSs of a TF in multiple different cell-types indeed have common features and the common features can be learned by the combined use of the available labeled data from multiple cell-types.

3.4 Results of data augmentation by MTTFsite

The feature space learned by the fully-shared model contains both common features of all the cell-types and private features of each cell-type. The prediction for each cell-type would be influenced by the private features of other cell-types as were the case of the fully-shared model. Our proposed MTTFsite separates the learning of private features of each cell-type from that of the common features. For MTTFsite in data augmentation, each

Table 1. Details of the AUC comparison between MTTFsite and the baseline method for data augmentation.

Cell-type	GM12878	H1-hESC	HeLa-S3	HepG2	K562	Average^c
Sample total	56	42	37	43	63	48.2
Improvement total	52	37	34	41	60	44.8
Improvement (%)	92.9	88.1	79.1	95.3	95.2	92.9
Maximum^a (%)	31.7	12.7	22.1	37.8	17.9	24.5
Average^b (%)	3.6	3.5	3.2	3.8	2.9	3.4

^{a} and ^{b} denotes the maximum improvement and the average improvement, respectively; ^{c} denotes the micro average over the total number of samples.

Table 2. Details of the AUC comparison between MTTFsite and the fully-shared model for data augmentation.

Cell-type	GM12878	H1-hESC	HeLa-S3	HepG2	K562	Average^c
Sample total	56	42	37	43	63	48.2
Improved total	47	39	37	39	59	44.2
Improvement %	83.9	92.9	100	90.7	93.7	91.7
Maximum^a (%)	2.2	2.8	2.9	2.2	4.7	3.1
Average^b (%)	0.6	1.2	0.7	0.6	0.8	0.8

^{a} and ^{b} denotes the maximum improvement and the average improvement, respectively; ^{c} denotes the micro average over the total number of samples.

private CNN is trained by the training set of the corresponding cell-type while the shared CNN is trained by combined training data of all cell-types. In order to evaluate the usefulness of feature separation, we compare the performance of MTTFsite with both the baseline method and the fully-shared model.

The comparison among the baseline method, the fully-shared model, and our proposed MTTFsite is shown in Supplementary Figure S1. Figure S1(B), (C) and (D) show that MTTFsite performs better than both the baseline method and the fully-shared model for most cell-type TF pairs, although the margin of improvement over the fully-shared model is smaller compared to that of the baseline method. The box plot in Fig. 2 shows that the first quartile, the median and the third quartile of the AUC for the MTTFsite are higher than that of the fully-shared model and the baseline method for all the five cell-types. Details of AUC, F1-measure and MCC for the baseline method, the fully-shared model, and our proposed MTTFsite for each TF of the five cell types are listed in Supplementary Table S2. Table 1 summarizes the AUC performance gain of MTTFsite compared to the baseline method. For the five cell-types, MTTFsite performs better than the baseline method on at least 79.1% TFs of all cell-types. The maximum improvement and the average improvement are 12.7% and 2.9% at least, respectively. On average, MTTFsite performs better than the baseline method in more than 92.9% of TFs. The micro average of the maximum improvement and the average improvement are 24.5% and 3.4%, respectively. The improvements are very significant as shown by p-value = 4.71×10^{-37} in Wilcoxon signed-ranks test.

Table 2 summarizes the AUC performance gain of MTTFsite compared to the fully-shared model. For the five cell-types, MTTFsite performs better than the fully-shared model in at least 83.9% of TFs for each cell-type. The maximum improvement and the average improvement are at least 2.2% and 0.6%, respectively. On average, MTTFsite performs better than the fully-shared model significantly in more than 91.7% of TFs with the maximum improvement and the average improvement of 3.1% and 0.8%, respectively (p-value = 7.71×10^{-30} by Wilcoxon signed-ranks test). Moreover, for some TFs, MTTFsite achieves very promising improvements. For example, the improvements on BCL11A and RXRA in H1-hESC are 2.0% and 2.3%, respectively; the improvements on RAD21 and SMC3 in HeLa-S3 are 2.5% and 2.0%, respectively; the improvements on RAD21 and TR4 in K562 are 2.5% and 3.7%, respectively.

3.5 Comparison between MTTFsite and state-of-the-art methods

Recent works with state-of-the-art performance include DNA shape based mehtod, PWM, DWM as well as deep learning methods. This section will first present comparison of our work with the use of DNA shape features and then proceed to comparison with PWM, DWM and deep learning methods.

DNA shapes represent the 3D structures of DNA. Recently, Mathelier at al. (Mathelier *et al.*, 2016a) proposed four models for TFBS predictions in vivo by using DNA shape features including helix twist (HelT), minor groove width (MGW), propeller twist (ProT), and the Roll. These four DNA shape features and their corresponding second-order shape features (Zhou *et al.*, 2015), used to represent putative TFBSs, were computed by DNAshape (Chiu *et al.*, 2015; Zhou *et al.*, 2013). Four DNAshape based models we compared with include: (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 (Mathelier and Wasserman, 2013a) and DNA shape features, and (4) TFFM_f+shape, which combines 1st-order TFFM scores (Mathelier and Wasserman, 2013a) and DNA shape features. The implementation of the four existing models is all available from the software download webpage (http://github.com/amathelier/DNAshapedTFBS). They are implemented in our comparison using their default setup and parameters. In addition to DNAshape, DynaSeq proposed by Andrabi et al. (Andrabi *et al.*, 2017) can also be used to predict DNA shape features. DynaSeq predicts molecular dynamics-derived ensembles of a more exhaustive set of DNA shape features. In this study, we also compare MTTFsite with DynaSeq. Supplementary Table S3 shows the AUC of MTTFsite, the four DNAshape based models and DynaSeq on five TFs in the five cell-types with a total of 24 cell-type TF pairs. Results show that DynaSeq achieves higher AUC than the four DNAshape based models on 14 cell-types TF pairs. It indicates that DNA shape features predicted by DynaSeq are more useful than those predicted by DNAshape, which is consistent with the conclusion drawn in the original publication (Andrabi *et al.*, 2017). Results also show that our proposed MTTFsite achieves higher AUC than the four DNAshape based models and DynaSeq for 22 cell-type TF pairs. The minimum improvement and the maximum improvement are 2% on GABP in HepG2 and 30% on JunD in GM12878, respectively. The average improvement is 11.6%, which is a very large improvement for TFBS predictions. This first confirms that MTTFsite is more useful than the four DNAshape based models and DynaSeq for TFBS prediction. One possible reason that MTTFsite outperforms the use of DNA shape features is that DNA shape features are predicted by computational methods from DNA sequences. Thus, there may be redundancy with sequence features. Furthermore, predicted DNA shape features may contain many noises.

Current state-of-the-art methods include PWM (Stormo, 2000, 2013), DWM (Siddharthan, 2010) and three deep learning methods: DeepSEA (Zhou and Troyanskaya, 2015), DanQ (Quang and Xie, 2016) and DanQ-JASPAR (Quang and Xie, 2016). PWM and DWM are two useful representation methods for TFBSs and achieved good performance (Mathelier and Wasserman, 2013b). DeepSEA applies CNN and DanQ combines CNN with Recurrent neural network (RNN) to learn features for TFBSs. DanQ-JASPAR, an alternative model of DanQ, was developed by initializing half of the kernels in CNN with motifs from the JASPAR database (Mathelier *et al.*, 2016b). In this evaluation, we implemented PWM and DWM based on Mathelier's work (Mathelier and Wasserman, 2013b). We downloaded DeepSEA from its software's webpage (http://DeepSEA.princeton.edu/) and DanQ as well as DanQ-JASPAR from their software's webpage (http://github.com/uci-cbcl/DanQ). Performance data for DeepSEA, DanQ and DanQ-JASPAR

Table 3. The AUC of five state-of-the-art methods and MTTFsite on five TFs in five cell-types.

TF	Cell-type	PWM	DWM	DanQ	DanQ-J	DeepSEA	MTTFsite
CTCF	GM12878	0.586	0.578	0.765	0.731	0.677	**0.859**
	H1-hESC	0.566	0.575	0.794	0.758	0.689	**0.816**
	HeLa-S3	0.505	0.509	0.720	0.698	0.670	**0.834**
	HepG2	0.523	0.527	0.796	0.757	0.697	**0.871**
	K562	0.923	**0.938**	0.728	0.693	0.635	0.839
GABP	GM12878	0.844	0.844	0.797	0.845	0.791	**0.934**
	H1-hESC	0.721	0.740	0.789	**0.791**	0.763	0.729
	HeLa-S3	0.877	0.875	0.658	0.681	0.630	**0.946**
	HepG2	0.786	0.791	0.794	0.838	0.795	**0.864**
	K562	0.756	0.754	0.775	0.793	0.763	**0.913**
JunD	GM12878	0.906	0.919	0.621	0.606	0.589	**0.957**
	H1-hESC	0.557	0.566	0.693	0.686	0.643	**0.876**
	HeLa-S3	0.863	0.860	0.777	0.788	0.711	**0.942**
	HepG2	**0.925**	0.878	0.813	0.826	0.738	0.829
	K562	0.684	0.687	0.655	0.653	0.595	**0.912**
REST	GM12878	0.906	0.919	0.621	0.606	0.589	**0.957**
	HeLa-S3	0.899	0.922	0.602	0.597	0.559	**0.940**
	HepG2	0.886	0.902	0.630	0.603	0.602	**0.911**
	K562	0.867	0.890	0.646	0.645	0.623	**0.905**
USF2	GM12878	0.891	0.891	0.673	0.698	0.615	**0.938**
	H1-hESC	0.841	0.851	0.729	0.752	0.662	**0.887**
	HeLa-S3	0.908	0.912	0.641	0.654	0.561	**0.938**
	HepG2	0.952	**0.953**	0.697	0.751	0.591	0.904
	K562	0.921	0.926	0.660	0.715	0.580	**0.945**

DanQ-J denotes DanQ-JASPAR. The bold and underscore numbers denote the best performer and second best performer, respectively.

is the result of using their default setup and parameters. We compare MTTFsite with these five state-of-the-art methods by five TFs in the five cell-types with a total of 24 cell-type TF pairs. As MTTFsite is trained by datasets in five cell-types and 7 histone marks, we trained DeepSEA, DanQ and DanQ-JASPAR for each TF with the TF binding profiles in the five cell-types and the 7 histone-mark profiles to make a fair comparison. Table 3 shows the AUC of our proposed MTTFsite and the five state-of-the-art methods on the 24 cell-type TF pairs. Results show that DWM achieves higher or equal AUC than PWM for 20 cell-type TF pairs, which is consistent with the conclusion of the original publication (Siddharthan, 2010). DanQ achieves higher AUC than DanQ-JASPAR on 12 cell-type TF pairs and achieves lower AUC than DanQ-JASPAR on the remaining pairs. This indicates that DanQ and DanQ-JASPAR have comparable performances. DanQ performs better than DeepSEA for most cell-type TF pairs, which is consistent with the result reported in the original publication (Quang and Xie, 2016). Most noticeably, MTTFsite performs better than the five state-of-the-art methods in 21 out of the 24 cell-type TF pairs. On the 21 pairs, the minimum, the maximum and the average improvement are 0.9%, 22.5% and 6.0%, respectively. It should be noted that the performance of DeepSEA, DanQ and DanQ-JASPAR are much better in their reported original publications. However, their performance in this study is much worse. The main reason is that the original models are trained by 690 TF binding profiles for 160 different TFs, 125 DHS profiles as well as 104 histone-mark profiles while the models in this study is trained by TF binding profiles of only five cell-types and 7 histone-mark profiles. It indicates that the performance of DeepSEA, DanQ and DanQ-JASPAR closely relies on large number of data sets.

3.6 Results of cross-cell-type prediction by MTTFsite

Due to the high cost of TF ChIP-seq experiments, many cell-types only have labeled data for very limited portion of TFs. Most TFs are not labeled. This motivates us to use computational methods to predict TFBSs for TFs in those cell-types that have no labeled data for them. As our proposed MTTFsite can use a shared CNN to learn common features by leveraging on the available labeled data from available cell-types, it aims to predict TFBSs for TFs in the cell-types without labeled data for them. This is what we refer to as cross-cell-type predictions. To evaluate the performance of MTTFsite for cross-cell-type TFBS prediction, we assume that only the test set of the target cell-type is available while the training set as well as the validation set are unavailable. In cross-cell-type prediction, MTTFsite trains both the shared CNN of all cell-type and the private CNN of the target cell-type by combined training data of cross-cell-types. MTTFsite is validated by combined validation set of cross-cell-types and then tested on the test set of the target cell-type. We compare the performance of cross-cell-type prediction by MTTFsite with the fully-shared model and the baseline method. The fully-shared model is trained and validated by cross-cell-types like MTTFsite and the baseline method is trained and validated by the target cell-type.

Table 4. Details of the AUC comparison between MTTFsite and the baseline method for cross-cell-type prediction.

Cell-type	GM12878	H1-hESC	HeLa-S3	HepG2	K562	Average[c]
Sample total	56	42	37	43	63	48.2
Improvement total	46	31	29	35	54	39
Improvement (%)	82.1	73.8	78.4	81.4	85.7	80.9
Maximum[a] (%)	40.9	31.0	25.7	42.0	34.7	36.9
Average[b] (%)	5.1	8.0	4.1	5.1	4.0	5.1

[a] and [b] denotes the maximum improvement and the average improvement, respectively; [c] denotes the micro average over the total number of samples.

The comparison among the baseline method, the fully-share model and our proposed MTTFsite is shown in Supplementary Figure S2. Figure S2(A) and (B) show that the fully-shared model performs better than the baseline method for most cell-type TF pairs. The box plot in Fig. 3 shows that the first quartile, the median and the third quartile of the AUC for the fully-shared model are higher than that of the baseline method for all the five cell-types. It indicates that the use of information of cross-cell-types is useful and can achieve better performance than the baseline method which is trained by the target cell-type. Figure S2(B), (C) and (D) show that MTTFsite performs better than both the baseline method and the fully-shared model for most cell-type TF pairs. The box plot in Fig. 3 shows that the first quartile, the median and the third quartile of the AUC for MTTFsite are higher than that of both the baseline method and the full-shared model for all the five cell-types. Details of AUC, F1-measure and MCC for these three methods on TFs in the five cell-types are listed as Supplementary Table S4. Table 4 summarizes the AUC performance gain of MTTFsite compared to the baseline method for cross-cell-type TFBS predictions. For the five cell-types, MTTFsite outperforms the baseline method on at least 73.8% TFs of each cell-type. The maximum improvement and the average improvement are at least 25.7% and at least 5.1%, respectively. On average, MTTFsite outperforms the baseline method in more than 80.9% of TFs. The micro average of the maximum improvement and the average improvement are 36.9% and 5.1%, respectively. The improvement is very significant according to p-value = 1.42×10^{-23} by Wilcoxon signed-ranks test.

Table 5 summarizes the AUC performance gain of MTTFsite compared to the fully-shared model. For the five cell-types, MTTFsite performs better than the fully-shared model in at least 88.1% of TFs for each cell-type. The maximum improvement and the average improvement are at least 3.5% and at least 1.2%, respectively. On average, MTTFsite performs better than the fully-shared model significantly in more than 94.2% of TFs with the maximum improvement and the average improvement of 4.0% and 1.3%, respectively (p-value = 4.55×10^{-13} by Wilcoxon signed-ranks test).

Table 5. Details of the AUC comparison between MTTFsite and the fully-shared model for cross-cell-type prediction.

Cell-type	GM12878	H1-hESC	HeLa-S3	HepG2	K562	Average^c
Sample total	56	42	37	43	63	48.2
Improvement total	54	37	36	41	59	45.4
Improvement (%)	96.4	88.1	97.3	95.3	93.7	94.2
Maximum^a (%)	4.2	3.6	3.5	4.0	4.4	4.0
Average^b (%)	1.2	1.5	1.2	1.4	1.3	1.3

^{a} and ^{b} denotes the maximum improvement and the average improvement, respectively; ^{c} denotes the micro average over the total number of samples.

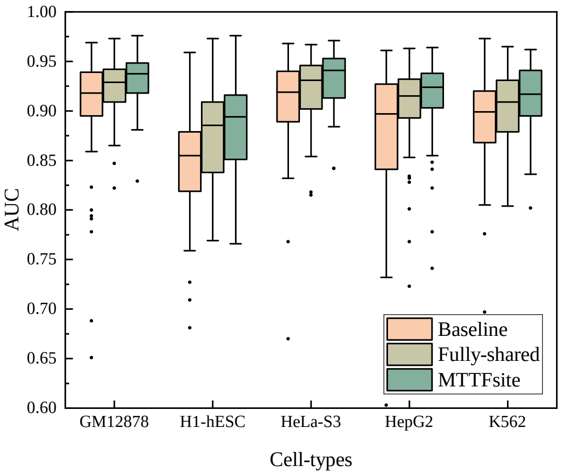

Fig. 3. Box plot depicting the AUC performance of cross-cell-type prediction by the baseline method, the fully-shared model and MTTFsite on TFs in the five cell-types.

The improvements for many TFs are quite promosing. For example, the improvements for RAD21 and MAFK in H1-hESC and CTCF, RAD21 and SMC3 in K562 are more than 3.0%; the improvements for CTCF and EZH2 in GM12878, CTCF in HeLa-S3, NRSF in HepG2 are more than 4.0%. It is a strong indication that MTTFsite has a better prediction power than that of the fully-shared model.

By comparing MTTFsite with the full-shared model, we find that the private CNNs of the cell-types without labeled data in MTTFsite function similarly to the shared CNN in the fully-shared model because they both are trained by the combined training data from cross-cell-types. The only difference is that MTTFsite contains both features learned by private CNNs and by the shared CNN whereas the fully-shared model only uses features learned by the shared CNN. In the fully-shared model, if some cell-types contain too much training data, the learned features are dominated by private features of these cell-types such that many common features are lost. As MTTFsite can separate private features from common features, the lost common features in the private CNNs can be complemented by the common features learned by the shared CNN. Therefore, the features learned by MTTFsite for each cell-type contain more common features than that learned by the fully-shared model.

In order to further demonstrate the performance of MTTFsite for cross-cell-type prediction, we evaluate MTTFsite on TFs in K562 cells from PIQ study (Sherwood *et al.*, 2014), which are available from online resource located at (http://piq.csail.mit.edu/data/141105-3618f89-hg19k562.calls/141105-3618f89-hg19k562.calls.tar.gz). Although there are a total of 1316 TFs with genome-wide TFBSs available in K562 from PIQ study, only 28 TFs have training set in at least one cell-type of the five cell-types in this study except K562. So MTTFsite is only tested on the 28 TFs with available training data. In Andrabi's work (Andrabi *et al.*, 2017), TFBSs are selected from the "calls" data and equal number of non-TFBSs are selected with the cut-off score of 0.25, where the maximum number of TFBSs and non-TFBSs was fixed at 2000 by random sampling. However, in order to evaluate MTTFsite on genome scale, we collected all the TFBSs from the "calls" data and equal number of non-TFBSs to make up test set. Thus, for each TF, MTTFsite is trained by the combined training data available in the four cell-types in this study and tested on the test set from PIQ study. The performance is listed in Supplementary Table S5. Results show that MTTFsite achieves good performance on most TFs and the AUC performance on seven TFs is more than 0.8. As training data comes from ChIP-Seq while testing data comes from DNAse-Seq, results indicate that MTTFsite can be applied for cross-platform prediction.

ENCODE-DREAM in vivo Transcription Factor Binding Challenge contains a Across-Cell Type Prediction Challenge, in which each TF has cell-types for training and held-out cell-types for testing. We downloaded the 13 cell-type TF pairs in the Final Submission Round. For each TF, MTTFsite is trained by at least one cell-type and tested by held-out cell-types, which are newly generated and have never been previously released by ENCODE. As the challenge do not provide histone modification features, MTTFsite is trained only from DNA sequences and chromatin accessibility measured by DNAse-Seq. The advantage of MTTFsite is that it can learn common features in histone modification features for TFBSs shared by multiple cell-types. Even though, MTTFsite trained from DNA sequence and chromatin accessibility cannot fully demonstrate the advantages of our method , MTTFsite still achieves very good performance on the 13 cell-type TF pairs. The performance is listed in Supplementary Table S6. Table S6 shows that MTTFsite achieves good performance for all the 13 cell-type TF pairs. Specifically, AUC of all the 13 pairs is more than 0.9 and AUC of 7 pairs is even more than 0.95. Results indicate that MTTFsite can achieve good performance for cross-cell-type TFBS prediction even when histone modification features are not available.

3.7 Results on cell-type shared TFBS and cell-type specific TFBS

One advantage of MTTFsite is that it can leverage on cell-type shared TFBSs available in other cell-types to train the shared CNN. To validate this, we evaluate the performance of MTTFsite on cell-type shared TFBSs and cell-type specific TFBSs, separately. In this study, cell-type shared TFBSs of a cell-type are defined as the TFBSs which have at least a TFBS of other cell-types in its range of 100 bp. The remaining TFBSs are referred to as cell-type specific TFBSs. According this criterion, TFBSs of each cell-type are divided into cell-type shared TFBSs and cell-type specific TFBSs. Details of the number of cell-type shared TFBSs and cell-type specific TFBSs for TFs in the five cell-types is listed in Supplementary Table S7. For each target cell-type, MTTFsite is trained by combined labeled data available in cross-cell-types and tested on cell-type shared TFBSs and cell-type specific TFBSs of the target cell-type, separately. Sensitivity is used to evaluate the performance of MTTFsite. The sensitivity of MTTFsite for TFs in the five cell-types is listed in Fig. 4. Fig. 4(A) shows that MTTFsite achieves higher sensitivity on cell-type shared TFBSs than cell-type specific TFBSs for all cell-type TF pairs except one. Fig.4(B) shows that the first quartile, the median and the third quartile of the sensitivity for cell-type shared TFBSs are higher than that for cell-type specific TFBSs for TFs in all the five cell-types. Details of the sensitivity for cell-type shared and specific TFBSs for TFs in the five cell-types are listed as Supplementary Table S8. Results indicate that MTTFsite indeed can effectively leverage on cell-type shared TFBSs available in cross-cell-types to learn common features of all cell-types.

As MTTFsite achieves higher sensitivity on shared TFBSs than specific TFBSs in almost all the five cell-types for each TF, the shared TFBSs dominate the performance of MTTFsite. If other cell-types have more shared TFBSs available by target cell-types, MTTFsite can achieve higher prediction performance. Therefore, high-quality predictions of MTTFsite

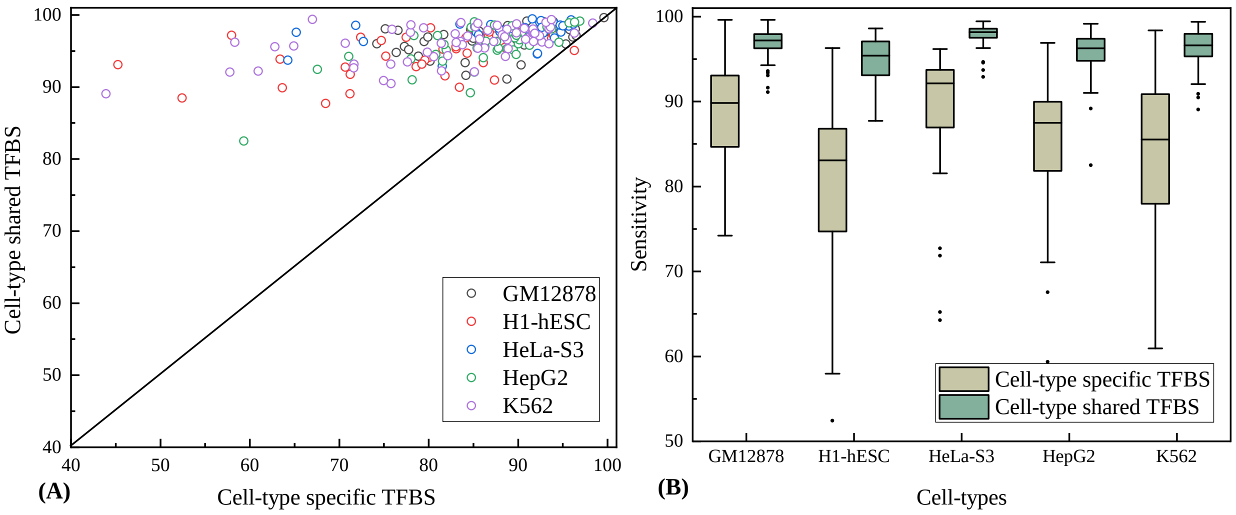

Fig. 4. (A) Scatter plot depicting the distribution of the AUC performance for cell-type shared TFBSs and cell-type specific TFBSs. (B) Box plot depicting the AUC performance for cell-type shared TFBSs and cell-type specific TFBSs on TFs in the five cell-types.

for each TF rely on available TFBSs shared by target cell-types and other cell-types.

It should also be noted that Fig. 4(B) shows that specific TFBSs in H1-hESC achieve the lowest sensitivity among the five cell-types. It is possible that MTTFsite achieves low sensitivity scores for specific TFBSs in H1-hESC because specific TFBSs in H1-hESC have different characteristics compared to other cell-types for some TFs. Based on this hypothesis, we conducted an additional experiment to calculate the cosine similarities among the five cell-types for both specific TFBSs and shared TFBSs of each TF. For each TF, we first represent specific TFBSs and shared TFBSs by histone modification features and calculate their center in each cell-type by calculating the median value of each histone modification feature. Then, based on these centers, we calculate the cosine similarity between any two cell-types. Finally, for each cell-type, its cosine similarities to other cell-types are averaged. The average cosine similarities of the five cell-types for both specific TFBSs and shared TFBSs of each TF are shown in Fig. 5. The figure shows that the cosine similarities of shared TFBSs are higher than that of specific TFBSs in the five cell-types. This explains why MTTFsite achieves higher performance for shared TFBSs than that for specific TFBSs. Fig. 5 also shows that specific TFBSs in H1-hESC have the lowest cosine similarity to other cell-types among the five cell-types. This is indeed likely the reason that specific TFBSs in H1-hESC achieve the lowest sensitivity. Fig. 5 further shows that K562 has lower cosine similarity than the other three cell-types. This explains why specific TFBSs in K562 achieve lower sensitivity than the other three cell-types. The other three cell-types have small cosine similarity differences, so their specific TFBSs have small sensitivity differences.

nextPBM has been proposed to characterize the impact of cofactors and phosphorylation on TF binding and determine cell-type specific TFBSs (Mohaghegh *et al.*, 2019). The authors analyzed DNA binding of PU.1/SPI1 and IRF8 from human monocytes and found that cofactors and phosphorylation have no effect on autonomous PU.1/SPI1 binding and only have effect on its cooperative binding with monocyte-specific cofactors. Thus, nextPBMs can only identify cell-type specific cooperative TFBSs of PU.1/SPI1 with IRF8. As our proposed MTTFsite needs cell-type specific TFBSs to learn cell-type specific features by private CNNs and PU.1/SPI1 does not have cell-type specific TFBSs, current datasets used by nextPBM are inappropriate to improve MTTFsite. Nevertheless, nextPBM is capable of identifying cell-type specific TFBSs by comparing TFBSs from nuclear extracts to that from in vitro transcription/translation protein. Therefore, in the future, we can apply nextPBM to identify cell-type specific TFBSs for TFs. This should help to improve MTTFsite for cell-type specific TFBS prediction by learning cell-type specific features through private CNNs using the identified cell-specific TFBSs identified by nextPBM.

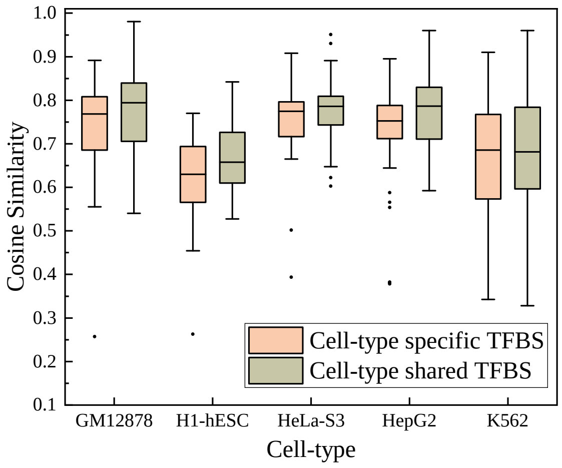

Fig. 5. Cosine similarities of cell-type specific TFBSs in different cell-types.

3.8 Application in gene expression prediction

TFs can bind to DNA through TFBSs to regulate gene expression. Therefore, we hypothesize that TFBSs are significant for gene expression regulations and can play an important role in gene expression prediction.

In this work, we propose a new gene expression prediction method, referred to as **TFChrome**, by combining the use of TFBSs predicted by MTTFsite and histone modification features. We evaluate TFChrome by 20 cell-types from the Roadmap Epigenomics Consortium (RMEC) (Kundaje *et al.*, 2015). These 20 cell-types have seven common histone modification features (Boyle *et al.*, 2008; Crawford *et al.*, 2006). Since these 20 cell-types do not have available labeled data for any TF, MTTFsite combines the available labeled data from GM12878, H1-hESC, HeLa-S3, HepG2

Table 6. The AUC of the gene expression predictions on the 20 cell-types from RMEC.

Cells	TFBS	Histone	Combine
Breast_vHMEC	0.779	0.859	**0.864**
Fetal_Brain	0.764	0.848	**0.855**
Fetal_Muscle_Leg	0.773	0.854	**0.858**
Fetal_Muscle_Trunk	0.759	0.802	**0.849**
Gastric	0.752	0.813	**0.819**
H1_BMP4_Derived_Mesendoderm_Cultured_Cells	0.746	0.787	**0.827**
H1_BMP4_Derived_Trophoblast_Cultured_Cells	0.751	0.831	**0.840**
H1_Cell_Line	0.754	0.837	**0.844**
H1_Derived_Mesenchymal_Stem_Cells	0.782	0.833	**0.839**
H1_Derived_Neuronal_Progenitor_Cultured_Cells	0.752	0.833	**0.839**
IMR90_Cell_Line	0.789	0.852	**0.860**
iPS_DF_19.11_Cell_Line	0.744	0.808	**0.813**
iPS_DF_6.9_Cell_Line	0.746	0.823	**0.826**
Mobilized_CD34_Primary_Cells	0.797	0.872	**0.878**
Pancreas	0.754	0.824	**0.832**
Penis_Foreskin_Fibroblast_Primary_Cells	0.815	0.885	**0.891**
Penis_Foreskin_Keratinocyte_Primary_Cells	0.794	0.872	**0.880**
Penis_Foreskin_Melanocyte_Primary_Cells	0.801	0.875	**0.881**
Psoas_Muscle	0.767	0.801	**0.858**
Small_Intestine	0.767	0.835	**0.840**

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

and K562 as training data to predict TFBSs for TFs in these 20 cell-types. More specifically, we predict the TFBSs for 72 TFs in the 20 cell-types, which are listed in Supplementary Table S1. As the 20 cell-types from RMEC and the five cell-types with labeled data contain seven common histone modification features including *H3K27ac, H3K37me3, H3K36me3, H3K9ac, H3K9me3, H3K4me1* and *H3K4me3*, these seven histone modification features are used in both the TFBS prediction and the gene expression prediction. Details of the definition of gene expression prediction and the used gene encoding method are given in Supplementary Methods.

To consider the relative importance of predicted TFBSs and histone modification features, we use two baseline methods for comparison: (1) using only predicted TFBSs and (2) using only histone modification features. our proposed TFChrome combines both the predicted TFBSs and histone modification features. Table 6 gives the performance evaluation of the three methods. Note that the maximum, the minimum and the average AUC of prediction using only predicted TFBSs are 0.815, 0.744 and 0.769, far better than random guessing. This is a strong indication that our hypothesis is correct that TFBSs indeed play an important role in gene expression predictions.

Table 6 also shows that TFchrome outperforms the method using only histone modification features. The Wilcoxon signed-ranks test with p-value of at least 3.36e-5 also indicates that the improvement is very significant. For some cell-types, the performance improvement by TFChrome is quite prominent. For example, for Fetal_Muscle_Trunk, H1_BMP4_Derived_Trophoblast_Cultured_Cells and Psoas_Muscle, the improve in AUC are 3.3%, 4.0% and 5.7%, respectively. These are evidences that TFBSs predicted by our proposed MTTFsite and histone modification features are complementary for gene expression predictions.

Several computational methods were proposed for gene expression predictions. TEPIC (Schmidt *et al.*, 2017), Zhang's method (Zhang and Li, 2017) and DeepChrome (Singh *et al.*, 2016) are three methods with state-of-the-art performance. As the used data sets and the definition for the problem of gene expression prediction in TFChrome are different from TEPIC and Zhang's method, we only compare TFChrome with DeepChrome. DeepChrome, proposed by Singh at al. (Singh *et al.*, 2016), uses CNN and histone modification features, which outperforms most previous methods. As TFChrome has 15 cell-types common with DeepChrome, we compare them on those 15 cell-types. Supplementary Table S9 shows the performance comparison of TFChrome and DeepChrome. Note that the AUC of DeepChrome on the 15 common cell-types are given directly from Singh's work. Table S9 shows that our proposed TFChrome performs far better than DeepChrome on 14 out of the 15 common cell-types. The maximum, the minimum and the average improvement in AUC is 12%, 1.7% and 6.2%, respectively, which are quite large. As both methods use the histone modification features, the main difference is that TFChrome also use the additional feature from the predicted TFBSs. Thus, it is fair to say that the improvement is contributed by the predicted TFBSs using MTTFsite.

4 Conclusion

In this paper, we present a novel data augmentation method using multi-task learning framework, MTTFsite, for TFBS predictions. MTTFsite contains a shared CNN to learn common features of all cell-types and a private CNN for each cell-type to learn private features. The aim of the algorithm is to make use of common features cross different cell-types to help predicting TFBSs for TFs in cell-types that have no labeled data. Performance evaluation shows MTTFsite can effectively leverage on labeled data available in cross-cell-types to learn common features of all cell-types. As MTTFsite can separate private features from common features, it outperforms the fully-shared model significantly. For cross-cell-type prediction, MTTFsite also outperforms the compared models. This is a clear indication that common features learned by MTTFsite from labeled data available in cross-cell-types are indeed useful for cross-cell-type predictions. To further prove the usefulness of MTTFsite, we propose to make use of the predicted TFBSs for gene expression prediction. The new gene expression prediction method TFChrome makes combined use of the TFBSs predicted by MTTFsite and histone modification features. The evaluation on 20 cell-types shows that TFBSs predicted by MTTFsite significantly improves the performance of gene expression predictions compared to the state-of-the art methods. Gene expressions of organisms are closely related to identification of diseases. For example, low expression of BRCA1 plays an important role in breast and ovarian cancers. Therefore, accurate gene expressions predicted by our proposed TFChrome can provide valuable reference and assistance for the diagnosis and treatment of dozens of diseases.

One direction of future works is to investigate the relative importance of labeled data from different cell-types in cross-cell-type TFBS prediction. The second direction is to investigate the prediction of TFs of cell-types without any labeled data by using labeled data of other TFs from the same cell-type, which is also referred to as cross-TF TFBS predictions.

Funding

This work was supported by the National Natural Science Foundation of China U1636103, 61632011, 61876053, Shenzhen Foundational Research Funding JCYJ20170307150024907, JCYJ20180507183527919, Key Technologies Research and Development Program of Shenzhen JSGG20170817140856618 and H-2020(no. 794196).

References

Alipanahi, B., Delong, A., Weirauch, M. T., and Frey, B. J. (2015). Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning. *Nature biotechnology*.

Andrabi, M., Hutchins, A. P., Mirandasaavedra, D., Kono, H., Nussinov, R., Mizuguchi, K., and Ahmad, S. (2017). Predicting conformational ensembles and genome-wide transcription factor binding sites from dna sequences. *Scientific Reports*, **7**(1), 4071.

Barash, Y., Elidan, G., Friedman, N., and Kaplan, T. (2003). Modeling dependencies in protein-DNA binding sites. In *Proceedings of the seventh annual international conference on Research in computational molecular biology*, pages 28–37. ACM.

Berger, M. F., Philippakis, A. A., Qureshi, A. M., He, F. S., Estep, P. W., and Bulyk, M. L. (2006). Compact, universal DNA microarrays to comprehensively determine transcription-factor binding site specificities. *Nature biotechnology*, **24**(11), 1429–1435.

Boyle, A. P., Davis, S., Shulha, H. P., Meltzer, P., Margulies, E. H., Weng, Z., Furey, T. S., and Crawford, G. E. (2008). High-resolution mapping andcharacterization of open chromatin across the genome. *Cell*, **132**(2), 311–322.

Breiman, L. (2001). Random forests. *Machine learning*, **45**(1), 5–32.

Bryne, J. C., Valen, E., Tang, M.-H. E., Marstrand, T., Winther, O., da Piedade, I., Krogh, A., Lenhard, B., and Sandelin, A. (2007). JASPAR, the open access database of transcription factor-binding profiles: new content and tools in the 2008 update. *Nucleic acids research*, **36**(suppl_1), D102–D106.

Bulyk, M. L. (2003). Computational prediction of transcription-factor binding site locations. *Genome biology*, **5**(1), 201.

Bulyk, M. L., Johnson, P. L., and Church, G. M. (2002). Nucleotides of transcription factor binding sites exert interdependent effects on the binding affinities of transcription factors. *Nucleic acids research*, **30**(5), 1255–1261.

Chiu, T. P., Comoglio, F., Zhou, T., Yang, L., Paro, R., and Rohs, R. (2015). DNAshapeR: an R/Bioconductor package for DNA shape prediction and feature encoding. *Bioinformatics*, **32**(8), 1211.

Crawford, G. E., Holt, I. E., Whittle, J., Webb, B. D., Tai, D., Davis, S., Margulies, E. H., Chen, Y. D., Bernat, J. A., and Ginsburg, D. (2006). Genome-wide mapping of dnase hypersensitive sites using massively parallel signature sequencing MPSS. *Genome research*, **16**(1), 123.

Dror, I., Rohs, R., and Mandel-Gutfreund, Y. (2016). How motif environment influences transcription factor search dynamics: Finding a needle in a haystack. *BioEssays*, **38**(7), 605–612.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, **12**(7), 257–269.

ENCODE Project Consortium (2004). The ENCODE (ENCyclopedia Of DNA Elements) Project. *Science*, **306**(5696), 636–640.

Harbison, C. T., Gordon, D. B., Lee, T. I., Rinaldi, N. J., Macisaac, K. D., Danford, T. W., Hannett, N. M., Tagne, J.-B., Reynolds, D. B., Yoo, J., *et al.* (2004). Transcriptional regulatory code of a eukaryotic genome. *Nature*, **431**(7004), 99–104.

Holloway, D. T., Kon, M., and De Lisi, C. (2005). Integrating genomic data to predict transcription factor binding. *Genome informatics*, **16**(1), 83–94.

Iyer, V. R., Horak, C. E., Scafe, C. S., Botstein, D., Snyder, M., and Brown, P. O. (2001). Genomic binding sites of the yeast cell-cycle transcription factors SBF and MBF. *Nature*, **409**(6819), 533–538.

Kim, T. H., Barrera, L. O., Zheng, M., Qu, C., Singer, M. A., Richmond, T. A., Wu, Y., Green, R. D., and Ren, B. (2005). A high-resolution map of active promoters in the human genome. *Nature*, **436**(7052), 876–880.

Kumar, S. and Bucher, P. (2016). Predicting transcription factor site occupancy using DNA sequence intrinsic and cell-type specific chromatin features. *BMC bioinformatics*, **17**(1), S4.

Kundaje, A., Meuleman, W., Ernst, J., Bilenky, M., Yen, A., Heravi-Moussavi, A., Kheradpour, P., Zhang, Z., Wang, J., Ziller, M. J., and et al. (2015). Integrative analysis of 111 reference human epigenomes. *Nature*, **518**(7539), 317–330.

Lenhard, B., Sandelin, A., Mendoza, L., Engström, P., Jareborg, N., and Wasserman, W. W. (2003). Identification of conserved regulatory elements by comparative genome analysis. *Journal of biology*, **2**(2), 13.

Liu, P., Qiu, X., and Huang, X. (2017). Adversarial multi-task learning for text classification. *arXiv preprint arXiv:1704.05742*.

Luscombe, N. M., Laskowski, R. A., and Thornton, J. M. (2001). Amino acid–base interactions: a three-dimensional analysis of protein–DNA interactions at an atomic level. *Nucleic acids research*, **29**(13), 2860–2874.

Man, T.-K. and Stormo, G. D. (2001). Non-independence of Mnt repressor–operator interaction determined by a new quantitative multiple fluorescence relative affinity (QuMFRA) assay. *Nucleic acids research*, **29**(12), 2471–2478.

Marinescu, V. D., Kohane, I. S., and Riva, A. (2005). The mapper database: a multi-genome catalog of putative transcription factor binding sites. *Nucleic acids research*, **33**(suppl_1), D91–D97.

Mathelier, A. and Wasserman, W. W. (2013a). The next generation of transcription factor binding site prediction. *PLoS computational biology*, **9**(9), e1003214.

Mathelier, A. and Wasserman, W. W. (2013b). The next generation of transcription factor binding site prediction. *PLoS Computational Biology*, **9**(9), e1003214.

Mathelier, A., Xin, B., Chiu, T.-P., Yang, L., Rohs, R., and Wasserman, W. W. (2016a). DNA shape features improve transcription factor binding site predictions in vivo. *Cell systems*, **3**(3), 278–286.

Mathelier, A., Fornes, O., Arenillas, D. J., Chen, C. Y., Denay, G., Lee, J., Shi, W., Shyr, C., Tan, G., and Worsley-Hunt, R. (2016b). JASPAR 2016: a major expansion and update of the open-access database of transcription factor binding profiles. *Nucleic acids research*, **44**(Database issue), D110–D115.

Matys, V., Kel-Margoulis, O. V., Fricke, E., Liebich, I., Land, S., Barre-Dirrie, A., Reuter, I., Chekmenev, D., Krull, M., Hornischer, K., *et al.* (2006). TRANSFAC® and its module TRANSCompel®: transcriptional gene regulation in eukaryotes. *Nucleic acids research*, **34**(suppl_1), D108–D110.

Mohaghegh, N., Bray, D., Keenan, J., Penvose, A., Andrilenas, K. K., Ramlall, V., and Siggers, T. (2019). Nextpbm: a platform to study cell-specific transcription factor binding and cooperativity. *Nucleic Acids Research*, **47**(6), e31.

Quang, D. and Xie, X. (2016). DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. *Nucleic acids research*, **44**(11), e107–e107.

Ren, B., Robert, F., Wyrick, J. J., Aparicio, O., Jennings, E. G., Simon, I., Zeitlinger, J., Schreiber, J., Hannett, N., Kanin, E., *et al.* (2000). Genome-wide location and function of DNA binding proteins. *Science*, **290**(5500), 2306–2309.

Schmidt, F., Gasparoni, N., Gasparoni, G., Gianmoena, K., Cadenas, C., Polansky, J. K., Ebert, P., Nordström, K., Barann, M., and Sinha, A. (2017). Combining transcription factor binding affinities with open-chromatin data for accurate gene expression prediction. *Nucleic acids research*, **45**(1), 54–66.

Sherwood, R. I., Tatsunori, H., OĎonnell, C. W., Sophia, L., Barkal, A. A., John Peter, V. H., Vivek, K., Tommi, J., and Gifford, D. K. (2014). Discovery of directional and nondirectional pioneer transcription factors by modeling dnase profile magnitude and shape. *Nature Biotechnology*, **32**(2), 171–178.

Siddharthan, R. (2010). Dinucleotide weight matrices for predicting transcription factor binding sites: generalizing the position weight matrix. *PloS one*, **5**(3), e9722.

Singh, R., Lanchantin, J., Robins, G., and Qi, Y. (2016). DeepChrome: deep-learning for predicting gene expression from histone modifications. *Bioinformatics*, **32**(17).

Stormo, G. D. (2000). DNA binding sites: representation and discovery. *Bioinformatics*, **16**(1), 16–23.

Stormo, G. D. (2013). Modeling the specificity of protein-DNA interactions. *Quantitative biology*, **1**(2), 115.

Tomovic, A. and Oakeley, E. J. (2007). Position dependencies in transcription factor binding sites. *Bioinformatics*, **23**(8), 933–941.

Tsai, Z. T.-Y., Shiu, S.-H., and Tsai, H.-K. (2015). Contribution of sequence motif, chromatin state, and DNA structure features to predictive models of transcription factor binding in yeast. *PLoS computational biology*, **11**(8), e1004418.

Wang, S., Peng, J., Ma, J., and Xu, J. (2016). Protein secondary structure prediction using deep convolutional neural fields. *Scientific reports*, **6**.

Wasserman, W. W. and Sandelin, A. (2004). Applied bioinformatics for the identification of regulatory elements. *Nature reviews genetics*, **5**(4), 276–287.

Won, K.-J., Ren, B., and Wang, W. (2010). Genome-wide prediction of transcription factor binding sites using an integrated model. *Genome biology*, **11**(1), R7.

Zambelli, F., Pesole, G., and Pavesi, G. (2012). Motif discovery and transcription factor binding sites before and after the next-generation sequencing era. *Briefings in bioinformatics*, **14**(2), 225–237.

Zeng, H., Edwards, M. D., Ge, L., and Gifford, D. K. (2016). Convolutional neural network architectures for predicting DNA-protein binding. *Bioinformatics*, **32**(12), i121–i127.

Zhang, L. Q. and Li, Q. Z. (2017). Estimating the effects of transcription factors binding and histone modifications on gene expression levels in human cells. *Oncotarget*, **8**(25), 40090–40103.

Zhou, J. and Troyanskaya, O. G. (2015). Predicting effects of noncoding variants with deep learning-based sequence model. *Nature methods*, **12**(10), 931–934.

Zhou, Q. and Liu, J. S. (2004). Modeling within-motif dependence for transcription factor binding site predictions. *Bioinformatics*, **20**(6), 909–916.

Zhou, T., Yang, L., Lu, Y., Dror, I., Dantas Machado, A., Ghane, T., Di Felice, R., and Rohs, R. (2013). DNAshape: a method for the high-throughput prediction of dna structural features on a genomic scale. *Nucleic Acids Research*, **41**, W56âŁ“W62.

Zhou, T., Shen, N., Yang, L., Abe, N., Horton, J., Mann, R. S., Bussemaker, H. J., Gordan, R., and Rohs, R. (2015). Quantitative modeling of transcription factor binding specificities using DNA shape. *Proceedings of the national academy of sciences*, **112**(15), 4654–4659.