Genomics Proteomics Bioinformatics 20 (2022) 496–507

Genomics Proteomics Bioinformatics

www.elsevier.com/locate/gpb
www.sciencedirect.com

METHOD

DeepCAGE: Incorporating Transcription Factors in Genome-wide Prediction of Chromatin Accessibility

Qiao Liu [1,2], Kui Hua [1], Xuegong Zhang [1], Wing Hung Wong [2,*], Rui Jiang [1,*]

[1] *Ministry of Education Key Laboratory of Bioinformatics; Bioinformatics Division, Beijing National Research Center for Information Science and Technology; Center for Synthetic and Systems Biology, Department of Automation, Tsinghua University, Beijing 100084, China*
[2] *Department of Statistics, Stanford University, Stanford, CA 94305, USA*

Received 31 January 2021; revised 31 May 2021; accepted 27 September 2021
Available online 12 March 2022

Handled by Zhihua Zhang

KEYWORDS

Chromatin accessibility;
Deep learning;
Transcription factor;
Gene expression

Abstract Although computational approaches have been complementing high-throughput biological experiments for the identification of functional regions in the human genome, it remains a great challenge to systematically decipher interactions between **transcription factors** (TFs) and regulatory elements to achieve interpretable annotations of **chromatin accessibility** across diverse cellular contexts. To solve this problem, we propose DeepCAGE, a **deep learning** framework that integrates sequence information and binding statuses of TFs, for the accurate prediction of chromatin accessible regions at a genome-wide scale in a variety of cell types. DeepCAGE takes advantage of a densely connected deep convolutional neural network architecture to automatically learn sequence signatures of known chromatin accessible regions and then incorporates such features with expression levels and binding activities of human core TFs to predict novel chromatin accessible regions. In a series of systematic comparisons with existing methods, DeepCAGE exhibits superior performance in not only the classification but also the regression of chromatin accessibility signals. In a detailed analysis of TF activities, DeepCAGE successfully extracts novel binding motifs and measures the contribution of a TF to the regulation with respect to a specific locus in a certain cell type. When applied to whole-genome sequencing data analysis, our method successfully prioritizes putative deleterious variants underlying a human complex trait and thus provides insights into the understanding of disease-associated genetic variants. DeepCAGE can be downloaded from https://github.com/kimmo1019/DeepCAGE.

* Corresponding authors.
E-mail: whwong@stanford.edu (Wong WH), ruijiang@tsinghua.edu.cn (Jiang R).

Peer review under responsibility of Beijing Institute of Genomics, Chinese Academy of Sciences / China National Center for Bioinformation and Genetics Society of China.

https://doi.org/10.1016/j.gpb.2021.08.015

Introduction

One of the fundamental questions in functional genomics is how activities of genes are spatially and temporally controlled through interactive effects of transcription factors (TFs) and regulatory elements such as promoters, enhancers, and silencers. These regulatory elements, as short regions of non-coding DNA sequence, are known to typically reside in chromatin accessible regions and be bound by a set of TFs to carry out regulatory functions in a manner specific to cellular contexts [1]. Therefore, the exploration of a landscape of chromatin accessible regions across major cell types will greatly facilitate the deciphering of gene regulatory mechanisms and further provide insights into cell differentiation, tissue homeostasis, and disease development [2].

Recent advances in deep sequencing techniques have enabled genome-wide assays of chromatin accessibility. For example, DNase-seq utilizes the DNase I enzyme to digest DNA sequences and identify DNase I-hypersensitive regions that are largely chromatin accessible [3]. ATAC-seq uses the Tn5 transposase to integrate primer DNA sequences into cleaved fragments that mainly come from chromatin accessible regions [4]. With the accomplishment of the ENCODE [5] and Roadmap [6] projects, these techniques have been successfully applied to the establishment of the chromatin accessibility landscape for dozens of cell lines across several species. The accumulation of these data provides an unprecedented opportunity for deepening our understanding of both gene regulation and occurrence of diseases [7–9].

However, due to limitations such as experimental cost, it is still impractical to further extend the landscape to cover all possible cell types, with the consideration of the huge variability in cellular biological contexts such as cell differentiation, environmental stimuli, and other factors. Toward this concern, computational approaches have been proposed to predict chromatin states by using such information as DNA sequence, gene expression, and other types of data [10–19]. For example, Kelley et al. proposed a deep convolutional neural network model called Basset to predict chromatin accessible regions purely relying on one-hot encoded DNA sequences [12]. Liu et al. developed a hybrid deep learning model for integrating multiple forms of sequence representations to achieve high prediction performance [14]. Quang et al. used a hybrid convolutional and recurrent neural network for predicting chromatin signals [18]. However, a model purely relying on sequence data can hardly be generalized to make predictions across different cell types as the sequence itself is not cell type-specific. To overcome this limitation, Zhou et al. proposed a regression model called BIRD that utilized only gene expression data to predict chromatin accessible regions [13]. Nevertheless, with the complete removal of sequence data, the scope of application of this method is limited because the availability of gene expression is not as wide as sequence data. With the aforementioned understanding, Nair et al. proposed a deep residual neural network [20] model called ChromDragoNN to combine both sequence and expression data toward the prediction of chromatin accessibility [21]. However, sequence signatures and expression features are combined by simple concatenation in this method. This formulation, though simple in computation, lacks enough interpretability and is not consistent with existing biological knowledge.

With the aforementioned understanding, we propose a method called DeepCAGE, that is, a Deep densely connected convolutional network for predicting Chromatin Accessibility by incorporating Gene Expression and binding statuses of TFs. Unlike BIRD and ChromDragoNN that take full expression data as predictors, our method carefully considers the binding statuses of chromatin-binding factors (*e.g.*, TFs), based on the biological understanding that chromatin accessibility is largely determined by chromatin-binding factors that have access to DNA [2]. In a series of systematic evaluations, DeepCAGE achieves state-of-the-art performance in not only the classification of chromatin accessible statuses but also the regression of DNase-seq signals. To make DeepCAGE more understandable, we propose a strategy for visualizing the weights in the first convolutional layer. Interestingly, many known motifs were successfully recovered by DeepCAGE. In the downstream application to whole-genome sequencing (WGS) data analysis, DeepCAGE effectively prioritizes deleterious variants for the prediction and interpretation of complex phenotypes.

Method

Overview of DeepCAGE

DeepCAGE was designed based on the premise that binding statuses and gene expression of TFs could complement sequence data toward the precise prediction of chromatin accessibility. With this understanding, we designed DeepCAGE as a hybrid neural network that consisted of a convolutional module for sequence data and a feedforward module for chromatin accessibility prediction (**Figure 1**). Briefly, we applied the one-hot encoding to the input sequence data, fed the encoded data to a densely connected convolutional neural network (DenseNet), and took the output as the sequence feature. For binding statuses, we scanned the input sequence for potential binding sites for a set of 402 human TFs by using non-redundant motifs in the HOCOMOCO database [22] with the tool Homer [23]. We then selected the maximum score of reported binding sites for each TF to obtain a vector of 402 dimensions as the motif feature. For gene expression, we focused on log-transformed transcripts per million (TPM) values of the 402 TFs and obtained a vector of 402 dimensions after quantile normalization as the expression feature. With these data, we combined the two vectors of the motif and expression features by taking the element-wise product, and we concatenated the result to the sequence feature to obtain the hybrid feature, which went through a feedforward neural network with a fully connected hidden layer and an output layer for either classification or regression. We presented detailed hyperparameters of the hybrid network in Table S1.

DeepCAGE extracts sequence features by using an architecture called the DenseNet, which has the advantage of alleviating the vanishing-gradient problem and strengthening the feature propagation [24]. As shown in Figure 1, there are three dense blocks in our model. Each block includes five convolutional layers, and each layer connects to every other layer in a feedforward fashion. A convolutional layer consists of two consecutive small kernels of size 1×1 and 3×1, where the former aims at reducing the concatenated channels to a fixed number, and the latter acts as the traditional convolution. A

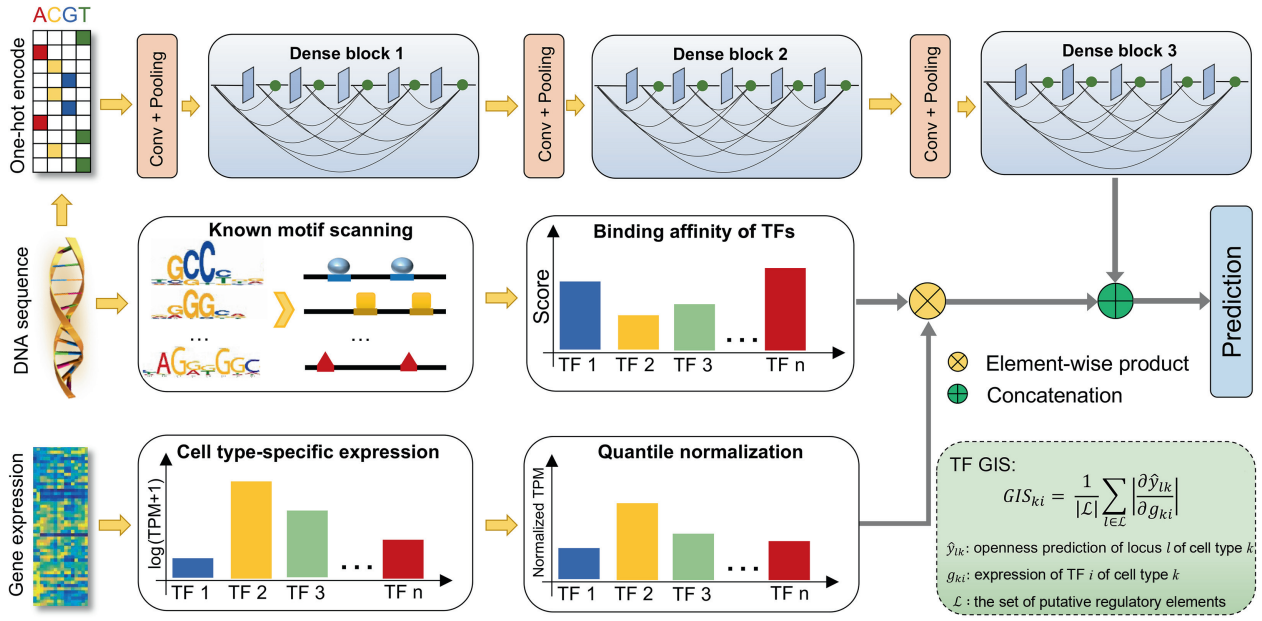


Figure 1 Overview of the DeepCAGE model

The sequence of the input DNA region is converted to a one-hot matrix and goes through a DenseNet to extract sequence features. Normalized expression levels of the 402 human TFs and the corresponding motif binding scores are combined by using an element-wise product and then concatenated with sequence features. The combined features are finally fed to a feedforward neural network for chromatin accessibility prediction. DenseNet, densely connected convolutional neural network; TPM, transcripts per million; Conv, convolution; GIS, gradient importance score; TF, transcription factor.

transition module is presented before a dense block for feature extracting and dimensionality reduction. An input sequence is first extended to a fixed length of 1000 bp centered at the midpoint of the sequence and then converted to a 1000×4 binary matrix by using the one-hot encoding. The matrix is then fed to the first transition module that contains a convolutional layer and a max-pooling layer. The convolutional layer has 160 kernels of size 4×15 for extracting low-level features and detecting DNA binding motifs, while the max-pooling layer is present for finding the most significant activation signal in a given sliding window of each kernel. Similar settings are used for the other two transition modules for extracting high-level features and dimensionality reduction. Rectified linear units (ReLU) are used after each convolution operation for keeping positive activations and setting negative activation values to zeros. Batch normalization [25] and dropout [26] strategies are used after each ReLU function for reducing internal covariate shift and avoiding overfitting, respectively. For the DeepCAGE regression model, there are two major differences from the classification model. First, the output layer directly uses a linear transformation instead of a sigmoid function. Second, the mean square error (MSE) instead of the cross-entropy is used as the loss function.

Data processing

DNase-seq bam files and narrow peaks across 55 human cell types were downloaded from the ENCODE project [5] (Tables S2 and S3). The human hg19 reference genome was divided into non-overlapping regions (loci) of 200 bp. Considering that a cell type may have multiple DNase-seq replicates, a locus is regarded as chromatin accessible if it overlaps with narrow peak regions of at least half of the replicates and inaccessible otherwise (Figure S1). For the classification design, a binary label y_{lk} is assigned to locus l, representing whether it is accessible in cell type k. For the regression design, bam files of multiple replicates for a cell type are pooled, and the raw read counts, n_{lk}, is obtained for locus l in cell type k. To eliminate the effect of sequencing depths, the normalized read count, $\widetilde{n}_{lk} = Nn_{lk}/N_k$, is calculated, where N_k denotes the total number of pooled reads for cell type k, and $N = \min\{N_k\}$ is the minimal number of pooled reads across all cell types. The normalized read counts are further log-transformed after adding a pseudocount of one. The transformed data represent the level of chromatin accessibility and are then used as the response variable in the regression model.

RNA-seq data across the same 55 human cell types were also downloaded from the ENCODE project (Table S4). TPM of the 402 core human TFs were extracted from the gene expression data. After further log transformation and quantile normalization based on TPM values, the normalized expression within each cell type was averaged across multiple replicates, and the mean expression profile of each cell type was finally used.

WGS data and RNA-seq profiles of Genotype-Tissue Expression (GTEx) muscle tissues were downloaded from the Database of Genotypes and Phenotypes (dbGaP: phs000424.v7.p2). Matching these two types of data, a total of 491 donors were selected for downstream analysis (Table S5). For each of these donors, RNA-seq data were processed in the same way as ENCODE data, and WGS data were filtered by excluding all insertions/deletions (indels) and rare variants whose minor allele frequencies were less than or equal to 5 across all donors.

Model evaluation

Cell type-level five-fold cross-validation experiments are designed for evaluating our method. In each fold, the 55 cell types are partitioned into a training set with 44 cell types and a testing set with the remaining 11 cell types (Tables S6 and S7). Putative known accessible loci are identified as genomic regions (loci) that are chromatin accessible in at least two cell types in the training set. Putative novel accessible loci are identified as genomic regions that are accessible in at least two testing cell types and are not present in the training data.

Cell type-wise and locus-wise metrics are defined to evaluate our method from different perspectives (Figure S2). Cell type-wise metrics are calculated within a testing cell type across genomic regions to provide high-level assessment of a method. Locus-wise metrics are calculated based on a genomic region across cell types to give a detailed analysis of the performance. These metrics provide a comprehensive and systematic evaluation of our method in both the classification and the regression designs.

Let $\boldsymbol{Y}_{L\times K}$ and $\widehat{\boldsymbol{Y}}_{L\times K}$ be the true label matrix and predicted matrix, where L denotes the number of putative loci and K denotes the number of cell types. In the classification design, y_{lk} and $\widehat{y}_{lk}$ denote the true binary label and predicted probability of chromatin accessible status for locus l in cell type k, respectively. In this situation, the cell type-wise area under the precision-recall curve (auPR) for cell type k is calculated based on $\boldsymbol{y}_{*k} = (y_{1k}, y_{2k}, \cdots, y_{Lk})$ and $\widehat{\boldsymbol{y}}_{*k} = (\widehat{y}_{1k}, \widehat{y}_{2k}, \cdots, \widehat{y}_{Lk})$ as follows. Given a threshold t for a cell type k, the precision is defined as the number of correct predictions ($\sum_l y_{lk} I(\widehat{y}_{lk} > t)$) over the number of all predictions ($\sum_l I(\widehat{y}_{lk} > t)$), and the recall is defined as the number of correct predictions over the number of truly accessible loci ($\sum_l y_{lk}$), where $I(x)$ is an indicator function that is equal to 1 if x is true and 0 otherwise. Varying the threshold from 0 to 1 and calculating the precision and recall at each threshold value, the precision-recall curve can be drawn, and the area under this curve can be obtained. The locus-wise auPR for locus l is calculated based on $\boldsymbol{y}_{l*} = (y_{l1}, y_{l2}, \cdots, y_{lK})$ and $\widehat{\boldsymbol{y}}_{l*} = (\widehat{y}_{l1}, \widehat{y}_{l2}, \cdots, \widehat{y}_{lK})$ in a similar way.

In the regression design, y_{lk} and $\widehat{y}_{lk}$ denote the true and predicted DNase-seq signals for locus l in cell type k, respectively. In this situation, the cell type-wise Pearson correlation coefficient (PCC) for cell type k is calculated as the PCC of $\boldsymbol{y}_{*k}$ and $\widehat{\boldsymbol{y}}_{*k}$, and the locus-wise PCC is calculated based on $\boldsymbol{y}_{l*}$ and $\widehat{\boldsymbol{y}}_{l*}$ in a similar way. The prediction squared error (PSE), which considers both cell type-wise prediction and locus-wise prediction, is calculated as $\mathrm{PSE} = \sum_k \sum_l (y_{lk} - \widehat{y}_{lk})^2 / \sum_k \sum_l (y_{lk} - \bar{y}_{*k})^2$, where $\bar{y}_{*k} = \sum_l y_{lk} / L$ is the mean of $\boldsymbol{y}_{*k}$.

Two statistics, cell range and cell variability, are introduced to describe the activity of a locus based on the true DNase-seq signals across testing cell types. The cell range of locus l is calculated by $\max(\boldsymbol{y}_{l*}) - \min(\boldsymbol{y}_{l*})$, and the cell variability of locus l is defined by the standard deviation of $\boldsymbol{y}_{l*}$.

Baseline methods

Basset [12], DeepSEA [10], and DanQ [18] are three representative neural network models that take only DNA sequences as input. BIRD [13] is a regression model that takes only gene expression data as input. ChromDragoNN [21] is a neural network-based model that takes both DNA sequences and gene expression data as input. Our method and ChromDragoNN have the following major differences. First, the design principles of these two methods are notably different. ChromDragoNN predicts chromatin accessibility through directly concatenating DNA sequences and expression data of all genes. DeepCAGE explains chromatin accessibility with DNA sequences and binding statuses of TFs. Therefore, DeepCAGE tries to interpret chromatin accessibility in a more natural way since chromatin accessibility is believed to be largely determined by the occupancy and topological organization of nucleosomes as well as chromatin-binding factors [2]. Second, the network architectures of these two methods are different. ChromDragoNN uses a ResNet to extract sequence features, while DeepCAGE uses a DenseNet that is a relatively new architecture and has also been experimentally validated to outperform ResNet in many tasks [20]. Third, inputs of these two methods are also different. ChromDragoNN requires DNA sequences and expression data of all genes, while DeepCAGE takes DNA sequences and expression data of 402 human core TFs as input. Motif binding profiles of these TFs can be annotated with the existing motif database, which can be precomputed without additional experimental cost.

Gradient importance score

DeepCAGE takes advantage of the gradient importance score (GIS) to prioritize TFs given a pair of cell types and a genomic locus. Briefly, a locus is extended to a 200 kb genomic region centered at the midpoint of the locus. Then, the average absolute gradient of predicted accessibility within the extended region with respect to the expression of a TF is calculated as:

$$GIS_{ki} = \frac{1}{|\mathscr{L}|} \sum_{l \in \mathscr{L}} \left| \frac{\partial \hat{y}_{lk}}{\partial g_{ki}} \right|$$

where $\widehat{y}_{lk}$ denotes the predicted accessibility of locus l in cell type k, g_{ki} denotes the expression of TF i in cell type k, and $\mathscr{L}$ denotes the set of putative regulatory elements that contains all accessible loci within the extended region. The GIS gives an intuition of which TFs play an important role in a specific cell type.

Motif analysis

The weights of the kernels from the first convolutional layer are converted into position weight matrices (PWMs) by counting subsequence occurrences in a set of input sequences that activate a kernel at a threshold value. All subsequences with activation values that greater than the threshold of a kernel are pooled together and aligned. The PWMs are then composed of the frequencies of the four nucleotides (A, C, G, and T) at each position. A subsequence at position i is regarded as activated if

$$\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} w_{m,n}^{k} x_{i+m,n}^{j} > \alpha \cdot MAV^{k}$$

where $M \times N$ denotes the size of the kernels (4×15 in the first convolutional layer), and α is the control coefficient with the default value of 0.7 in all experiments. MAV^k denotes the maximal activation value of kernel k and is represented as:

$$MAV^k = \max_{i,j}\left(\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} w^k_{m,n} x^j_{i+m,n}\right)$$

Motifs are identified using the tool TomTom (v4.12.0) [27] with the *E*-value threshold of 0.05 and are compared to known motifs in the JASPAR database (v2018) [28]. Besides, the information content of recovered motifs is calculated based on the information entropy, as $IC = \sum_{i,j}\left(p_{ij}\log_2 p_{ij} - b_i \log_2 b_i\right)$, where p_{ij} is the element in PWM, i and j are the nucleotide type and position, respectively, and b_i (default value: 0.25) is the background frequency of nucleotide i.

Phenotype prediction

A linear regression model with l_1 penalty is adopted to predict the heights of GTEx donors using the deleterious scores of variants, as:

$$h = \alpha_0 + \sum_{k=1}^{K} \alpha_k \Delta O_k$$

where h is the height of a GTEx donor, and ΔO_k denotes the deleterious score of variant k calculated using DeepCAGE. The coefficient of the l_1 penalty is set to 0.5. A ten-fold cross-validation experiment is used in validation, and the average coefficient of determinant (R^2) is used for evaluating how much variance in the phenotype can be explained.

Results

DeepCAGE accurately predicts binary chromatin accessibility statuses

We first evaluated the performance of DeepCAGE in predicting whether an input DNA sequence is chromatin accessible or not. To achieve this objective, we downloaded paired DNase-seq and RNA-seq data across 55 cell types from the ENCODE project [5] and conducted a five-fold cross-validation experiment at the cell type level. In each fold of the validation, we partitioned the data into a training set of 44 cell types and a testing set of the remaining 11 cell types. We then defined putative known accessible loci as genomic regions that are chromatin accessible in at least two cell types in the training data. For each cell type, we further identified a positive set of putative loci that are accessible in the cell type and a negative set of putative loci that are inaccessible. After that, we trained our model on the training data and classified positive loci against negative ones for each testing cell type. Finally, we calculated a criterion called the cell type-wise auPR (see Method) to evaluate the performance of a classification method.

We compared the performance of DeepCAGE with four existing methods, including Basset [12], DeepSEA [10], DanQ [18], and ChromDragoNN [21] in the aforementioned cross-validation experiment. Results (**Figure 2**A) show that DeepCAGE achieves the highest performance with the mean cell type-wise auPR of 0.418 for known accessible loci, compared to 0.166 of Basset, 0.195 of DeepSEA, 0.188 of DanQ, and 0.319 of ChromDragoNN. Particularly, DeepCAGE outperforms sequence-based methods by a large margin, suggesting that these methods may fail in capturing cell type-specific information. Further analysis shows that the proportion of positive loci is in general small in a cell type and exhibits large variation (ranging from 2.6% to 29%), suggesting the ability of our method in dealing with unbalanced data.

We then took one step further to assess the ability of our method in predicting novel chromatin accessible loci. In each fold of the validation experiment, we identified putative novel accessible loci as genomic regions that are accessible in at least two testing cell types and are not present in the training data, and we applied the trained model to predict whether these loci are accessible or not in a testing cell type. Results, as shown in Figure 2A, also suggest the superiority of DeepCAGE with a mean cell type-wise auPR of 0.181, compared to 0.107 of Basset, 0.104 of DeepSEA, 0.110 of DanQ, and 0.151 of ChromDragoNN.

We finally analyzed how the cell type specificity of accessible regions affects the prediction performance of our method. To achieve this objective, we divided the putative known accessible loci into three groups based on the proportion of cell types in which a locus is accessible. We then evaluated the cross-validation results using a criterion called the locus-wise auPR that evaluated the prediction performance of a method on an accessible locus across cell types (see Method). Results show that for a locus accessible in less than 10% cell types, DeepCAGE achieves a mean locus-wise auPR of 0.578, and this criterion increases when a locus is accessible in more cell types (Figure 2B). These results suggest that the cell type specificity is likely a factor that affects the prediction performance of a method.

DeepCAGE recovers a continuous degree of chromatin accessibility

In the aforementioned classification experiments, we only considered the binary accessible status of a genomic region in a specific cell type. In the real situation, however, the accessibility of a genomic region given by a DNase-seq experiment is in a continuous form. Considering this situation, we further proposed a DeepCAGE regression model to predict the degree of chromatin accessibility for a DNA region , which is defined as the normalized average count of raw reads that fall into the corresponding region.

With the same cross-validation settings as in the aforementioned section, we compared the performance of DeepCAGE to two baseline methods, BIRD [13] and ChromDragoNN [21], and we assessed regression results in terms of two criteria, the cell type-wise PCC and PSE (**Figure 3**A–C; see Method). Results show that DeepCAGE achieves a mean cell type-wise PCC of 0.785, compared to 0.637 for BIRD and 0.735 for ChromDragoNN (Figure 3B). Further analysis shows that in 18.2% of the testing cell types, DeepCAGE achieves a cell type-wise PCC of 0.85 or higher. In two cell types, DeepCAGE even achieves a cell type-wise PCC of 0.9 or higher (see examples in Figure 3A). DeepCAGE also achieves the minimal PSE (0.42), outperforming the two baseline methods (0.77 for BIRD and 0.57 for ChromDragoNN) by a quite large margin (Figure 3C).

We then explored the performance of DeepCAGE for putative accessible loci with different cell type specificity by introducing two statistics, cell range and cell variability, to describe the activity dynamics of a genomic region based on

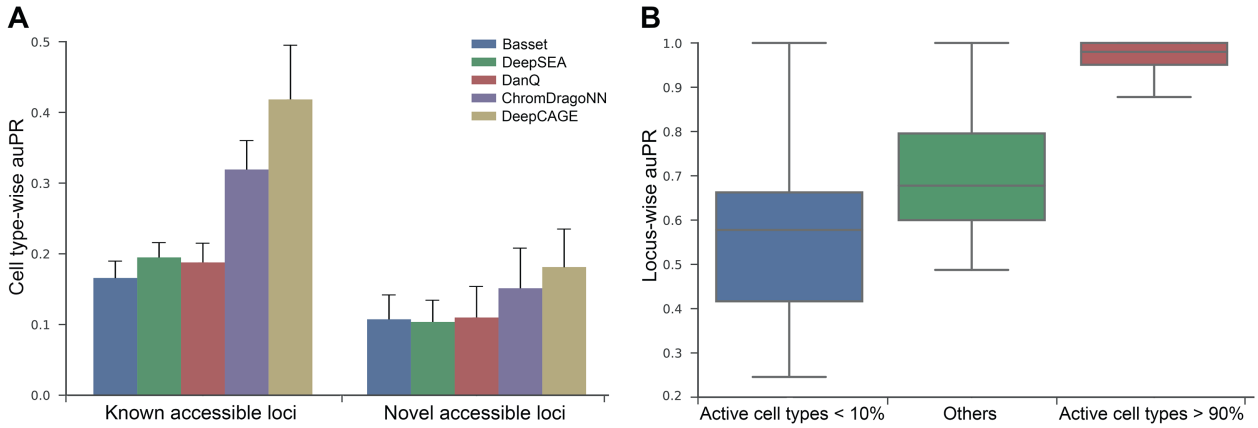


Figure 2 Performance of the DeepCAGE classification model

A. DeepCAGE achieves the highest cell type-wise auPR for both known accessible loci and novel accessible loci compared to baseline methods (Basset, DeepSEA, DanQ, and ChromDragoNN). **B.** The performance of DeepCAGE for loci with different activities across testing cell types. auPR, area under the precision-recall curve.

the true DNase-seq signals cross cell types (see Method). We divided known and novel accessible loci into three groups (low, medium, and high) according to the 1/3 and 2/3 quantiles of these statistics. Results show that DeepCAGE has high performance for accessible loci with medium cell range and cell variability (Figure 3D), consistent with the results in BIRD [13]. Briefly, DeepCAGE achieves a median locus-wise PCC (see Method) of 0.512 for known accessible loci with medium cell range, compared to 0.435 and 0.399 for loci with low and high cell ranges, respectively. When using the statistic of cell variability, DeepCAGE achieves median locus-wise PCCs of 0.384, 0.514, and 0.448 for known accessible loci with low, medium, and high cell variabilities, respectively. The results are similar for novel accessible loci, except that the values of the criteria are slightly low. We further divided known accessible loci into five groups based on the number of cell types in which a locus is accessible. Results (Figure 3E) show that the performance of DeepCAGE varies a lot for loci accessible in different numbers of cell types. Briefly, the performance is high for loci accessible in the medium proportion of cell types and low for those accessible in only a small proportion of cell types.

Finally, we visualized both the true (green) and predicted (yellow) DNase-seq signals of a sample genomic region across three testing cell types (GM12878, HepG2, and H1-hESC) in the UCSC genome browser [29]. In addition, we also provided the mean signal (red; calculated by taking the average DNase-seq signals across all training cell types) as a reference. As shown in Figure 3F, obviously, DeepCAGE well distinguishes the difference of DNase-seq signals among the three testing cell types while the mean signal fails.

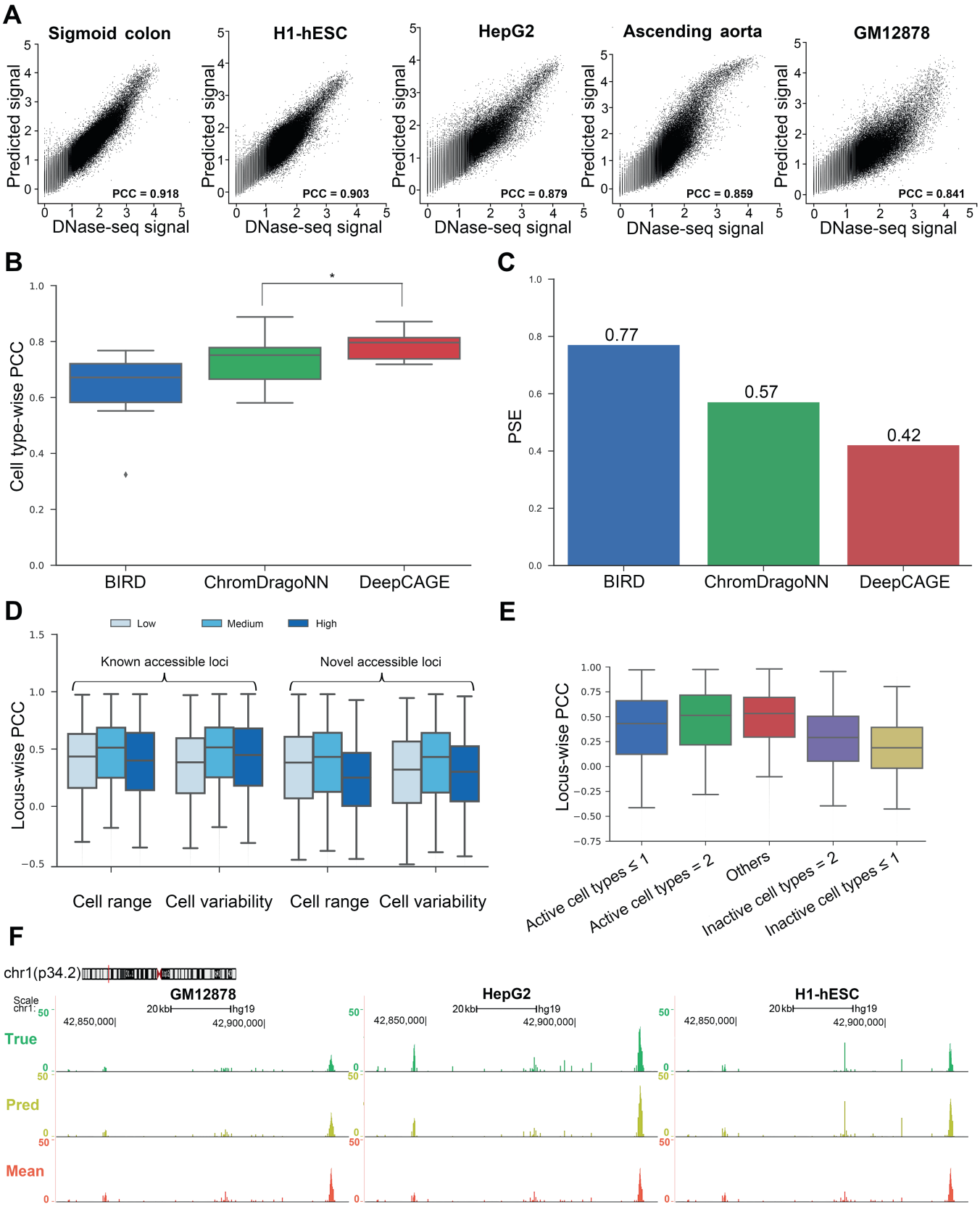
Model ablation analysis of DeepCAGE

We studied the contributions of gene expression and binding scores of TFs to the performance of our method. Taking the DeepCAGE regression model as an example, by discarding gene expression data, the median cell type-wise PCC decreased by 13.1% (Figure S3; $P = 6.53 \times 10^{-11}$, one-sided paired-sample Wilcoxon signed-rank test). By removing binding scores, the median cell type-wise PCC decreased by 3.6% (Figure S3; $P = 3.78 \times 10^{-4}$, one-sided paired-sample Wilcoxon signed-rank test). These results suggest that gene expression data could significantly help improve the performance of DeepCAGE in cross-cell type prediction, while binding scores slightly increase the performance. One potential reason behind this observation is that a large proportion of DNA sequence motifs have already been learned in the convolution layers of the neural network, and thus the binding scores only provide complementary information regarding DNA sequence features.

Besides, to demonstrate the superiority of the network architecture used by DeepCAGE, we additionally conducted the following two experiments. First, we replaced the DenseNet with a ResNet which had the same number of layers as the number of dense blocks and the same hidden nodes in the convolutional layers. Results show that DenseNet leads to 6.4% increment in performance over ResNet in terms of the median cell type-wise PCC (Figure S4; $P = 3.15 \times 10^{-6}$, one-sided paired-sample Wilcoxon signed-rank test). Second, we explored the influence of two key hyperparameters (the number of residual blocks and the convolutional layers within a residual block) on the performance of ChromDragoNN. It is noted that a deeper model architecture does not help improve the performance significantly (Figure S5).

GIS helps prioritize cell type-related TFs

We proposed a strategy for prioritizing cell type-related TFs according to the absolute gradient of the predicted accessibility with respect to the expression of a TF. Taking the K562 cell line as an example, we calculated the average GISs of all TFs from all putative loci within up-streaming 100 kb to down-streaming 100 kb of a tumor suppressor gene *TP53*, which has been shown to have a key role in myeloid blast transformation [30]. The average GISs of all TFs across cell types with respect to the transcription start site (TSS) of this gene are shown in **Figure 4**A. The 402 human core TFs were then prioritized by their average GISs in K562 cell line (Figure 4B). Interestingly, many top-ranked TFs were related to functions in leukemia cells validated by literature. For example, *EGR1* (rank^{1st}) was involved in regulating PMA-induced megakaryocytic differentiation of K562 cell line [31];


A
Sigmoid colon
H1-hESC
HepG2
Ascending aorta
GM12878
Predicted signal
DNase-seq signal
PCC = 0.918
PCC = 0.903
PCC = 0.879
PCC = 0.859
PCC = 0.841
B
Cell type-wise PCC
BIRD
ChromDragoNN
DeepCAGE
C
PSE
0.77
0.57
0.42
D
Low
Medium
High
Known accessible loci
Novel accessible loci
Locus-wise PCC
Cell range
Cell variability
E
Locus-wise PCC
Active cell types ≤ 1
Active cell types = 2
Others
Inactive cell types = 2
Inactive cell types ≤ 1
F
chr1(p34.2)
GM12878
HepG2
H1-hESC
True
Pred
Mean

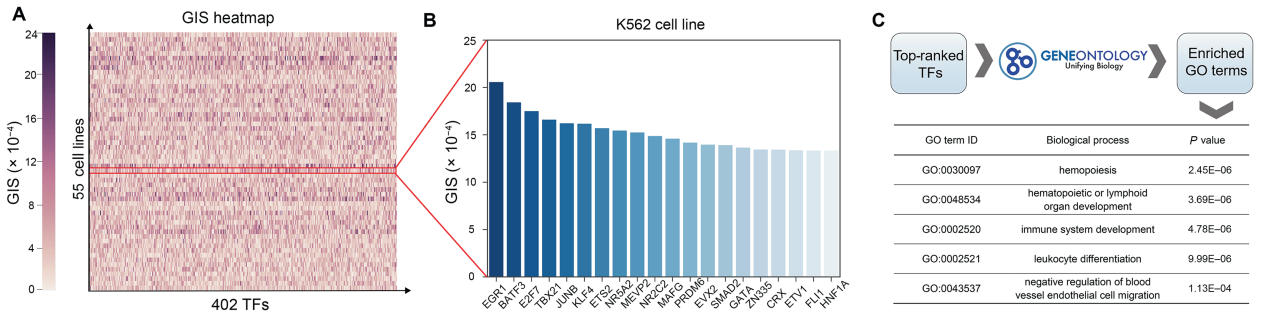


Figure 4 GIS helps identify important TFs

A. GIS heatmap of the 402 human core TFs across 55 cell types. **B.** Bar chart showing the GISs of the 20 top-ranked TFs in the K562 cell line. **C.** Enriched GO terms by top-ranked TFs in the K562 cell line. GO, Gene Ontology.

the inhibition of *E2F7* (rank^{3rd}) might lead to a reduction of miRNAs involved in leukemic cell lines [32]; the expression of *JunB* (rank^{5th}) was inactivated by methylation in chronic myeloid leukemia [33]. The Gene Ontology (GO) terms enriched by the top 5% prioritized TF coding genes also included biological processes of leukocyte differentiation and hematopoietic development (Figure 4C). To sum up, the GIS gives us an intuitive interpretation of which TF may play an important role in predicting chromatin accessibility given a specific cell type and a genomic region.

DeepCAGE automatically learns binding motifs of TFs

In order to make DeepCAGE more understandable, we explored the features that were automatically learned by DeepCAGE by investigating the weights of the 160 kernels in the first convolutional layer. Briefly, we converted the weights into PWMs (see Method) and then compared them with known motifs in the JASPAR database [28]. We found that 48 (30%) of the kernels could match known motifs at the *E*-value threshold of 0.05. Among the matched kernels, 25 (52%) had at least one matched core human TF used in DeepCAGE model. We then calculated the information content (see Method), set the weights of each kernel to zeros, and denoted the decrease in the cell type-wise PCC as the influence score for each kernel. We showed several learned unmatched motifs that have a high influence score (**Figure 5**A) and illustrated a few examples of learned motifs that could match known motifs in JASPAR database (Figure 5B). These results demonstrate that DeepCAGE can not only help us find potential binding motifs but also has the potential to guide the finding of novel motifs which are not discovered by experiments yet.

DeepCAGE prioritizes putative deleterious variants in personal genomes

We applied DeepCAGE to WGS data analysis and demonstrated how our method could benefit the detection of individual-specific deleterious variants in regulatory elements that potentially influence phenotype. The principle was to quantify the degree that a genetic variant affects the chromatin accessibility of a nearby genomic region and then prioritize variants accordingly. As shown in **Figure 6**A, for an individual, we fed the individual genome and the reference genome separately to the trained DeepCAGE regression model and calculated prediction scores for each of them. We then took the absolute $\log_2$ fold change of these two scores as a measure of the change in chromatin accessibility. For a variant, we defined its individual-level deleterious score by the change of chromatin accessibility of a 200 bp genomic region around. Finally, we obtained the cohort-level deleterious score for a variant by applying the aforementioned procedure to all individuals in a cohort who contain the variant and then averaging the individual-level deleterious scores for the variant. Note that we also took as input the expression profile of TFs in the muscle tissue and only considered WGS variants with the minor allele frequency larger than 5.

We downloaded WGS data of 491 donors with the height phenotype from the dbGap of the GTEx project (Table S5). We collected 3290 risk single nucleotide polymorphisms (SNPs) that were associated with height by a large-scale

Figure 3 Performance of the DeepCAGE regression model

A. DeepCAGE predicts DNase-seq signals in five testing cell types. **B.** Cell type-wise PCC for three different methods across all testing cell types. *, two-sided paired-sample Wilcoxon signed-rank test *P* value = 3.37×10^{-5}. **C.** PSE for three different methods across all testing cell types. **D.** Locus-wise PCC achieved by DeepCAGE with respect to two statistics with both known accessible loci and novel accessible loci. **E.** Locus-wise PCC achieved by DeepCAGE considering the number of accessible cell types under known accessible loci. **F.** An example of true (green) and predicted (yellow) DNase-seq signals of three testing cell types under the same genomic region (Chr1:42.83–42.93 Mb). Mean signal (red) denotes the average DNase-seq signal across all training cell types. PCC, Pearson correlation coefficient; PSE, prediction squared error.

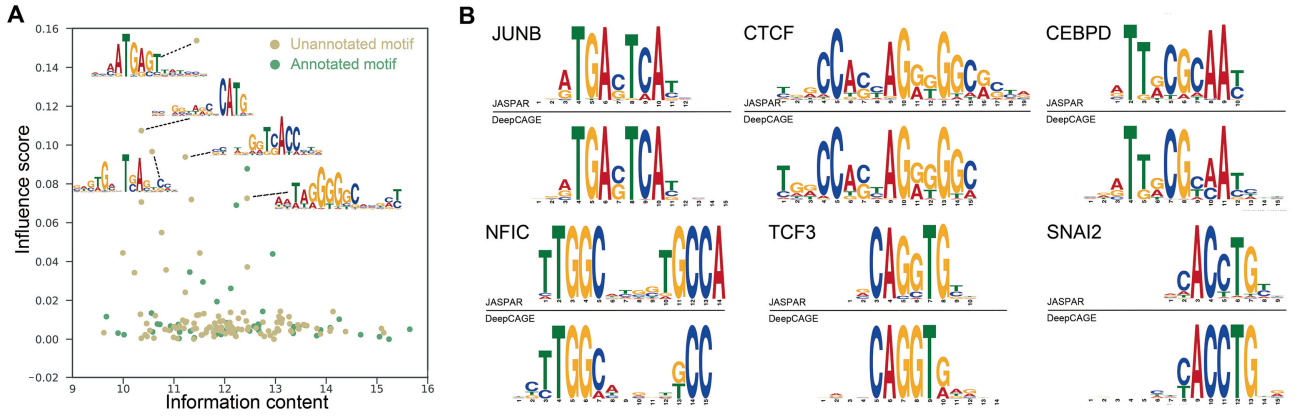


Figure 5 DeepCAGE recovers both known and novel motifs

A. DeepCAGE identifies both known and novel motifs in the learning process. Green dots and yellow dots represent known and novel motifs recovered by DeepCAGE, respectively. **B.** Matched motifs with an *E*-value threshold of 0.05 in the format of sequence logos (above: known motif from the JASPAR database; below: motif learned by DeepCAGE).

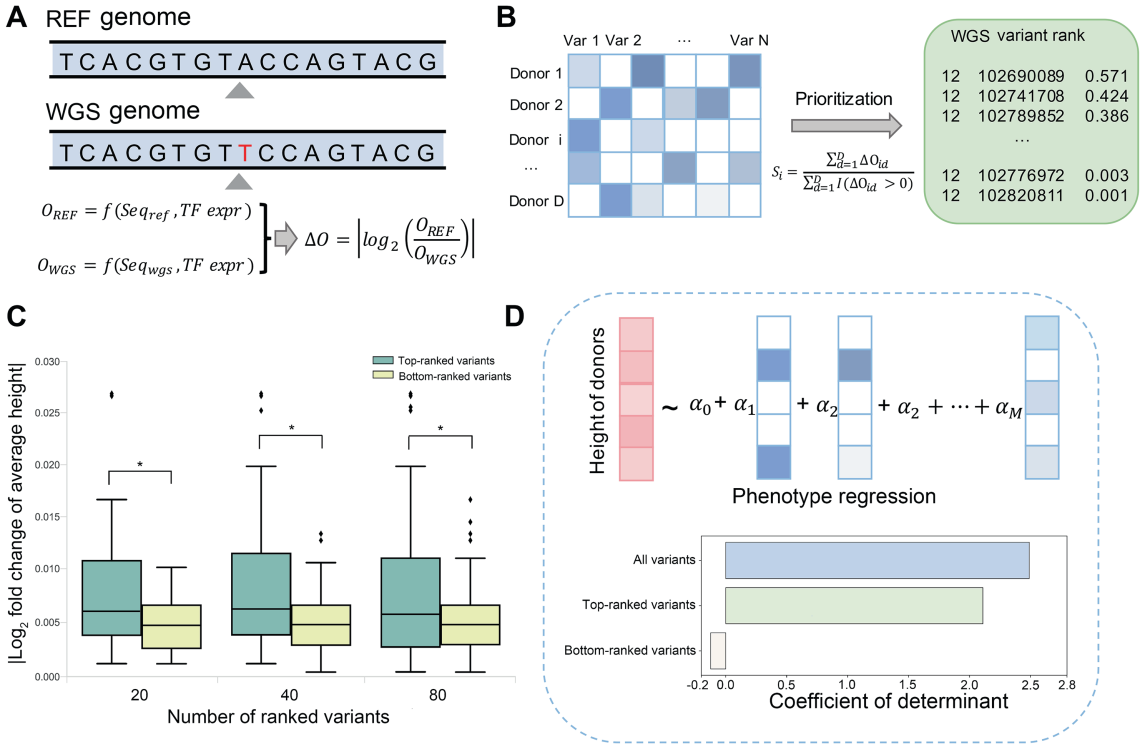


Figure 6 DeepCAGE helps prioritize and interpret WGS variants

A. The deleterious score is calculated by the absolute value of $\log_2$ fold change of predicted chromatin accessibility of the REF genome and the personal genome from WGS data. **B.** WGS variants within a risk region were ranked by averaging deleterious scores across donors containing the variant. **C.** The absolute $\log_2$ fold change of average height with respect to top-*K* and bottom-*K* ranked variants (K = 20, 40, and 80) around a height-associated gene. *, $P < 0.05$. **D.** Predicting phenotype height with deleterious scores with all variants, top-ranked variants, and bottom-ranked variants. REF, reference; WGS, whole-genome sequencing.

genome-wide association study [34]. For each risk SNP, we defined a risk region as a 200 kb genomic region centered at the SNP. We then ranked SNPs within a risk region according to their cohort-level deleterious scores obtained from donors (Figure 6B). As an illustration, we examined the risk region around a risk SNP (rs5742714) in the promoter region of *IGF1*, a gene encoding a well-known growth factor [35]. The top-ranked variants within this risk region showed an obviously greater absolute $\log_2$ fold change of average height than the bottom-ranked variants (Figure 6C).

We then quantitatively explored how much variance of the height phenotype can be explained by the deleterious scores of risk variants. To achieve this objective, we proposed a linear regression model with l_1 penalty, which took deleterious scores of a set of variants as predictors and the height phenotype as the response variable (see Method). Results show that the 1,103,572 WGS variants within the 3290 risk regions together interpreted 2.49% of the height variance. Furthermore, the variants ranked among the top 10% according to their deleterious scores in each risk region together can interpret 2.11% of the height variance. These results suggest that the small portion of variants prioritized by our method already contained most information that is helpful in the explanation of the phenotype. We also noticed that the bottom-ranked 10% variants, on the contrary, failed to interpret the height phenotype (Figure 6D). To conclude, DeepCAGE is capable of giving a fine mapping of putative risk genetic variants and prioritizing WGS variants that might be associated with a specific phenotype.

Discussion

In this study, we introduce a deep learning framework called DeepCAGE toward genome-wide prediction of chromatin accessibility. A hallmark of our method is the incorporation of the sequence data and the binding statuses of TFs into a unified deep neural network. With these two types of information complementing each other, our method overcomes the limitations of existing approaches and demonstrates state-of-the-art performance in not only classification but also regression of chromatin accessibility signals. Our method provides insights into functional genomics in two aspects. First, the GIS can give us an intuitional measurement of the contribution of a TF to the regulation with respect to a specific locus in a certain cell type. Second, the visualization of convolutional kernels demonstrates that features automatically extracted by our method are not only consistent with existing knowledge but also contain potentially novel binding motifs of TFs. Such interpretability of our model will benefit the dissection of the regulatory landscape under a variety of cell conditions. Our method also provides the possibility of interpreting and prioritizing putative deleterious variants in genetic studies. Such ability in explaining complex traits can further be explored to promote the understanding of disease-associated genetic variants.

Certainly, our model can be further improved from the following aspects. First, currently, we ignore the expression of genes that direct the synthesis of proteins other than TFs. However, it has been shown that proteins such as chromatin regulators, a class of enzymes with specialized function domains, can shape and maintain the epigenetic state in a cell context-dependent fashion [36], and thus can also provide information for inferring chromatin accessible state [37,38]. How to incorporate information on these chromatin regulators into our model is one of the directions in our future work. Second, predicting chromatin accessibility has been explored in a single-cell level [39–42], it is possible to extend the predictive power of DeepCAGE to a single-cell level by incorporating the single-cell gene expression data. Third, our model currently identifies chromatin accessible regions in a cell type-specific manner but cannot further distinguish the specific type of potential regulatory elements in these regions. With the accumulation of annotations regarding *cis*-regulatory elements such as enhancers and silencers [43–46], as well as computational methods for predicting interactions between these elements [47–51], it is expected that our framework can further be extended to uncover the comprehensive relationship between different types of genomic regulatory elements and the genome-wide transcriptomic profile.

Code availability

DeepCAGE is freely available at https://github.com/kimmo1019/DeepCAGE with step-by-step instructions. DeepCAGE is also available at NGDC BioCode with accession https://ngdc.cncb.ac.cn/biocode/tools/BT007170.

CRediT author statement

Qiao Liu: Conceptualization, Software, Formal analysis, Writing - original draft, Writing - review & editing, Visualization. **Kui Hua:** Writing - review & editing. **Xuegong Zhang:** Supervision. **Wing Hung Wong:** Conceptualization, Investigation, Supervision, Writing - review & editing, Funding acquisition. **Rui Jiang:** Conceptualization, Investigation, Supervision, Writing - review & editing, Funding acquisition. All authors have read and approved the final manuscript.

Competing interests

The authors have declared no competing interests.

Acknowledgments

This work has been partly supported by the National Natural Science Foundation of China (Grant Nos. 61721003, 61873141, and 61573207), the National Key R&D Program of China (Grant No. 2018YFC0910404), and the Tsinghua-Fuzhou Institute for Data Technology. This work was also supported by the National Institutes of Health grants (Grant Nos. P50HG007735 and R01HG010359). The Genotype-Tissue Expression project was supported by the Common Fund of the Office of the Director of the National Institutes of Health. We thank Mengmeng Wu, Zhana Duren, and Fengling Chen for their helpful discussions.

Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.gpb.2021.08.015.

ORCID

ORCID 0000-0002-9781-3360 (Qiao Liu)
ORCID 0000-0003-2228-7025 (Kui Hua)
ORCID 0000-0002-9684-5643 (Xuegong Zhang)
ORCID 0000-0001-7466-2339 (Wing Hung Wong)
ORCID 0000-0002-7533-3753 (Rui Jiang)

References

[1] Kellis M, Wold B, Snyder MP, Bernstein BE, Kundaje A, Marinov GK, et al. Defining functional DNA elements in the human genome. Proc Natl Acad Sci U S A 2014;111:6131–8.

[2] Klemm SL, Shipony Z, Greenleaf WJ. Chromatin accessibility and the regulatory epigenome. Nat Rev Genet 2019;20:207–20.

[3] Crawford GE, Holt IE, Whittle J, Webb BD, Tai D, Davis S, et al. Genome-wide mapping of DNase hypersensitive sites using massively parallel signature sequencing (MPSS). Genome Res 2006;16:123–31.

[4] Buenrostro JD, Giresi PG, Zaba LC, Chang HY, Greenleaf WJ. Transposition of native chromatin for fast and sensitive epigenomic profiling of open chromatin, DNA-binding proteins and nucleosome position. Nat Methods 2013;10:1213.

[5] ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. Nature 2012;489:57–74.

[6] Roadmap Epigenomics Consortium, Kundaje A, Meuleman W, Ernst J, Bilenky M, Yen A, et al. Integrative analysis of 111 reference human epigenomes. Nature 2015;518:317–30.

[7] Corces MR, Granja JM, Shams S, Louie BH, Seoane JA, Zhou W, et al. The chromatin accessibility landscape of primary human cancers. Science 2018;362:6413.

[8] Trevino AE, Sinnott-Armstrong N, Andersen J, Yoon SJ, Huber N, Pritchard JK, et al. Chromatin accessibility dynamics in a model of human forebrain development. Science 2020;367:6476.

[9] Song S, Cui H, Chen S, Liu Q, Jiang R. EpiFIT: functional interpretation of transcription factors based on combination of sequence and epigenetic information. Quant Biol 2019;7:233–43.

[10] Zhou J, Troyanskaya OG. Predicting effects of noncoding variants with deep learning-based sequence model. Nat Methods 2015;12:931–4.

[11] Liu Q, Gan M, Jiang R. A sequence-based method to predict the impact of regulatory variants using random forest. BMC Syst Biol 2017;11:7.

[12] Kelley DR, Snoek J, Rinn JL. Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. Genome Res 2016;26:990–9.

[13] Zhou W, Sherwood B, Ji Z, Xue Y, Du F, Bai J, et al. Genome-wide prediction of DNase I hypersensitivity using gene expression. Nat Commun 2017;8:1038.

[14] Liu Q, Xia F, Yin Q, Jiang R. Chromatin accessibility prediction via a hybrid deep convolutional neural network. Bioinformatics 2018;34:732–8.

[15] Min X, Zeng W, Chen N, Chen T, Jiang R. Chromatin accessibility prediction via convolutional long short-term memory networks with *k*-mer embedding. Bioinformatics 2017;33:i92–101.

[16] Xu C, Liu Q, Zhou J, Xie M, Feng J, Jiang T. Quantifying functional impact of non-coding variants with multi-task Bayesian neural network. Bioinformatics 2020;36:1397–404.

[17] Yin Q, Wu M, Liu Q, Lv H, Jiang R. DeepHistone: a deep learning approach to predicting histone modifications. BMC Genomics 2019;20:193.

[18] Quang D, Xie X. DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. Nucleic Acids Res 2016;44:e107.

[19] Ding K, Liu Q, Lee E, Zhou M, Lu A, Zhang S. Feature-enhanced graph networks for genetic mutational prediction using histopathological images in colon cancer. Proc Int Conf Med Image Comput Comput Assist Interv 2020:294–304.

[20] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. Proc IEEE Conf Comput Vision Pattern Recognit 2016:770–8.

[21] Nair S, Kim DS, Perricone J, Kundaje A. Integrating regulatory DNA sequence and gene expression to predict genome-wide chromatin accessibility across cellular contexts. Bioinformatics 2019;35:i108–16.

[22] Kulakovskiy IV, Vorontsov IE, Yevshin IS, Soboleva AV, Kasianov AS, Ashoor H, et al. HOCOMOCO: expansion and enhancement of the collection of transcription factor binding sites models. Nucleic Acids Res 2015;44:D116–25.

[23] Heinz S, Benner C, Spann N, Bertolino E, Lin YC, Laslo P, et al. Simple combinations of lineage-determining transcription factors prime *cis*-regulatory elements required for macrophage and B cell identities. Mol Cell 2010;38:576–89.

[24] Huang G, Liu Z, Weinberger KQ, van der Maaten L. Densely connected convolutional networks. Proc IEEE Conf Comput Vision Pattern Recognit 2017;1:3.

[25] Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. Proc 32^{ed} Inter Conf Mach Learn 2015:448–56.

[26] Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 2014;15:1929–58.

[27] Gupta S, Stamatoyannopoulos JA, Bailey TL, Noble WS. Quantifying similarity between motifs. Genome Biol 2007;8:R24.

[28] Khan A, Fornes O, Stigliani A, Gheorghe M, Castro-Mondragon JA, van der Lee R, et al. JASPAR 2018: update of the open-access database of transcription factor binding profiles and its web framework. Nucleic Acids Res 2017;46:D260–6.

[29] Kent WJ, Sugnet CW, Furey TS, Roskin KM, Pringle TH, Zahler AM, et al. The human genome browser at UCSC. Genome Res 2002;12:996–1006.

[30] Law JC, Ritke MK, Yalowich JC, Leder GH, Ferrell RE. Mutational inactivation of the p53 gene in the human erythroid leukemic K562 cell line. Leuk Res 1993;17:1045–50.

[31] Cheng T, Wang Y, Dai W. Transcription factor egr-1 is involved in phorbol 12-myristate 13-acetate-induced megakaryocytic differentiation of K562 cells. J Biol Chem 1994;269:30848–53.

[32] Gabra MM, Salmena L. MicroRNAs and acute myeloid leukemia chemoresistance: a mechanistic overview. Front Oncol 2017;7:255.

[33] Yang MY, Liu TC, Chang JG, Lin PM, Lin SF. *JunB* gene expression is inactivated by methylation in chronic myeloid leukemia. Blood 2003;101:3205–11.

[34] Yengo L, Sidorenko J, Kemper KE, Zheng Z, Wood AR, Weedon MN, et al. Meta-analysis of genome-wide association studies for height and body mass index in ~ 700000 individuals of European ancestry. Hum Mol Genet 2018;27:3641–9.

[35] Becker NS, Verdu P, Georges M, Duquesnoy P, Froment A, Amselem S, et al. The role of *GHR* and *IGF1* genes in the genetic determination of African pygmies' short stature. Eur J Hum Genet 2013;21:653–8.

[36] Chen T, Dent SY. Chromatin modifiers and remodellers: regulators of cellular differentiation. Nat Rev Genet 2014;15:93.

[37] Duren Z, Chen X, Jiang R, et al. Modeling gene regulation from paired expression and chromatin accessibility data. Proc Natl Acad Sci U S A 2017;114:E4914–23.

[38] Wang Y, Jiang R, Wong WH. Modeling the causal regulatory network by integrating chromatin accessibility and transcriptome data. Natl Sci Rev 2016;3:240–51.

[39] Chen S, Yan G, Zhang W, et al. RA3 is a reference-guided approach for epigenetic characterization of single cells. Nat Commun 2021;12:1–13.

[40] Liu Q, Xu J, Jiang R, et al. Density estimation using deep generative neural networks. Proc Natl Acad Sci U S A 2021;118:e2101344118.

[41] Liu Q, Chen S, Jiang R, et al. Simultaneous deep generative modelling and clustering of single-cell genomic data. Nat Mach Intell 2021;3:536–44.

[42] Chen X, Chen S, Song S, et al. Cell type annotation of single-cell chromatin accessibility data via supervised Bayesian embedding. Nat Mach Intell 2022;4:116–26.

[43] Khan A, Zhang X. dbSUPER: a database of super-enhancers in mouse and human genome. Nucleic Acids Res 2016;44:D164–71.

[44] Zeng W, Min X, Jiang R. EnDisease: a manually curated database for enhancer-disease associations. Database (Oxford) 2019;2019: baz020.
[45] Chen S, Liu Q, Cui X, Feng Z, Li C, Wang X, et al. OpenAnnotate: a web server to annotate the chromatin accessibility of genomic regions. Nucleic Acids Res 2021;49:W483–90.
[46] Zeng W, Chen S, Cui X, Chen X, Gao Z, Jiang R. SilencerDB: a comprehensive database of silencers. Nucleic Acids Res 2021;49: D221–8.
[47] Li W, Wong WH, Jiang R. DeepTACT: predicting 3D chromatin contacts via bootstrapping deep learning. Nucleic Acids Res 2019;47:e60.
[48] Liu Q, Lv H, Jiang R. hicGAN infers super resolution Hi-C data with generative adversarial networks. Bioinformatics 2019;35: i99–107.
[49] Zeng W, Xin J, Jiang R, Wang Y. Reusability report: compressing regulatory networks to vectors for interpreting gene expression and genetic variants. Nat Mach Intell 2021;3:576–80.
[50] Liu Q, Hu Z, Jiang R, Zhou M. DeepCDR: a hybrid graph convolutional network for predicting cancer drug response. Bioinformatics 2020;36:i911–8.
[51] Singh S, Yang Y, Poczos B, Ma J. Predicting enhancer-promoter interaction from genomic sequence with deep neural networks. Quant Biol 2019;7:122–37.