PLOS Computational Biology

RESEARCH ARTICLE

Predictive modeling of gene expression and localization of DNA binding site using deep convolutional neural networks

Arman Karshenas[1*], **Tom Röschinger**[2], **Hernan G. Garcia**[1,3,4,5,6]

1 Biophysics Graduate Group, University of California at Berkeley, Berkeley, California, United States of America, **2** Division of Biology and Biological Engineering, California Institute of Technology, Pasadena, California, United States of America, **3** Department of Molecular and Cell Biology, University of California Berkeley, Berkeley, California, United States of America, **4** Department of Physics, University of California, Berkeley, California, United States of America, **5** Institute for Quantitative Biosciences-QB3, University of California, Berkeley, California, United States of America, **6** Chan Zuckerberg Biohub – San Francisco, San Francisco, California, United States of America

* karshenas@berkeley.edu

OPEN ACCESS

Citation: Karshenas A, Röschinger T, Garcia HG (2026) Predictive modeling of gene expression and localization of DNA binding site using deep convolutional neural networks. PLoS Comput Biol 22(4): e1014092. https://doi.org/10.1371/journal.pcbi.1014092

Editor: Xiao-Jun Tian, Arizona State University, UNITED STATES OF AMERICA

Received: December 18, 2024

Accepted: March 5, 2026

Published: April 1, 2026

Data availability statement: All data and resources associated with this study are openly accessible to promote transparency and reproducibility. The raw data, processed data, analysis results, code, and trained models are available on the GitHub repository at https://github.com/armankarshenas/DARSI. This

Abstract

Despite the sequencing revolution, large swaths of the genomes sequenced to date lack any information about the arrangement of transcription factor binding sites on regulatory DNA. Massively Parallel Reporter Assays (MPRAs) have the potential to dramatically accelerate our genomic annotations by making it possible to measure the gene expression levels driven by thousands of mutational variants of a regulatory region. However, the interpretation of such data often assumes that each base pair in a regulatory sequence contributes independently to the overall gene expression. To enable the analysis of this data in a manner that accounts for possible correlations between distant bases along a regulatory sequence, we developed the Deep learning Adaptable Regulatory Sequence Identifier (DARSI). This convolutional neural network leverages MPRA data for training specific models for each operon to predict gene expression levels directly from raw regulatory DNA sequences. By harnessing this predictive capacity, DARSI systematically identifies transcription factor binding sites within regulatory regions at single-base pair resolution. To validate its predictions, we benchmarked DARSI against curated databases, confirming its accuracy in predicting known transcription factor binding sites. Additionally, DARSI predicted novel unmapped binding sites, paving the way for future experimental efforts to confirm the existence of these binding sites and to identify the transcription factors that target those sites. Thus, DARSI provides a new framework for MPRA experimental data analysis, it generates experimentally actionable predictions that can feed iterations of the theory-experiment cycle aimed at reaching a predictive understanding of transcriptional control.

repository provides detailed instructions and documentation to facilitate the reuse and adaptation of these materials for further research.

Funding: This work was supported by the National Institutes of Health (R01GM139913, R01GM152815, and R35GM158200 to HGG); the Koret-UC Berkeley-Tel Aviv University Initiative in Computational Biology and Bioinformatics (to HGG); the Winkler Scholar Faculty Award (to HGG); the Chan Zuckerberg Initiative (Grant CZIF2024-010479 to HGG); the Miller Institute for Basic Research in Science, University of California Berkeley (to HGG); and the Phyllis B. Blair Graduate Fellowship from the University of California, Berkeley (to AK). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Here, we developed a deep learning approach—called DARSI—that leverages these massively parallel reporter assays to predict levels of gene expression from DNA sequences and help locate these important binding sites. By training our model to recognize DNA sequence patterns that affect gene expression, our method not only finds known binding sites with high accuracy, but also predicts new binding sites that call for future experimental scrutiny.

Author summary

Understanding how genes are turned on and off is a fundamental question in biology. This process is often controlled by transcription factor proteins that bind to specific regions of DNA to activate or repress nearby genes. Identifying where these proteins bind is key to decoding how genes are regulated. Recently, large experimental datasets stemming from so-called massively parallel reporter assays in which thousands of slightly different DNA sequences are tested have opened the door to the identification of these binding sites in high-throughput.

Introduction

A central challenge in biology is to accurately predict gene regulatory programs and their functions from knowledge of genome sequences [1–3]. These programs are governed, in large part, by DNA regulatory regions containing binding sites for transcription factors. These proteins interact with the transcriptional machinery to modulate gene expression by enhancing or repressing transcription.

Achieving such predictive understanding of transcriptional regulation requires addressing two key challenges: (i) identifying and characterizing transcription factor binding sites within regulatory regions and (ii) integrating this knowledge into theoretical models capable of quantitatively predicting how the number, placement and affinity of these binding sites dictate gene expression [2–4]. Thus, the foundational step toward predicting the regulatory outcomes encoded by DNA regulatory regions involves determining the location and identity of transcription factor binding sites.

Despite the key need to map transcription factor binding sites in regulatory regions, our ability to accurately identify these sites is still lacking [5,6]. For instance, in the bacterium *Escherichia coli*, one of the most thoroughly studied model organisms, binding sites regulating only about 33% of genes have been mapped to date [6–8]. While some genes may not be transcriptionally regulated and thus lack transcription factor binding sites, this figure more likely reflects the limited number of detailed case studies conducted so far. The challenge is even more pronounced in multicellular organisms, such as the fruit fly *Drosophila melanogaster*, where regulatory networks are considerably more intricate and less well characterized [9].

Classic approaches for finding and validating binding sites within regulatory regions are typically manual and, therefore, low-throughput. Specifically, these

approaches rely on the creation of reporter constructs where suspected binding sites are mutagenized. By correlating DNA sequence with the resulting reporter expression level, transcription factor binding sites can be validated. As a result of the low-throughput nature of this pipeline, the binding sites controlling only a handful of genes in model organisms have been mapped in detail [10–14].

Massively Parallel Reporter Assays (MPRAs) have recently emerged as a powerful tool for mapping regulatory sequences [8,15–22]. These assays involve synthesizing a large library (>1,000s) of mutagenized variants of a regulatory region and incorporating them into plasmids (Fig 1A and 1B). The plasmid library is then transfected into cells, where, after cell lysis, gene expression levels for each variant are measured in high-throughput using sequencing (Fig 1C and 1D).

By linking the sequences of these mutated regulatory regions to their corresponding gene expression levels (Fig 1E), MPRAs allow for the identification of positions within the sequence that influence gene expression when mutated. As illustrated in Fig 1F, this approach makes it possible to pinpoint transcription factor binding sites in uncharacterized regulatory regions: mutations in activator binding sites typically decrease gene expression, whereas mutations in repressor binding sites tend to increase expression [8,22]. Table A in S1 Text provides an example MPRA dataset.

While MPRAs have significantly advanced the study of regulatory sequences [20,21], key challenges remain in systematically analyzing the resulting datasets to reveal transcription factor binding sites. For example, an important potential limitation lies in the reliance of these analyses on metrics such as gene expression sensitivity to mutation (Fig 1F) or mutual information between gene expression and base pair identity [8,16,23]. These measures often assume that base pairs contribute independently to gene expression: because these metrics evaluate the impact of mutations at specific positions by effectively averaging their effects across all other positions in the sequence, they potentially ignore nucleotide interactions within the regulatory sequence.

Given the recent advances in machine learning and foundational models, we explored how modern sequence modeling techniques could address these limitations and more fully capture dependencies across regulatory regions. In this study,

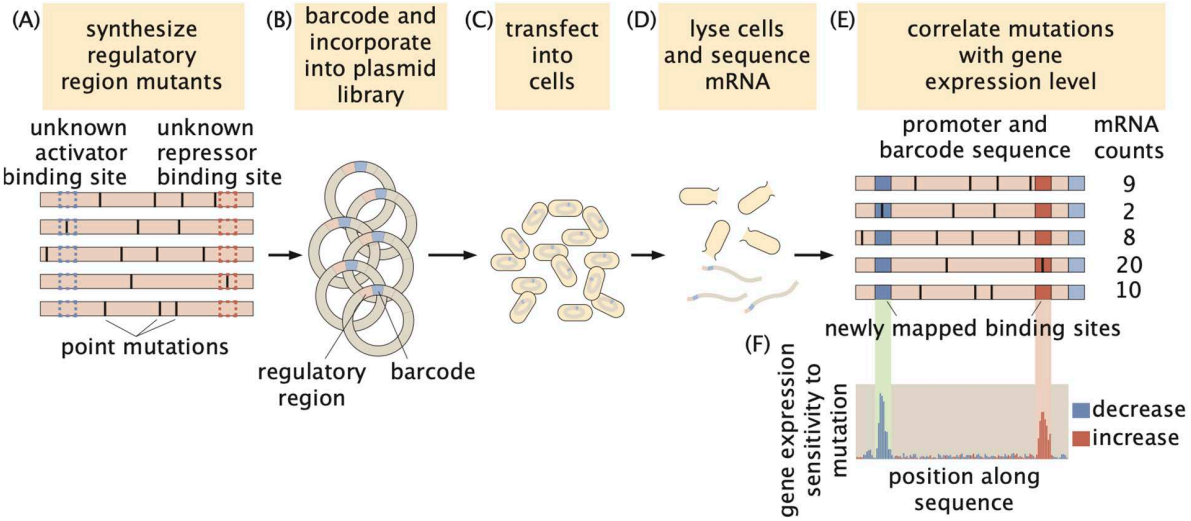

Fig 1. Schematic example of a massively parallel reporter assay to dissect regulatory regions in *E. coli*. (A) A library of mutated versions of a previously uncharted regulatory sequence is synthesized. **(B)** Each sequence is barcoded and incorporated into constructs that drive the expression of a reporter gene, forming a plasmid library. **(C)** The plasmid library is transformed into cultured cells such as *E. coli*. **(D)** After cell lysis, the reporter mRNA is extracted and quantified by sequencing. **(E)** An illustrative example showing the correlation between the regulatory sequence variants and their corresponding gene expression levels. **(F)** These data make it possible to capture the shift in gene expression upon mutagenesis of each base pair along the sequence, leading to the identification of activator and repressor binding sites.

https://doi.org/10.1371/journal.pcbi.1014092.g001

we introduce the Deep Learning Adaptive Regulatory Sequence Identifier (DARSI), a framework developed to address the challenge of binding site identification while accounting for spatial correlations along DNA sequences. DARSI integrates the benefits of convolutional modeling established in recent advances [24–30] while remaining computationally efficient for the scale of typical MPRA datasets. Our method enables prediction of discretized gene expression levels directly from raw regulatory sequences without relying on prior knowledge of regulatory architecture.

The *predictive power* enabled by DARSI, although far from the *predictive understanding* we ultimately seek through physical models [16,22,31–34], makes it possible to obtain detailed insights into the number and spatial arrangement of transcription factor binding sites within regulatory sequences. Hypothesized binding sites are identified through the integration of saliency mapping techniques—akin to an *in silico* mutagenesis experiment—which allow us to interpret the impact of specific nucleotide sequence changes on gene expression outcomes. It must be noted that while DARSI facilitates an in-depth and automated analysis of MPRA datasets, it is not an out-of-the-box machine learning model that can be used on any random sequence data. The pipeline developed here allows the user to feed in any type of MPRA dataset and train DARSI models to infer the locations of high saliency in the input data which, in the particular case of the MPRA experiments considered here, corresponds to transcription factor binding sites.

We applied DARSI to MPRA data from 95 operons in *E. coli* published by [8]. First, we demonstrate that the trained networks achieve an average accuracy of ~80% in predicting the expression levels of the reporter gene directly from the raw sequences. Building on this predictive power, we show that the networks can be leveraged to identify transcription factor binding sites. Specifically, DARSI identified over 170 binding sites, including more than 88% of the previously mapped sites [7], and uncovered 73 new hypothesized binding sites across these operons. Thus, we demonstrate that the convolutional neural network architecture within DARSI can be used to augment analyses of gene expression MRPA data to both achieve predictive power and identify binding sites that can guide further experiments.

Results

DARSI: A convolutional neural network for gene expression prediction from MPRA data

To develop a model capable of predicting gene expression from regulatory sequence variation, we began by evaluating the landscape of existing machine learning architectures in genomics. In particular, large-scale pretrained transformer architectures such as *DNA BERT* and the *Nucleotide Transformer* [35–38] have demonstrated the ability to learn complex long-range interactions in biological sequences. Likewise, deep convolutional neural networks, including *BPNet* and *Enformer* [39,40], have achieved high-resolution predictions of regulatory activity directly from raw DNA input. As described in detail in S2 Text, we reviewed these and related frameworks to assess their potential utility in the context of MPRA-derived data, weighing considerations such as predictive accuracy, interpretability, scalability, and suitability for modestly sized experimental libraries.

Guided by our evaluation, we concluded that a convolutional neural network provided the best compromise between prediction accuracy and interpretability when trained on small datasets. As a result, we implemented a convolutional neural network specifically designed to learn from MPRA-derived regulatory sequences and predict gene expression levels. To reach predictive power over regulatory regions and capture correlations between nucleotides, the model leverages convolutional filters capable of detecting both local and long-range interactions between base pairs [41], potentially enabling the identification of regulatory features distributed across the input sequence. The network takes as input the sequence variants of a given operon and their corresponding expression levels. After iterative optimization, we settled on a 12-layer architecture (see S3 Text) that shares design elements with previously established models in the field [39 –41].

As a case study, we utilized data from a recent MPRA study conducted by [8] in *E. coli*. This work dissected the regulatory information of 114 bacterial operons by randomly mutating a 160 bp region upstream of the transcription start site of each operon at a mutation rate of 10%. This process generated a dataset akin to that featured in the schematic shown in Fig 2A that correlates sequence and gene expression. To ensure sufficient coverage of mutations, we selected operons

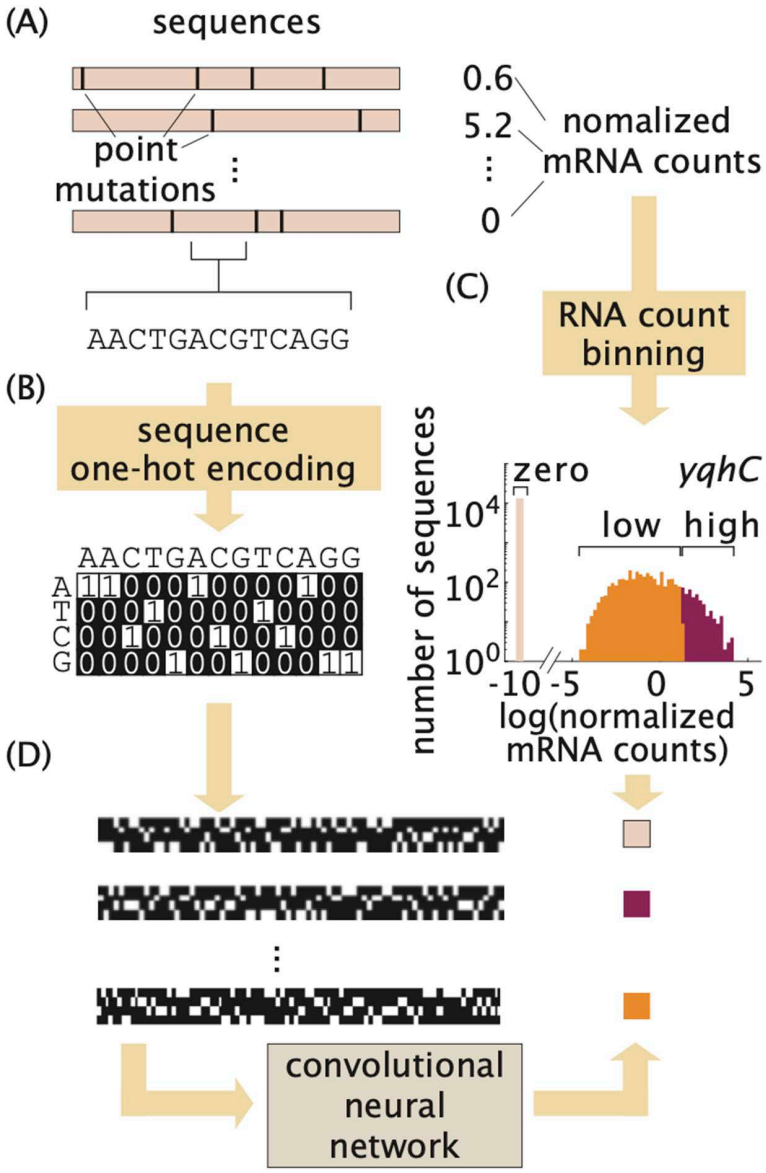

Fig 2. The DARSI pipeline. (A) MPRA dataset that makes it possible to correlate regulatory sequence with gene expression. **(B)** One-hot encoding scheme used to convert each DNA sequence into a binary image. **(C)** Distribution of the log(normalized mRNA count) together with the bin of gene expression assigned to each value for the illustrative case of the *yqhC* operon. Note that the overlap observed between the low and high expression bins is an artifact of the histogram binning and does not reflect an actual overlap between the classes of expression. **(D)** The images and the corresponding gene expression bins constitute the inputs and outputs of our convolutional neural network, respectively.

https://doi.org/10.1371/journal.pcbi.1014092.g002

with at least 1,000 sequence variants. This number guaranteed that, for every base pair along the sequence, our dataset contained at least 100 sequence variants in which that base pair is mutated. We further cross-referenced all sequences with the annotated *E. coli* genome available on *EcoCyc* [42] to verify that the sequences encompassed regions upstream of the genes. The lower bound used for number of variants and the cross-validation of the data with annotated databases reduced the dataset to 95 mutagenized operons, each originating from *E. coli* colonies cultivated in LB medium. Across the 95 operons, the mean number of unique sequences per operon is 2083 ± 960, with 847 ± 193 unique barcodes per

operon. This results in an overall mean of 8313 ± 3228 sequence variants across all operons. The sequence data for each operon served as input to the network, while discretized normalized mRNA counts (described below) were used as the output. A separate convolutional neural network was trained for each operon, resulting in a total of 95 independently trained networks.

Convolutional neural networks are designed to take images or matrices as inputs. Thus, to prepare the DNA sequence data for use as input to our networks, we transformed the sequences into a two-dimensional matrix representation. Specifically, each 160 bp regulatory sequence from the MPRA dataset was encoded as a 4×160 binary image using a so-called one-hot encoding scheme, as illustrated in Fig 2B and detailed in the "One-hot encoding" section of the Materials and methods. Consequently, the data for each operon is represented as a stack of 4×160 images, with each image corresponding to a specific sequence variant for that operon.

In designing the network outputs, we considered two possible approaches: predicting continuous gene expression levels through regression, or discretizing the outputs and framing the problem as a classification task. To ensure comparability across sequences, we first normalized mRNA counts by dividing the number of sequenced mRNAs by the corresponding DNA copy number for each variant (Fig 2A). To assess the regression approach, we trained regression models on the ten operons with the largest number of sequence variants (*leuABCD*, *rumB*, *zupT*, *yncD*, *uvrD*, *mscK*, *ftsK*, *yqhC*, *groSL*, and *xylA*). As detailed in the S4 Text, regression resulted in substantially lower predictive performance, with average R^2 value of 0.06 and normalized RMSE value of 1.02, indicating poor generalization.

Given the poor performance of our networks in the context of predicting continuous gene expression levels through regression, we decided to explore the discretization approach. Specifically, we discretized normalized gene expression values into expression bins, referred to as "classes", and trained the networks to predict the bin associated with each regulatory sequence. This binning allowed us to cast the task as a multi-class classification problem.

Mutations within the sequences can lead to various outcomes, such as a complete absence of detectable gene expression, a measurable reduction, or an increase in gene expression compared to typical levels observed across the sequence variants for each operon. To capture this variation in expression level, we examined the distribution of the logarithm of the normalized expression counts. Based on this distribution, we defined three distinct expression classes: (1) sequences resulting in no detectable gene expression (zero expression bin), (2) sequences yielding low but measurable levels of gene expression (low expression bin), and (3) sequences associated with high levels of gene expression (high expression bin). While the decision to use three bins was informed by the natural clustering of data in the logarithmic space, this choice is ultimately a simplification that, as we will show in the next sections, can still lead to predictive power and the ability to identify transcription factor binding sites.

For each operon we determined the thresholds of log(normalized mRNA count) for each bin to partition the gene expression counts into the three classes. The zero gene expression bin corresponds to sequences that yielded no detectable mRNA. The threshold between the low and high gene expression bins were chosen so as to lead to statistically significant differences in mean gene expression levels between these two classes, as described in detail in the "RNA count labeling" section of the Materials and methods. Fig 2C shows the distribution of log(normalized mRNA count) and the associated bins color-coded for the illustrative *yqhC* operon from the MPRA dataset by [8]. This operon is used throughout the text to illustrate our pipeline and its results, as it represents the average performance of the pipeline. Similar plots to Fig 2C, showing the distribution of expression counts for the rest of the operons in the dataset can be accessed through our GitHub repository.

The number of observations in each expression bin vary significantly. Indeed, as shown in Fig 2C, the bin corresponding to zero gene expression was typically overrepresented with respect to the low and high gene expression bins. To account for this over-representation, the zero gene expression bin was under sampled when training the networks, while the low and high bins were over sampled to create an evenly split processed dataset [43,44].

Training for each network utilized 70% of the processed data, following adjustments for data imbalance. Training was conducted in *MATLAB* using standard optimization toolboxes, with parameters optimized via stochastic gradient descent

[45–47]. An additional 15% of the data (3,000–5,000 variants across the 95 operons) was reserved for validation during training, serving to optimize network architecture as discussed below. The remaining 15% of the data was allocated for final evaluation of the predictive power of each network.

Before engaging in the training of all 95 networks, we optimized the overall network architecture for accuracy in predicting gene expression in our dataset. While adding more convolutional layers should allow the network to extract longer-range interactions between nucleotides along the sequence, increasing the depth of the network leads to a substantial rise in the number of trainable parameters, potentially resulting in overfitting [24]. As a result, we systematically and iteratively modulated the network architecture to assess its impact on prediction accuracy.

To optimize the network architecture, we focused on data from the 10 operons with the largest number of sequence variants. As expected, our optimization revealed that increasing model complexity (e.g., by adding layers and channels) generally improves training accuracy but can lead to overfitting, where the model performs poorly when validated using unseen data. We converged onto an optimal architecture, detailed in Table A in S3 Text, that strikes a balance between model complexity and performance. This chosen architecture is consistent with similar networks implemented in prior studies [27,39,40]. Using this optimized architecture, we independently trained 95 convolutional neural networks, one for each operon in our dataset. Further details on the DARSI architecture, its optimization, and training specifications can be found in the S3 Text.

Finally, we also explored whether training a single model across all operons could enhance predictive power by leveraging a pooled dataset across variants. However, as shown in S5 Text, this approach performed no better than random classification, further motivating our operon-specific modeling strategy.

DARSI can predict gene expression from raw sequence

As outlined above, each trained network, corresponding to an individual operon in the dataset, was evaluated using the reserved 15% of the processed data designated as the test set. For each operon, raw sequences from the test partition were input into the trained network, which then predicted the corresponding output gene expression bin. The predicted bins were compared against the experimentally measured gene expression values to calculate an accuracy score for each network. As illustrated in Fig 3A, the networks achieved an average predictive accuracy of 79.8% across all 95 operons.

To more rigorously evaluate the effectiveness of our DARSI model in predicting gene expression from raw sequence input, we generated confusion matrices. In these matrices, each column represents predicted expression bin (i.e., zero expression, low expression or high expression), while each row indicates the actual bin to which the sequences belong as reported by measurements. Each entry within the matrix indicates the number of sequences belonging to each combination of predicted and measured gene expression bins. Consequently, these matrices provide a summary of false positives, false negatives, true positives, and true negatives for each of the three discrete expression bins.

The confusion matrix for the *yqhC* operon is displayed in Fig 3B. This matrix indicates that the trained model classifies the majority of unseen data for each operon with *high specificity* (low false positive rate) and *high sensitivity* (low false negative rate), as evidenced by the diagonal dominance and the row and column projections shown in Fig 3B. To access a full list of confusion matrices for all the models trained, the reader is referred to the GitHub repository.

Further to benchmark the efficacy of DARSI in predicting gene expression level (bin) from raw sequence, we investigated how its predictive power compares to pre-trained foundational models that have been published recently—see [38,48] for examples. We utilized the 50 million-parameter-all-species model from Nucleotide Transformer (NT) foundation model [38] to generate embeddings from raw sequence data. Said embeddings were then used to train a logistic regression to classify the bin of expression for each gene. The mean classification accuracy across all operons for this model was 48.5% as opposed to 79.8% for the same task from DARSI. This shows that a much simpler DARSI model not only outperforms the large foundational model in this predictive task. The detail of this one-shot foundational model-based classification method is discussed in the S6 Text.

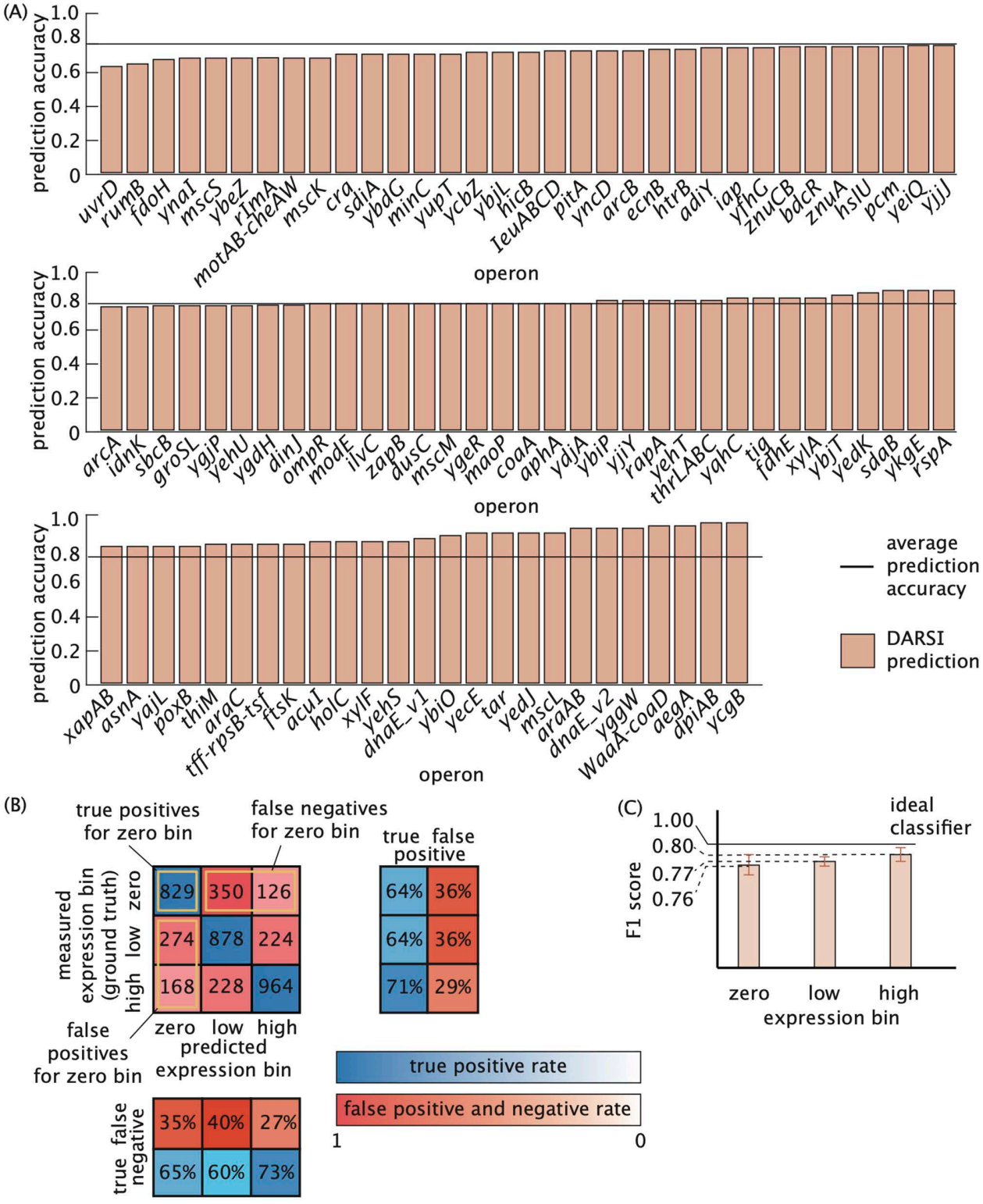

Fig 3. Predictive Performance of DARSI Models on Bacterial MPRA Data. (A) The accuracy of the DARSI model in predicting gene expression levels from previously unseen DNA sequences for each *E. coli* operon in the dataset. The average prediction accuracy of 79.8% is shown as a horizontal line. **(B)** Confusion matrix illustrates the comparison between measured and predicted expression levels for the *yqhC* operon in *E. coli*, highlighting

model performance. Each entry in the matrix represents the number of sequences classified as true positives, true negatives, false positives and false negatives for each gene expression bin. The row projections of the confusion matrix in blue and red are true positive and false positive rates, respectively, while the column projections in blue and red are the true negative and false negative rates, respectively. **(C)** Average F1 score values for the three expression bins are shown with the ideal classifier represented by the solid horizontal line. The values of the F1 score are close to 1, corresponding to an ideal classifier.

https://doi.org/10.1371/journal.pcbi.1014092.g003

To evaluate the overall performance of DARSI across all 95 operons, we computed the average F1 score for each expression bin. The F1 score is a metric that assesses both specificity and sensitivity of classifiers, and is commonly employed to gauge classifier performance [24,49–51]. The F1 score for a given bin of expression is defined as

$$\mathrm{F1} = \frac{\text{true positive}}{\text{true positive} + \frac{1}{2}\left(\text{false positive} + \text{false negative}\right)}, \tag{1}$$

where, for example, "true positive" indicates the number of true positives resulting from our model for a specific bin. According to this definition, an ideal classifier with 100% true positive and 0% false positive and false negative rates will have an F1 score of one. True positives, false negatives, and false positives have been highlighted for the zero gene expression bin of the representative *yqhC* operon in Fig 3B, leading to an average F1 score of 0.64 across the three bins for this operon.

By averaging the F1 score of all DARSI networks, we can compare the average network performance to that of an ideal classifier. Fig 3C presents the F1 score values for the zero expression bin (0.76 ± 0.10), low expression bin (0.77 ± 0.09) and high expression bin (0.80 ± 0.09) averaged over all 95 trained convolutional neural networks, where the error bars indicate the standard deviation. The F1 scores for all three expression bins exceed the threshold of 0.7 that is commonplace in most fields [52,53], indicating that the model effectively distinguishes sequences within these bins with high specificity and sensitivity. Thus, we deemed the gene expression predictions made by the trained DARSI models to be reliable for them to be leveraged in our exploration of regulatory architectures in *E. coli*.

Uncovering binding sites using saliency maps

We next leveraged the predictive power of our networks to identify transcription factor binding sites within unmapped regulatory regions. To achieve this task using MPRA datasets, existing methods typically analyze the mutual information or the shift in gene expression resulting from mutations at individual base pairs (Fig 1F; [8,22,54,33]). These analyses aim to isolate the impact of mutations at each nucleotide on the overall gene expression levels. To make this possible, the contributions of mutations across all other nucleotides within the sequence are averaged. Consequently, these approaches assume that the contributions of individual base pairs to gene expression are independent from one another. In contrast, the convolutional neural network architecture employed in this study makes it possible to account for spatial correlations between nucleotides throughout the sequence.

To examine the capacity of our networks to identify clusters of nucleotides corresponding to binding sites, we created so-called saliency maps. These saliency maps are better understood in the context of convolutional neural networks for image classification. For example, a neural network can be trained to classify images between those featuring a dog and those not featuring a dog [55,56,57]. A saliency map is a heat map that reports on how important each pixel within an image was in making the decision of how to classify that image. In the specific case of the classification of images featuring a dog, we would expect pixels that fall within the dog to carry more information than those pixels that are in the background of the image.

Similarly, because regulatory sequences (Fig 4A) are represented as images using the one-hot encoding approach (Fig 4B), the saliency map of each sequence describes how important each pixel within the image—a binary pattern unique to

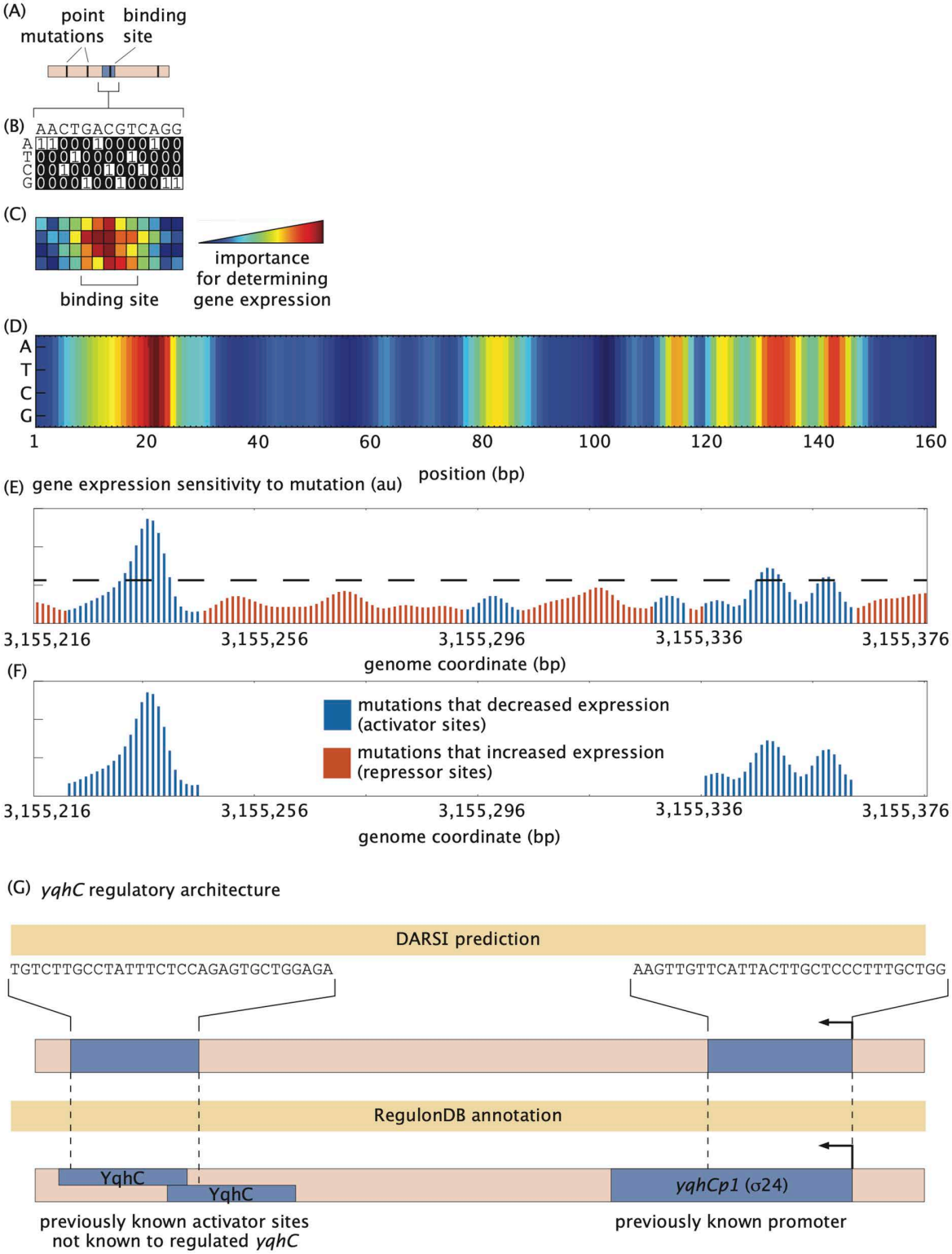

Fig 4. Saliency map generation and binding site identification. (A) Sequence variants from the test subset of a given operon, each containing random point mutations, are processed through the pre-processing pipeline to generate one-hot encodings, as illustrated in **(B)**. **(C)** These one-hot encoded sequences are input into the trained DARSI model, where the gradient of the network loss is computed with respect to each pixel in the input

image for each variant. These gradients measure the sensitivity of the network output to each nucleotide. Gradients are averaged across all variants in the test subset to generate saliency maps, which are represented as 4×160 heatmaps. These heatmaps indicate the pixels containing the most information used by the network to classify the gene expression bin of the input sequence. **(D)** An example saliency map for the illustrative *yqhC* operon highlights regions of high sensitivity. Notably, the saliency values at the same sequence position are relatively insensitive to base pair identity, suggesting that DARSI primarily relies on positional information rather than specific nucleotide identity to predict gene expression. **(E)** Maximum saliency values at each position are normalized and exponentiated to produce unitless plots of gene expression sensitivity to mutations, as shown for the *yqhC* operon. **(F)** These sensitivity plots are further refined by filtering peaks that exceed one standard deviation above the mean (dashed line in **(E)**), span at least 10 bp, and show contiguous effects as either activators or repressors. **(G)** The refined plots enable the identification of potential binding sites and operon regulatory architectures. For the *yqhC* operon, two previously annotated regions were identified: a promoter associated with the operon and an activator binding site, which, although annotated, had not been previously associated with the regulation of *yqhC*.

https://doi.org/10.1371/journal.pcbi.1014092.g004

each sequence—is for determining the gene expression bin corresponding to that sequence (Fig 4C). As a result, these saliency maps can be loosely thought of as heatmaps reporting on the sensitivity of the predicted gene expression level to mutating each nucleotide at every position along the regulatory sequence. As described in detail in the "Saliency maps" section of the Materials and methods, the generation of these maps involves calculating the derivative of the network loss function—a measure of how well the network does at predicting output gene expression—with respect to each pixel of the images encoding for the DNA sequence in the binary input layer of the network.

By applying this process to all sequence variants of a given operon in the test dataset, we generated an ensemble of saliency maps, one for each sequence variants. These maps are then averaged to produce a final saliency map for the operon. As shown in Fig 4D, the saliency map for the illustrative *yqhC* operon used throughout this study reveals several segments along the sequence that exhibit higher information content for predictions made by the trained DARSI model. Notably, minimal variation is observed along individual columns, suggesting that the network primarily considers the positional context of the base pair rather than its specific nucleotide identity when classifying expression levels. These clusters of highly sensitive positions form the initial hypotheses for the locations of binding sites within the sequence.

To interpret the information encoded within the saliency maps, the maximum saliency value among the four nucleotides at each position along the regulatory sequence—that is, along each column of the saliency map—is calculated. The result is a saliency vector that reports on the sensitivity of output gene expression to mutation along the regulatory sequence. Note that the absolute values of saliency maps generated by the network are not inherently interpretable; only relative changes in these values are meaningful. As a result, we normalize the saliency vector by subtracting its mean and dividing by its standard deviation. Subsequently, the normalized saliency vectors are exponentiated to represent likelihoods or probabilities (see the "Binding site identification" section in the Materials and methods). These processed values are visualized as "gene expression sensitivity to mutation" plots. Fig 4E shows an example of this plot for the *yqhC* operon. Because we are after binding site-sized features within these plots, we smoothed the curve by averaging the data using a sliding window of size 5 bp as in previous studies [8]. Further examples are provided in the S1 Fig.

To identify transcription factor binding sites, we examined the smoothed values of the gene expression sensitivity to mutation shown in Fig 4E. Here, red bars correspond to positions along the sequence where mutations led to an increase in expression, suggesting potential repressor binding sites. Conversely, blue bars represent nucleotides where mutations resulted in decreased expression, indicating potential activator binding sites.

To predict binding sites along regulatory sequences, we identify clusters of base pairs with high sensitivity. Specifically, following the approach of [58], we detect positions where the maximum sensitivity exceeds one standard deviation above the mean sensitivity across the entire sequence (horizontal line in Fig 4E). In accordance with the minimum length of DNA binding sites in *E. coli* reported by [59–61], potential binding sites are defined as regions exceeding this threshold and spanning at least 10 base pairs. Fig 4F presents a filtered expression sensitivity-to-mutation plot, highlighting two prominent peaks (blue) corresponding to regulatory regions annotated in RegulonDB [7]. The first peak aligns with a promoter previously mapped to *yqhC*, while the second corresponds to activator binding sites that, although annotated in

RegulonDB, had not been associated with the regulation of *yqhC*. This finding highlights DARSI's ability to identify functional connections between regulatory elements and their target genes. Notably, these activator binding sites are not obvious when examining the mutual information (S2 Fig), which constitutes the basis of previous approaches for identifying binding sites using MPRA data [8]. This particular example demonstrates DARSI's capacity to reveal regulatory features overlooked by traditional methods. A detailed description of the filtering steps employed to generate these expression shift plots is provided in the "Binding site identification" section of the Materials and methods. Further, a comparison of the sensitivity of all operons predicted by DARSI to the same analysis based on mutual information can be found in the GitHub repository.

Using this pipeline, we identified a total of 172 binding sites across all 95 operons, successfully capturing 88.4% of the previously documented sites in published and curated databases [7]. In addition to these annotated binding sites, DARSI predicted 73 hypothetical novel binding sites, spanning more than one-third of the operons in the MPRA dataset (Fig 5A). We classify sites as promoters only if they have been previously mapped as such; otherwise, we label them as activator sites. Additionally, binding sites located within 5–6 bp of each other are reported as a single site for a more conservative assessment.

Fig 5B provides a detailed summary of the binding sites identified by DARSI, the missed binding sites, and the newly predicted hypothetical sites for each operon, alongside annotations from the RegulonDB database [7]. Notably, DARSI failed to identify 13 previously annotated sites in RegulonDB for the *ykgE*, *yicJ*, *rapA*, *yeiQ*, *ybeZ*, *yjjJ*, *tff-rpsB-tsf*, *poxB*, *rspA*, *ompR*, *yjiY*, *znuA*, and *leuABCD* operons. These missed sites were primarily located within regulatory architectures containing multiple binding sites, such as the *ykgE* and *ompR* operons. Examples of regulatory architectures inferred through DARSI are presented in S2 Fig, with comprehensive visualizations for all operons accessible via the GitHub repository. Further, detailed information, including the sequences of each binding site, their genomic coordinates, strand orientation, and prior annotations, is provided in the supplementary table, available for download on the GitHub repository.

Discussion

MPRAs have become a fundamental experimental tool in the high-throughput dissection of the regulatory genome. The data stemming from these experiments has been matched by an increasingly sophisticated suite of approaches to extract as much information as possible. However, it is clear that there is still much room for improvement. For instance, conventional approaches for finding transcription factor binding sites and promoters such as mutual information rely on local measures and assume independence between base pairs [8]. This study highlights the potential of breaking free from the base pair independence assumption and accounting for possible interactions between distant base pairs in a regulatory sequence when finding binding sites within that sequence. In particular, we explored whether the convolutional neural network-based framework embodied in DARSI could enhance the identification of regulatory binding sites and improve our understanding of their roles in dictating gene expression.

For our work, we favored a convolutional neural network architecture as opposed to more recent machine learning approaches such as transformer-based architectures and other large pretrained models for regulatory genomics [40,37,38]. Although these arguably more cutting-edge methods have achieved remarkable predictive performance in large-scale datasets, their application in the context of MPRA experiments presents significant challenges. Specifically, models such as the Nucleotide Transformer [38] contain hundreds of millions to billions of parameters and require extensive training data (e.g., 174B nucleotides for the model trained in [38]) to achieve satisfactory performance. In contrast, our MPRA datasets typically consist of approximately 1,000 variants per operon, which amounts to 160k nucleotides. Thus, the typical amount of available MPRA data is orders of magnitude smaller than the data necessary to train data-hungry architectures such as transformer-based architectures effectively. Moreover, we showed that zero-shot predictions made by pre-trained foundational models without any fine-tuning lack the predictive power when compared to DARSI, which is further discussed in S6 Text.

Fig 5. Benchmarking binding sites sites identified by DARSI against curated RegulonDB dataset. (A) Venn diagram showing the number of binding sites—both transcription factor binding sites and promoters—identified by DARSI and how that number is compared to the previously known sites. DARSI identified a total of 172 binding sites across all 95 operons, capturing 88.4% of previously known sites documented in published and curated databases [7], and predicted the existence of 73 new binding sites. **(B)** Bar plots illustrating the number of sites uncovered by DARSI, the number of sites previously annotated in RegulonDB [7], and the newly identified sites and the sites missed by DARSI across all 95 operons analyzed in this study.

https://doi.org/10.1371/journal.pcbi.1014092.g005

Another important consideration in selecting DARSI over more complex alternatives concerns interpretability. CNNs enable straightforward backpropagation and interpretation of nucleotide-level contributions to model predictions. This backpropagation was fundamental to the generation of saliency maps that constitute the basis of our approach to identifying binding sites. These maps are computed directly on the input matrix (i.e., one-hot encoded nucleotide sequences), allowing us to trace prediction-driving features back to specific base pair positions. By contrast, interpreting transformer-based models with saliency maps or similar methods is still an active area of research. One reason for this difficulty is the use of tokenization—where the input DNA sequence is split into short segments, called k-mers (such as 6-mers), rather than being analyzed as individual nucleotides. This approach fragments the original sequence and makes it challenging to directly link model attention or importance scores back to specific nucleotide positions in the sequence. These practical and methodological factors, together with our emphasis on maintaining compatibility with current MPRA experimental designs, motivated our decision to adopt a convolutional architecture. A detailed discussion of these trade-offs and further justification is provided in S2 Text.

To identify binding sites within regulatory sequences using MPRA data, we first demonstrated that DARSI accurately predicts gene expression levels directly—albeit discretely—from raw regulatory sequences. Importantly, this predictive power is achieved without any underlying assumptions about the physical mechanisms and regulatory grammar dictating gene expression. Building on this foundation, we leveraged saliency maps to highlight regions of high information density that drive model predictions to infer locations of transcription factor binding sites and promoters, which are indistinguishable in this approach. Trained DARSI models identified over 170 binding sites across all 95 operons, including 73 previously unannotated sites and 99 previously mapped sites, accounting for approximately 90% of previously annotated sites in curated databases. Thus, our findings highlight that, while CNN-based models may not achieve the absolute highest predictive accuracy observed in large foundational models, convolutional neural networks provide an effective balance between performance, interpretability, and feasibility in MPRA-scale applications.

To better understand the effectiveness and limitations of DARSI, we benchmarked its predictions of gene expression sensitivity to mutations against those obtained using traditional mutual information approaches [8,16]. S2 Fig presents these comparisons for three representative operons. This comparison highlights how peaks are not always identified by both measures of MRPA data, and how those peaks can be slightly displaced and broader in DARSI with respect to those identified by mutual information. A detailed examination of these two measures—highlighting the potential advantages and challenges of DARSI with respect to mutual information—is provided in S7 Text. While, as discussed in that section, the "black box" nature of machine learning models makes it challenging to dissect the source of these differences, the ultimate proof of the usefulness of DARSI and how it compares against well-established approaches will have to stem from future experiments aimed at validating hypothesized binding sites. Accordingly, we interpret DARSI's novel predictions as prioritized, testable hypotheses rather than definitive regulatory annotations, and emphasize the importance of systematic downstream validation. Future work will focus on integrating conservation analysis, motif-based transcription factor assignment, and existing ChIP or regulatory datasets to further refine and experimentally assess these predictions.

Further, while saliency maps provide valuable insights, it is important to acknowledge their limitations, particularly when applied to discrete variables like base pair identity. This approach can introduce challenges in interpretation due to the underlying reliance on derivatives such as vanishing gradients, where the signals for learning and attribution become very small computationally and make it difficult to assess the importance of individual bases [62].

Regardless of its limitations, These findings establish DARSI, and convolutional neural networks more in general, as a valuable platform for advancing experimental studies of regulatory architectures. For instance, large *in silico* sequence libraries could be generated to complement and refine *in vivo* experiments, facilitating the design of regulatory sequences tailored for specific expression levels [63,64]. It is important to note that while DARSI effectively identifies binding sites, it does not predict the specific transcription factors that bind to these hypothetical sites. Addressing this limitation—as well as elucidating their molecular mechanisms and characterizing their biophysical properties such as binding

affinity—remains a significant challenge that will require the integration of additional computational approaches and experimental validation [8,22,33]. These efforts are essential for advancing our understanding of transcriptional regulation and for improving the utility of predictive models like DARSI in functional genomics.

Although this study focused on bacterial regulatory sequences, the DARSI framework is broadly applicable to differential expression datasets across both bacterial and eukaryotic systems. Beyond gene expression and regulatory sequence activity, DARSI could be adapted for other phenotypic analyses, uncovering causal links and axes of variation in sequence-to-phenotype relationships. For instance, the DARSI methodology could be used to predict protein properties such as solubility, hydrophobic surface composition [65], or binding affinity [66,67], given appropriate training datasets.

Importantly, as with other machine learning approaches, the efficacy of DARSI depends heavily on the quality and scale of its training data. Advances in mutagenesis technologies leading to larger MPRA datasets with higher numbers of variants and broader coverage promise to further amplify the utility of frameworks like DARSI, opening new avenues for precision in computational biology. In particular, our lab envisions exploring DARSI in the context of new mutagenesis technologies that will make it possible to implement MPRAs in multicellular organisms [68].

Materials and methods

MPRA dataset from *E. coli*

The dataset used throughout this article was generated through the work by [8]. Here, as shown diagrammatically in Fig 1A, a 160-bp long region around the transcription start site of 114 operons in *E. coli* were randomly mutated at a 10% rate (i.e., each base pair along the sequence had a 10% chance of being mutated from its wild-type base to any of its three alternative bases). This library was then cloned into plasmids driving the expression of a reporter gene (Fig 1B and 1C). Plasmid libraries were transformed into cells and grown in various growth conditions (though in this work we only focus on bacteria grown in LB). The expression from each operon variant was measured by sequencing (Fig 1C).

To normalize for the variation in copy number for each reporter construct, DNA counts of the barcode were also included in the table and were used to normalize the expression counts as discussed in the main text. Processed dataset, therefore, provides both the sequence of the regulatory region and the normalized expression count of the gene regulated by that sequence (Fig 1E). In this study, we only considered operons with enough sequence variants to ensure that, on average, each base pair was mutated in at least 100 variants (i.e., 100x coverage). Given a 10% mutation rate, this corresponds to a minimum of $\sim 1,000$ variants. Examples of some of these sequence variants within this dataset for the *yqhC* operon are provided in S1 Text.

RNA-seq raw data processing

The sequencing datasets used in this work are deposited in the SRA database as PRJNA599253 and PRJNA603368. Code for sequence processing is provided in the Github repository together with example datasets and Jupyter Notebooks that display how to use the data to generate, for example, the processed data outlined in S1 Text. Here, we give a brief description of the process.

Random barcodes were cloned between the promoter and the reporter gene in order to identify the promoter variant through the RNA reads, as well as provide multiple distinct data points that reduce possible bias introduced by barcodes. In an initial sequencing run the promoter sequence and barcodes were sequenced simultaneously to obtain a map that links a regulatory region variant with the corresponding barcode. Pair end reads were merged, quality filtered, and filtered for read length using “fastp” [69]. Promoter sequence and barcode were extracted from each read and the number of occurrences of each barcode and promoter combination counted. A promoter variant can have multiple barcodes associated with it, however, a barcode has to be unique. If a barcode was observed for multiple promoter variants, the barcode was then removed. Additionally, combinations with less than 3 reads were removed due to the possibility of sequencing errors.

In Reg-Seq, the promoter library is grown in various growth conditions to assess a variety of regulatory conditions. For the purpose of this paper, we take one of these growth conditions: growth in LB. From each culture both RNA and DNA (plasmids) are extracted. Using specific primers, the reporter gene mRNA, including the barcode, is reverse transcribed and amplified to generate cDNA and measure gene expression. Barcodes are also amplified from plasmids using PCR in order to count the number of plasmids present with a specific regulatory sequence. Sequencing adapters are added by another PCR and both barcodes obtained from cDNA and plasmid DNA are sequenced. Reads are trimmed and quality filtered using 'fastp' [69]. The occurrence of each barcode is counted in the RNA-Seq and DNA-Seq datasets. Finally, using the results from the initial sequencing run, each corresponding promoter variant is identified through its barcode.

One-hot encoding

Every 160 bp long regulatory sequence from the MPRA dataset is converted to a two-dimensional binary image $A \in \mathbb{R}^{4\times 160}$, where each entry of the matrix $A_{i,j}$ takes the form

$$A_{i,j} = \begin{cases} A_{1,j} = 1 & \text{if } n_j = A \quad 0 \quad \text{otherwise} \\ A_{2,j} = 1 & \text{if } n_j = T \quad 0 \quad \text{otherwise} \\ A_{3,j} = 1 & \text{if } n_j = C \quad 0 \quad \text{otherwise} \\ A_{4,j} = 1 & \text{if } n_j = G \quad 0 \quad \text{otherwise} \end{cases} \tag{2}$$

Here, n_j is the j^{th} nucleotide in each of the sequences. Fig 2D shows an example of these binary images generated for the regulatory sequences for the *yqhC* operon.

RNA count labeling

The RNA count corresponding to each sequence variant reports on the gene expression level driven by that mutated regulatory region. These values are normalized by dividing the RNA count by the DNA copy number count to ensure that variability in the normalized RNA count is not due to variability in the plasmid copy number.

To bin the normalized expression counts, we developed a binning algorithm to categorize sequence variants into three discrete groups based on their normalized RNA counts: (1) sequences that resulted in zero gene expression, (2) sequences that resulted in low gene expression levels, and (3) sequences that resulted in high gene expression levels. The pipeline first bins all the zero expression counts to the zero expression bin. It then automatically determines the best threshold for separating the remaining gene expression data into the low and high gene expression bins based on their log(normalized mRNA count). To make this possible, a t-test is conducted on the difference of the mean gene expression of the low and high expression bin, leading to a separation of data that minimizes the p-value between the bins.

The vector $\mathcal{Y}_i \in \mathbb{R}^{N_i\times 1}$ encodes for the expression bins associated with each of the N_i sequence variants for the operon i in the dataset. Each sequence variant for a given operon i is given a label from a set $K_i = \{1,2,3\}$ where 1, 2, and 3 denote the zero, low, and high expression bins respectively. The algorithm we implemented attempts to partition the vector $\mathcal{Y}_i$ into three bins such that the p-value associated with a t-test conducted between each pair of bins is minimized [71,70]. The iterative algorithm used to bin the normalized mRNA counts is given in Algorithm 1.

Algorithm 1 RNA count binning algorithm

```
1: Take vector y_i ∈ ℝ^(N_i×1) as input, where y_i = log(normalized expression count)
2: Values that are infinite in the input vector y_i are associated with a label of 1 for the zero
   expression bin
3: Initialize best_bins as an empty cell array of size 2 to store the label associated with vari-
   ant that have finite log(normalized expression count)
```

```
Set best_p_values as an array of size 2 to store lowest p-values obtained throughout the algorithm with all values initialized to ∞
for iter=1 to max_iteration (set to 10,000 in this paper) do
  Generate random thresholds for 1 bin edge separating the low and high expression classes in the range [min(RNA count), max(RNA count)]
  bins←Label observations with labels {2,3} that fall within the thresholds for each bin
  p-values←the p-value associated with a t-test performed on the mean of the low and high expression bins log(normalized expression count)
  if p_values < best_p_values then
    Update best_p_values←p_values
    Update best_bins←bins
  end if
end for
```

Saliency maps

Trained DARSI models demonstrate the capability to predict operon expression levels directly from their nucleotide sequences. Beyond prediction, these models can be utilized to identify potential binding sites within regulatory sequences. This is achieved by analyzing the derivative of the network's loss function (described in detail below) with respect to the input sequence. By computing these gradients, the nucleotides most critical to the model's predictive understanding of gene expression can be identified.

The performance of the network is quantified using a cross-entropy loss function defined as

$$L(\vec{p}, \vec{y}) = \sum_{i=1}^{3} y_i \log(p_i), \tag{3}$$

where, $\vec{y} = \{y_1, y_2, y_3\}$ denotes the ground truth label vector for each variant of a given operon. For instance, a variant assigned to the zero expression bin is represented as $\vec{y} = \{1, 0, 0\}$. Similarly, $\vec{p}$ represents the probability vector predicted by the network for the same sequence. This three-dimensional vector specifies the predicted probabilities of the sequence belonging to each expression bin, as determined by the model.

This loss function measures the discrepancy between the network's predictions and the ground truth, serving as an indicator of model accuracy. To assess the sensitivity of classification outputs to perturbations in nucleotide sequences, we leverage the gradient-weighted class activation mapping (Grad-CAM) approach [55,57,72]. Grad-CAM computes the gradient of a selected, differentiable output—such as the cross-entropy loss—with respect to neurons or nodes in a specified layer of the network, typically a convolutional layer. For a toy example demonstrating the computation of saliency maps through backpropagation, please refer to S8 Text.

This method allows for the visualization of features critical to the model's predictions by backpropagating the gradients through the network and overlaying them on the input sequence. The resulting gradient map highlights the pixels within the input image (corresponding to nucleotides within the sequence) that significantly contribute to the network's decision-making process. By identifying these key features, Grad-CAM enhances the interpretability of deep learning models and provides insights into the regulatory architecture underlying gene expression [55,57,72].

For a two-dimensional image classifier such as DARSI, the saliency score for any given channel in a convolutional layer with k channels is computed by

$$\alpha_k^c = \frac{1}{N} \sum_i \sum_j \frac{\partial y^c}{\partial A_{i,j}^k}, \tag{4}$$

where y^c is the predicted posterior for the bin c, $A^k_{i,j}$ is the pixel located at (i,j) position of the k^{th} channel of the chosen convolutional layer, and N is the total number of pixels [55].

Equation 4 generates a weighting score for every channel k within a convolutional layer. In order to plot this score as a heatmap for any given sequence, similar to the ones shown in Fig 4C and 4D, a weighted-average mask U is computed such that

$$U^c = f\left(\sum_k \alpha_k^c A^k\right), \tag{5}$$

where f represents a non-linear activation function such as the rectified linear unit function ReLU $f = \max(0, x)$ [55]. Algorithm 2 summarizes the steps that are taken to generate these saliency maps for each of the genes within the expression shift dataset.

Algorithm 2 Saliency map generation.

```
for each operon i do
  Train and find the optimized DARSI model
  Load the test subset of the data for the operon i
  Run the Grad-CAM script to generate the masks (U^c) for each variant in the test subset
  Overlay the mask for each image to the input binary image (one-hot encoding representation of
  the sequence (Fig 4B) and plot this as a heatmap that shows the relative importance of each
  base pair (Fig 4C)
  Compute a final saliency map by taking an average over all the maps generated across the test
  data (Fig 4D)
  Compute a 1-dimensional saliency vector B_i ∈ ℝ^{160×1} for each operon i from the heatmap by taking
  the maximum value at each nucleotide position, namely B_i = max U_j^c for j ∈{1, 2,…, 160}
  Plot each of the B_i as a function of base pair position (Fig 4E and 4F).
end for
```

Binding site identification

As discussed in Algorithm 2 above, the saliency vector $\vec{B}_i \in \mathbb{R}^{1\times 160}$ computed for a given operon i captures the saliency of each regulatory sequence. The vector $\vec{B}_i$ is normalized about its mean and standard deviation, namely

$$\vec{S}_i^* = \frac{\vec{B}_i - \bar{\vec{B}}_i}{\sigma(\vec{B}_i)}, \tag{6}$$

where $\bar{\vec{B}}_i$ and $\sigma(\vec{B}_i)$ are the mean and standard deviation of the vector $\vec{B}_i$, respectively.

The vector $\vec{S}_i^*$ represents a difference from the mean in sensitivity of expression level to mutation at any given position j. Therefore we assume that this vector is proportional to the derivative of the dissociation constant with respect to that nucleotide, or more formally

$$\vec{S}_i^*(j) \propto \frac{\partial K_D}{\partial n_j}, \tag{7}$$

where K_D and n_j are the dissociation constant and nucleotide at position j, respectively.

Finally, we used this proportionality to estimate the probability of occupancy $P(j)_i$ as

$$\vec{P}_i(j) \propto \exp\left(|\vec{S}_i^*(j)|\right), \tag{8}$$

where $|\cdot|$ denotes the absolute value. The use of absolute values is necessary due to prior normalization of the vector $\vec{S}_i^*(j)$, which ensures that both strongly negative and strongly positive normalized values contribute to the probability estimate. This adjustment is crucial to account for regions associated with activators (negative values) and repressors or other functional elements (positive values), ensuring a strong signal is captured in both cases.

The probability was computed for every position *j* for every operon *i* and was plotted as bar charts to show the expression shift (Fig 4E). The peaks in probability that were more than one standard deviation from the mean were selected (Fig 4F). These filtered peaks were then passed through a secondary filter to select only regions where the length of a continuous region of repression or activation (i.e., the predicted binding site) is more than 10 bp long—the minimum length of the binding sites in *E. coli* [59–61]. The process for generating filtered expression shift plots is shown in Algorithm 3.

Algorithm 3 Binding sites identification.

for Each operon i **do**
  Generate saliency map $\vec{B}_i \in \mathbb{R}^{1\times 160}$.
  Normalize vector $\vec{B}_i$ using equation 6 to generate vector $\vec{S}_i^*$.
  Generate an exponential vector $\vec{E}_i = \exp|\vec{S}_i^*|$.
  Find elements within $\vec{E}_i$ that are one standard deviation above the mean of $\vec{E}_i$
  Initiate an empty vector $\vec{F}_i \in \mathbb{R}^{1\times 160}$ to store final expression shift data.
  for Each peak k found in previous step **do**
    Check that the region around the peak is continuous in either repression or activation using the sign of the normalized saliency vector $\vec{S}_i^*$ and that the region is larger than 10 nucleotide
    If a region is found to be above 10 bp long, store the region $\vec{S}_i(j:j+1)$ into the final expression shift vector $\vec{F}_i$
  end for
  Plot $\vec{F}_i$ for the operon i
end for

The DARSI pipeline repository

To enhance the accessibility and usability of DARSI for gene expression prediction and related applications, we have designed the implementation with a modular architecture. Each *MATLAB* script operates independently, accompanied by comprehensive documentation for ease of understanding. A master script is also provided to sequentially execute the necessary functions, offering detailed guidance on processing raw RNA-Seq data and training a DARSI model.

The complete set of scripts is available in a dedicated GitHub repository. The repository includes modules for processing raw RNA-Seq data, generating expression shift datasets, training DARSI models, performing cross-validation, evaluating model performance, and identifying binding sites. Additionally, all data used to train the DARSI model, along with outputs such as saliency maps, confusion matrices, and expression shift plots, are made available in the repository.

Supporting information

S1 Text. MPRA dataset example.
(PDF)

S2 Text. Comparing DARSI to other computational approaches.
(PDF)

S3 Text. DARSI architecture and training.
(PDF)

S4 Text. Regression vs classification for predicting gene expression.
(PDF)

S5 Text. Performance of a single CNN trained across all operons.
(PDF)

S6 Text. Foundational model predictive power.
(PDF)

S7 Text. Comparing the DARSI and mutual information approaches.
(PDF)

S8 Text. Example of saliency map computation using backpropagation.
(PDF)

S1 Fig. Gene expression sensitivity to mutation plots.
(PDF)

S2 Fig. Expression plots comparison.
(PDF)

Acknowledgments

We would like to thank Rob Philips and Julia Falo-Sanjuan for comments on the manuscript.

Author contributions

Conceptualization: Arman Karshenas, Hernan G. Garcia.

Data curation: Arman Karshenas, Tom Röschinger.

Formal analysis: Arman Karshenas.

Funding acquisition: Hernan G. Garcia.

Investigation: Arman Karshenas.

Methodology: Arman Karshenas.

Project administration: Hernan G. Garcia.

Resources: Arman Karshenas, Tom Röschinger.

Software: Arman Karshenas, Tom Röschinger.

Supervision: Hernan G. Garcia.

Validation: Arman Karshenas.

Visualization: Arman Karshenas, Hernan G. Garcia.

Writing – original draft: Arman Karshenas, Hernan G. Garcia.

Writing – review & editing: Arman Karshenas, Tom Röschinger, Hernan G. Garcia.

References

1. Pennacchio LA, Bickmore W, Dean A, Nobrega MA, Bejerano G. Enhancers: five essential questions. Nat Rev Genet. 2013;14(4):288–95. https://doi.org/10.1038/nrg3458 PMID: 23503198
2. Phillips R, Belliveau NM, Chure G, Garcia HG, Razo-Mejia M, Scholes C. Figure 1 theory meets figure 2 experiments in the study of gene expression. Annu Rev Biophys. 2019;48:121–63. https://doi.org/10.1146/annurev-biophys-052118-115525 PMID: 31084583
3. Bintu L, Buchler NE, Garcia HG, Gerland U, Hwa T, Kondev J, et al. Transcriptional regulation by the numbers: models. Curr Opin Genet Dev. 2005;15(2):116–24. https://doi.org/10.1016/j.gde.2005.02.007 PMID: 15797194

4. Stormo GD. DNA binding sites: representation and discovery. Bioinformatics. 2000;16(1):16–23. https://doi.org/10.1093/bioinformatics/16.1.16 PMID: 10812473
5. Minchin SD, Busby SJW. Analysis of mechanisms of activation and repression at bacterial promoters. Methods. 2009;47(1):6–12. https://doi.org/10.1016/j.ymeth.2008.10.012 PMID: 18952178
6. Santos-Zavaleta A, Salgado H, Gama-Castro S, Sánchez-Pérez M, Gómez-Romero L, Ledezma-Tejeida D, et al. RegulonDB v 10.5: tackling challenges to unify classic and high throughput knowledge of gene regulation in *E. coli* K-12. Nucleic Acids Res. 2019;47(D1):D212–20. https://doi.org/10.1093/nar/gky1077 PMID: 30395280
7. Tierrafría VH, Rioualen C, Salgado H, Lara P, Gama-Castro S, Lally P, et al. RegulonDB 11.0: comprehensive high-throughput datasets on transcriptional regulation in *Escherichia coli* K-12. Microb Genom. 2022;8(5):mgen000833. https://doi.org/10.1099/mgen.0.000833 PMID: 35584008
8. Ireland WT, Beeler SM, Flores-Bautista E, McCarty NS, Röschinger T, Belliveau NM, et al. Deciphering the regulatory genome of *Escherichia coli*, one hundred promoters at a time. Elife. 2020;9:e55308. https://doi.org/10.7554/eLife.55308 PMID: 32955440
9. Keränen SVE, Villahoz-Baleta A, Bruno AE, Halfon MS. REDfly: an integrated knowledgebase for insect regulatory genomics. Insects. 2022;13(7):618. https://doi.org/10.3390/insects13070618 PMID: 35886794
10. Müller-Hill B. The lac operon. Berlin, New York: De Gruyter; 1996.
11. Schleif R. AraC protein: a love-hate relationship. Bioessays. 2003;25(3):274–82. https://doi.org/10.1002/bies.10237 PMID: 12596232
12. Ptashne M. A genetic switch. 3 ed. New York, NY: Cold Spring Harbor Laboratory Press.
13. Weickert MJ, Adhya S. The galactose regulon of *Escherichia coli*. Mol Microbiol. 1993;10(2):245–51. https://doi.org/10.1111/j.1365-2958.1993.tb01950.x PMID: 7934815
14. Levine M. Transcriptional enhancers in animal development and evolution. Curr Biol. 2010;20(17):R754-63. https://doi.org/10.1016/j.cub.2010.06.070 PMID: 20833320
15. Patwardhan RP, Lee C, Litvin O, Young DL, Pe'er D, Shendure J. High-resolution analysis of DNA regulatory elements by synthetic saturation mutagenesis. Nat Biotechnol. 2009;27(12):1173–5. https://doi.org/10.1038/nbt.1589 PMID: 19915551
16. Kinney JB, Murugan A, Callan CG Jr, Cox EC. Using deep sequencing to characterize the biophysical mechanism of a transcriptional regulatory sequence. Proc Natl Acad Sci U S A. 2010;107(20):9158–63. https://doi.org/10.1073/pnas.1004290107 PMID: 20439748
17. Patwardhan RP, Hiatt JB, Witten DM, Kim MJ, Smith RP, May D, et al. Massively parallel functional dissection of mammalian enhancers in vivo. Nat Biotechnol. 2012;30(3):265–70. https://doi.org/10.1038/nbt.2136 PMID: 22371081
18. Melnikov A, Murugan A, Zhang X, Tesileanu T, Wang L, Rogov P, et al. Systematic dissection and optimization of inducible enhancers in human cells using a massively parallel reporter assay. Nat Biotechnol. 2012;30(3):271–7. https://doi.org/10.1038/nbt.2137 PMID: 22371084
19. Kwasnieski JC, Mogno I, Myers CA, Corbo JC, Cohen BA. Complex effects of nucleotide variants in a mammalian cis-regulatory element. Proc Natl Acad Sci U S A. 2012;109(47):19498–503. https://doi.org/10.1073/pnas.1210678109 PMID: 23129659
20. Kreimer A, Ashuach T, Inoue F, Khodaverdian A, Deng C, Yosef N, et al. Massively parallel reporter perturbation assays uncover temporal regulatory architecture during neural differentiation. Nat Commun. 2022;13(1):1504. https://doi.org/10.1038/s41467-022-28659-0 PMID: 35315433
21. Zheng Y, VanDusen NJ. Massively parallel reporter assays for high-throughput in vivo analysis of cis-regulatory elements. J Cardiovasc Dev Dis. 2023;10(4):144. https://doi.org/10.3390/jcdd10040144 PMID: 37103023
22. Belliveau NM, Barnes SL, Ireland WT, Jones DL, Sweredoski MJ, Moradian A, et al. Systematic approach for dissecting the molecular mechanisms of transcriptional regulation in bacteria. Proc Natl Acad Sci U S A. 2018;115(21):E4796–805. https://doi.org/10.1073/pnas.1722055115 PMID: 29728462
23. Röschinger T, Lee HJ, Pan RW, Solini G, Faizi K, Quan B, et al. The Environment-dependent regulatory landscape of the *E. coli* genome. bioRxiv. 2025.
24. Alzubaidi L, Zhang J, Humaidi AJ, Al-Dujaili A, Duan Y, Al-Shamma O, et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. J Big Data. 2021;8(1):53. https://doi.org/10.1186/s40537-021-00444-8 PMID: 33816053
25. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436–44. https://doi.org/10.1038/nature14539 PMID: 26017442
26. Park Y, Kellis M. Deep learning for regulatory genomics. Nat Biotechnol. 2015;33(8):825–6. https://doi.org/10.1038/nbt.3313 PMID: 26252139
27. de Almeida BP, Reiter F, Pagani M, Stark A. DeepSTARR predicts enhancer activity from DNA sequence and enables the de novo design of synthetic enhancers. Nat Genet. 2022;54(5):613–24. https://doi.org/10.1038/s41588-022-01048-5 PMID: 35551305
28. Kelley DR, Snoek J, Rinn JL. Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. Genome Res. 2016;26(7):990–9. https://doi.org/10.1101/gr.200535.115 PMID: 27197224
29. Kelley DR, Reshef YA, Bileschi M, Belanger D, McLean CY, Snoek J. Sequential regulatory activity prediction across chromosomes with convolutional neural networks. Genome Res. 2018;28(5):739–50. https://doi.org/10.1101/gr.227819.117 PMID: 29588361
30. Zrimec J, Buric F, Kokina M, Garcia V, Zelezniak A. Learning the regulatory code of gene expression. Front Mol Biosci. 2021;8:673363. https://doi.org/10.3389/fmolb.2021.673363 PMID: 34179082
31. Barnes SL, Belliveau NM, Ireland WT, Kinney JB, Phillips R. Mapping DNA sequence to transcription factor binding energy in vivo. PLoS Comput Biol. 2019;15(2):e1006226. https://doi.org/10.1371/journal.pcbi.1006226 PMID: 30716072

32. Tareen A, Kooshkbaghi M, Posfai A, Ireland WT, McCandlish DM, Kinney JB. MAVE-NN: learning genotype-phenotype maps from multiplex assays of variant effect. Genome Biol. 2022;23(1):98. https://doi.org/10.1186/s13059-022-02661-7 PMID: 35428271
33. Pan RW, Röschinger T, Faizi K, Phillips R. Dissecting endogeneous genetic circuits from first principles. bioRxiv. 2024.
34. Lagator M, Sarikas S, Steinrueck M, Toledo-Aparicio D, Bollback JP, Guet CC, et al. Predicting bacterial promoter function and evolution from random sequences. Elife. 2022;11:e64543. https://doi.org/10.7554/eLife.64543 PMID: 35080492
35. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, et al. Highly accurate protein structure prediction with AlphaFold. Nature. 2021;596(7873):583–9. https://doi.org/10.1038/s41586-021-03819-2 PMID: 34265844
36. Rao RM, Liu J, Verkuil R, Meier J, Canny J, Abbeel P, et al. Msa transformer. In Meila M, Zhang T, editors, In: *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 2021. p. 8844–56. PMLR.
37. Ji Y, Zhou Z, Liu H, Davuluri RV. DNABERT: pre-trained bidirectional encoder representations from transformers model for DNA-language in genome. Bioinformatics. 2021;37(15):2112–20. https://doi.org/10.1093/bioinformatics/btab083 PMID: 33538820
38. Dalla-Torre H, Gonzalez L, Mendoza-Revilla J, Lopez Carranza N, Grzywaczewski AH, Oteri F, et al. Nucleotide transformer: building and evaluating robust foundation models for human genomics. Nat Methods. 2025;22(2):287–97. https://doi.org/10.1038/s41592-024-02523-z PMID: 39609566
39. Avsec Ž, Weilert M, Shrikumar A, Krueger S, Alexandari A, Dalal K, et al. Base-resolution models of transcription-factor binding reveal soft motif syntax. Nat Genet. 2021;53(3):354–66. https://doi.org/10.1038/s41588-021-00782-6 PMID: 33603233
40. Avsec Ž, Agarwal V, Visentin D, Ledsam JR, Grabska-Barwinska A, Taylor KR, et al. Effective gene expression prediction from sequence by integrating long-range interactions. Nat Methods. 2021;18(10):1196–203. https://doi.org/10.1038/s41592-021-01252-x PMID: 34608324
41. Alipanahi B, Delong A, Weirauch MT, Frey BJ. Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. Nature Biotechnology. 2015;33(8):831–8.
42. Moore LR, Caspi R, Boyd D, Berkmen M, Mackie A, Paley S, et al. Revisiting the y-ome of *Escherichia coli*. Nucleic Acids Res. 2024;52(20):12201–7. https://doi.org/10.1093/nar/gkae857 PMID: 39373482
43. Bowyer KW, Chawla NV, Hall LO, Kegelmeyer WP. SMOTE: synthetic minority over-sampling technique. CoRR. 2011.
44. He H, Garcia EA. Learning from imbalanced data. IEEE Trans Knowl Data Eng. 2009;21(9):1263–84.
45. MathWorks. Matlab version: 9.13.0 (r2022b). 2022.
46. Bottou L. Online algorithms and stochastic approximations. In: Saad D, editor. Online learning and neural networks. Cambridge, UK: Cambridge University Press; 1998.
47. Sra S, Nowozin S, Wright SJ. Optimization for machine learning. London, England: MIT Press; 2011.
48. Alamos S, Reimer A, Westrum C, Turner MA, Talledo P, Zhao J, et al. Minimal synthetic enhancers reveal control of the probability of transcriptional engagement and its timing by a morphogen gradient. Cell Syst. 2023;14(3):220-236.e3. https://doi.org/10.1016/j.cels.2022.12.008 PMID: 36696901
49. Sokolova M, Lapalme G. A systematic analysis of performance measures for classification tasks. Inf Process Manag. 2009;45(4):427–37. https://doi.org/10.1016/j.ipm.2009.03.002
50. Powers DMW. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. 2020.
51. Aloysius N, Geetha M. A review on deep convolutional neural networks. In: 2017 International Conference on Communication and Signal Processing (ICCSP). 2017. 0588–92. https://doi.org/10.1109/iccsp.2017.8286426
52. Lipton ZC, Elkan C, Narayanaswamy B. Thresholding classifiers to maximize f1 score. 2014.
53. Hicks SA, Strümke I, Thambawita V, Hammou M, Riegler MA, Halvorsen P, et al. On evaluation metrics for medical applications of artificial intelligence. Sci Rep. 2022;12(1):5979. https://doi.org/10.1038/s41598-022-09954-8 PMID: 35395867
54. Kheradpour P, Ernst J, Melnikov A, Rogov P, Wang L, Zhang X, et al. Systematic dissection of regulatory motifs in 2000 predicted human enhancers using a massively parallel reporter assay. Genome Res. 2013;23(5):800–11. https://doi.org/10.1101/gr.144899.112 PMID: 23512712
55. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-CAM: visual explanations from deep networks via gradient-based localization. In: 2017 IEEE International Conference on Computer Vision (ICCV). 2017. 618–26. https://doi.org/10.1109/iccv.2017.74
56. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. ImageNet large scale visual recognition challenge. Int J Comput Vis. 2015;115(3):211–52. https://doi.org/10.1007/s11263-015-0816-y
57. Vinogradova K, Dibrov A, Myers G. Towards interpretable semantic segmentation via gradient-weighted class activation mapping (student abstract). AAAI. 2020;34(10):13943–4. https://doi.org/10.1609/aaai.v34i10.7244
58. Robison K, McGuire AM, Church GM. A comprehensive library of DNA-binding site matrices for 55 proteins applied to the complete Escherichia coli K-12 genome. J Mol Biol. 1998;284(2):241–54. https://doi.org/10.1006/jmbi.1998.2160 PMID: 9813115
59. Stewart AJ, Hannenhalli S, Plotkin JB. Why transcription factor binding sites are ten nucleotides long. Genetics. 2012;192(3):973–85. https://doi.org/10.1534/genetics.112.143370 PMID: 22887818
60. Rydenfelt M, Garcia HG, Cox RS 3rd, Phillips R. The influence of promoter architectures and regulatory motifs on gene expression in *Escherichia coli*. PLoS One. 2014;9(12):e114347. https://doi.org/10.1371/journal.pone.0114347 PMID: 25549361

61. Ruths T, Nakhleh L. Neutral forces acting on intragenomic variability shape the *Escherichia coli* regulatory network topology. Proc Natl Acad Sci U S A. 2013;110(19):7754–9. https://doi.org/10.1073/pnas.1217630110 PMID: 23610404

62. Kim B, Seo J, Jeon S, Koo J, Choe J, Jeon T. Why are saliency maps noisy? Cause of and solution to noisy saliency maps. CoRR. 2019;abs/1902.04893.

63. de Almeida BP, Schaub C, Pagani M, Secchia S, Furlong EEM, Stark A. Targeted design of synthetic enhancers for selected tissues in the *Drosophila* embryo. Nature. 2024;626(7997):207–11. https://doi.org/10.1038/s41586-023-06905-9 PMID: 38086418

64. Rafi AM, Nogina D, Penzar D, Lee D, Lee D, Kim N, et al. A community effort to optimize sequence-based deep learning models of gene regulation. Nat Biotechnol. 2024.

65. Sato K, Oide M, Nakasako M. Prediction of hydrophilic and hydrophobic hydration structure of protein by neural network optimized using experimental data. Sci Rep. 2023;13(1):2183. https://doi.org/10.1038/s41598-023-29442-x PMID: 36750742

66. Littmann M, Heinzinger M, Dallago C, Weissenow K, Rost B. Protein embeddings and deep learning predict binding residues for various ligand classes. Sci Rep. 2021;11(1):23916. https://doi.org/10.1038/s41598-021-03431-4 PMID: 34903827

67. Jones D, Kim H, Zhang X, Zemla A, Stevenson G, Bennett WFD, et al. Improved protein-ligand binding affinity prediction with structure-based deep fusion inference. J Chem Inf Model. 2021;61(4):1583–92. https://doi.org/10.1021/acs.jcim.0c01306 PMID: 33754707

68. Falo-Sanjuan J, Diaz-Tirado Y, Turner MA, Davis J, Medrano C, Haines J, et al. Targeted mutagenesis of specific genomic DNA sequences in animals for the in vivo generation of variant libraries. bioRxiv. 2024.

69. Chen S, Zhou Y, Chen Y, Gu J. fastp: an ultra-fast all-in-one FASTQ preprocessor. Bioinformatics. 2018;34(17):i884–90. https://doi.org/10.1093/bioinformatics/bty560 PMID: 30423086

70. Mann HB, Whitney DR. On a test of whether one of two random variables is stochastically larger than the other. Ann Math Statist. 1947;18(1):50–60. https://doi.org/10.1214/aoms/1177730491

71. Fay MP, Proschan MA. Wilcoxon-Mann-Whitney or t-test? On assumptions for hypothesis tests and multiple interpretations of decision rules. Stat Surv. 2010;4:1–39. https://doi.org/10.1214/09-SS051 PMID: 20414472

72. Kudo M, Toyama J, Shimbo M. Multidimensional curve classification using passing-through regions. Pattern Recognit Lett. 1999;20(11–13):1103–11. https://doi.org/10.1016/s0167-8655(99)00077-x