# DARE: Sequence-Structure Dual-Aware Encoder for RNA-Protein Binding Prediction

# DARE: Sequence-Structure Dual-Aware Encoder for RNA-Protein Binding Prediction

Luhan Shen*, Chengxin He*‡§, Haiying Wang†, Yuening Qu*, Lei Duan*‡§
*School of Computer Science, Sichuan University, Chengdu, China
†School of Computing, Ulster University, Northern Ireland, United Kingdom
‡Med-X Center for Informatics, Sichuan University, Chengdu, China
§Corresponding authors. Email: hechengxin@stu.scu.edu.cn, leiduan@scu.edu.cn

***Abstract*—Predicting RNA-protein binding sites helps to explore the mechanisms of the interaction between RNA and proteins. Numerous deep learning methods have been applied to predict RNA-protein binding sites. Some of these methods use only sequence information for prediction which could lose information about the topology. And there may be a loss of important information if the secondary structure features are simply represented as one-hot matrices. Furthermore, existing deep learning methods are usually based on convolutional neural networks for feature extraction, which tend to focus on local features. As for the information of the whole sequence, existing methods usually ignore global features. Therefore, we propose a novel deep learning model called DARE for RNA-protein binding sites prediction using both sequence and secondary structure information of RNA. DARE employs the secondary structure feature extraction module to capture the features of the RNA secondary structure and learn the topological information. Therefore, we design a local feature extraction module and a global feature integration module to capture the whole information of RNA. Thus we can achieve the purpose of complementary information. Extensive experiments demonstrate that DARE outperforms baselines. Our analysis of the case study further confirm the effectiveness of DARE.**

***Index Terms*—RNA-proteins binding prediction, RNA secondary structure, Transformer**

## I. Introduction

RNA-binding proteins (RBP) control processes like RNA processing and translation through specific sequence binding which are crucial for gene regulation and cellular functions [1, 2]. Mutations in these proteins can lead to diseases, such as amyotrophic lateral sclerosis caused by mutations in RBP-FUS and TDP43 [3]. Predicting RNA-protein binding sites is vital for comprehending these interactions.

The most direct way to explore RNA-protein binding sites is to identify them through biological experiments which are time-consuming and expensive. Machine learning methods [4–6] manually design features of RNA and build mathematical models for analysis. They tend to rely heavily on manually designed feature representations, which may limit their ability to capture the potential structural characteristics. In deep learning approaches, features are typically extracted from RNA sequence data [7, 8]. However, RNA's function is significantly influenced by its complex secondary structure, which is challenging to incorporate directly into neural networks. Some methods attempt to transform secondary structure information into one-hot matrices, risking loss of structural details. With the rise of graph neural networks (GCNs), they are employed to learn RNA's secondary structure information [9]. Both Convolutional Neural Networks (CNNs) and GCNs focus on node-to-neighbor relationships, rendering their output as local RNA features. However, global features are equally crucial, illustrated by the preferences of certain RBPs like *Vts1p*, which favor binding to loop sequences in RNA hairpins [10]. Existing deep learning methods struggle to capture these global features and long-range RNA connections. Considering this, the following questions need to be addressed when developing models for predicting RNA-protein binding sites:

- How to fully utilize the information of RNA to extract sequence and secondary structure features?
- How to capture global features of RNA so that models can consider both global and local information to improve the ability to predict RNA-protein binding sites?

To address these challenges, we propose a novel model, **DARE** (short for Sequence-structure Dual-Aware Encoder), to predict RNA-protein binding sites.

The main contributions of this work are as follows:

- We propose a novel deep learning model called DARE for RNA-protein binding sites prediction. By jointly applying Transformer and CNN, the global and local information of the RNA is complementary, thus improving the performance of RNA-protein interaction prediction.
- We use GCN to learn the secondary structural features of RNA. Then we fully integrate the semantic and topological structure information of RNA using subtle alignment and fusion strategies. This approach enriches the depth of the represented information and integrates the data.
- We conduct benchmark experiments to verify DARE's effectiveness and obtain motifs through case studies for further confirmation.

## II. Related Work

In this section, we will explore various aspects of research in RNA-Protein binding sites prediction field.

Early research heavily relied on biological experiments, but technical and cost limitations restricted their practicality for genome-wide predictions. Consequently, computational

This work was supported in part by the National Natural Science Foundation of China (61972268).

979-8-3503-3748-8/23/$31.00 ©2023 IEEE

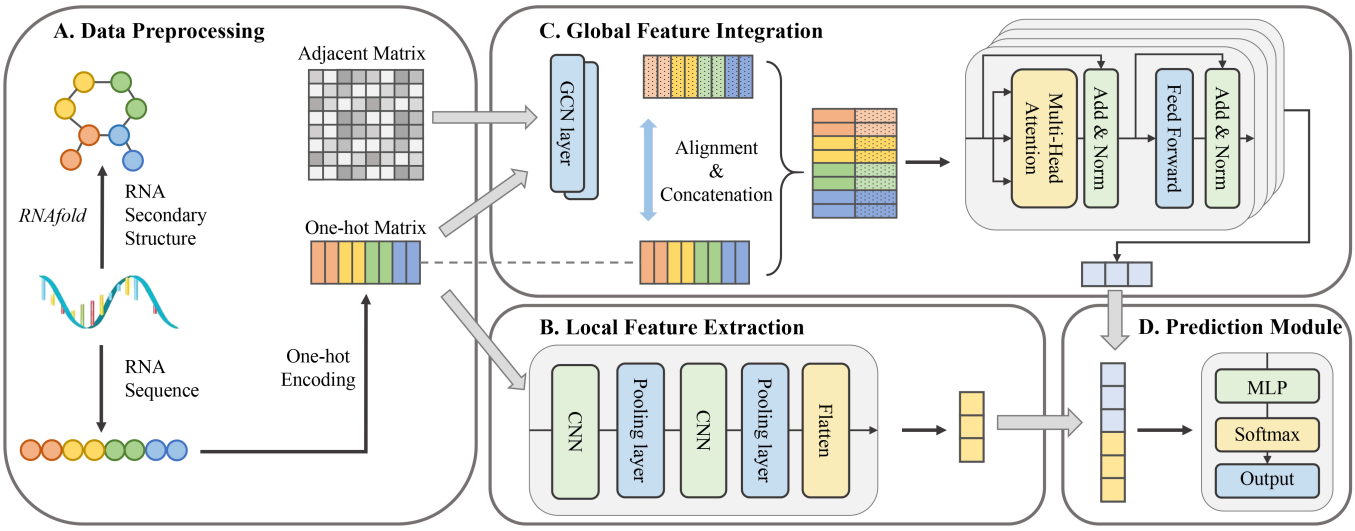


Fig. 1: Illustration of the overall framework of DARE: A) data preprocessing, B) local feature extraction, C) global feature integration, and D) prediction module.

approaches have become indispensable for accelerating predictions and improving accuracy.

The initial discovery of RNA motifs relied on the use of DNA motif search tools, which are limited in that they can only exploit the primary structure of RNA. Recently, machine learning methods constructed mathematical models to analyze various aspects of RNA features. Graphprot employed hypergraphs that encoded both sequence and secondary structure information, extracting features using graph kernels, and employing Support Vector Machine (SVM) for RNA binding sites prediction [4]. SARNAclust used a clustering algorithm that comprehensively studied the topology and annotation of RNA structures. It can generate motifs by converting the secondary structure into a graph input for the clustering model [5]. RNANetMotif leveraged graph theory knowledge and directly obtained abundant motifs through an enriched subgraph generation method [6]. However, traditional machine learning approaches often rely on representations of designed features manually, limiting their ability to capture complex potential essential RNA features.

DeepBind was the first to introduce CNNs for the prediction of RNA and protein binding sites based on the RNA sequence [7]. Deepnet-rbp integrated three dimensions of RNA information using deep belief networks (DBNs) [11]. iDeepE utilized both local multi-channel CNN and global CNN models to mine and integrate information [8]. RNASSR-net leveraged GCNs to learn RNA secondary structure representation and CNNs to learn RNA sequence features, thereby enhancing the prediction accuracy by incorporating both structural and sequence features [9]. DeepPN built a deep parallel neural network for predicting RNA-protein binding sites, using only the sequence information of the RNA [12].

## III. THE DESIGN OF DARE

In this section, we will provide a comprehensive overview of design details of DARE. The overall framework of DARE is illustrated in Figure 1. DARE consists of four main modules.

### A. Data Preprocessing

*1) RNA Sequence Coding:* To facilitate neural network computations, shorter sequences are padded with the symbol "$O$" to match the maximum length. For an RNA sequence $S = \langle s_1, s_2, s_3, \ldots, s_n \rangle$ containing $n$ nucleotides, it is represented as a one-hot matrix denoted by $\tilde{\boldsymbol{S}} = [\tilde{\boldsymbol{s}}_1, \tilde{\boldsymbol{s}}_2, \tilde{\boldsymbol{s}}_3, \ldots, \tilde{\boldsymbol{s}}_n] \in \mathbb{R}^{4\times|S|}$. When $s_i =$ "$O$", $\tilde{\boldsymbol{s}}_i$ is different from the one-hot encoding and is defined as $[0.25, 0.25, 0.25, 0.25]$.

*2) RNA Secondary Structure Construction:* We employ the biological tool *RNAfold* to analyze the RNA folding. The outcome of *RNAfold* [13] is a set of multiple potential secondary structures corresponding to the RNA sequence $S$. Edges may exist between two nucleotides in specific structures, but not in others. Therefore, we chose to use the entire set of potential structures instead of only the one with the minimum free energy. Finally, we can obtain the probability adjacency matrix $\boldsymbol{A}_{n\times n}$. Likewise, we employ a padding strategy to fill isolated empty nodes, thereby extending the number of nodes in the graph to align with the length of the RNA sequence. Thus, we can obtain the RNA secondary structure, and define it as $G$.

### B. Local Feature Extraction

In the local feature extraction module, we employ CNNs to extract local features from RNA sequences $\tilde{\boldsymbol{S}}$. By leveraging convolution, ReLU activation functions, and max-pooling procedures, local feature extraction module adeptly captures essential local structures and features embedded within the RNA sequences. The output of the local feature extraction module is denoted as $\boldsymbol{h}_c$:

$$\boldsymbol{h}_c = F_{flatten}\left(Pool\left(ReLU\left(Conv\left(S\right)\right)\right)\right) \tag{1}$$

where $Conv(\cdot)$ to represent the convolution operation; $ReLU(\cdot)$ is the non-linear function; $Pool(\cdot)$ is the maximum pooling method; $F_{\text{flatten}}$ is used to transform the pooled feature maps into a flat vector suitable.

### C. Global Feature Integration

*1) Secondary Structure Feature Extraction:* The module employs GCN to capture features from the RNA secondary structure $G$ through message passing. The graph convolution process can be mathematically expressed as follows:

$$\boldsymbol{H}^{(l)} = ReLU\left(\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\boldsymbol{H}^{(l-1)}\boldsymbol{W}^{(l)}\right) \tag{2}$$

where $\boldsymbol{H}^{(l)}$ is the feature representation at the $l$-th layer of the GCN. $\tilde{\boldsymbol{A}} = \boldsymbol{A} + \boldsymbol{I}_N$. $\tilde{\boldsymbol{D}}$ is the degree matrix of $\tilde{\boldsymbol{A}}$. $\boldsymbol{W}^{(l)}$ is the trainable parameter matrix. Then we can acquire the feature vector $\boldsymbol{h}_1$ generated by the GCN.

*2) Feature Alignment:* We align $\boldsymbol{h}_1$ with the order of RNA sequences, ensuring that each node's representation corresponds to the respective nucleotide's position in the sequence, which can be represented as $\boldsymbol{h}_2$. Concurrently, we represent the nucleotide information as one-hot embeddings $\tilde{\boldsymbol{S}}$ and seamlessly concatenate it with $\boldsymbol{h}_2$:

$$\boldsymbol{h}_g = \boldsymbol{h}_2 \oplus \tilde{\boldsymbol{S}} \tag{3}$$

where $\oplus$ represents the concatenation operation.

*3) Transformer-based Feature Integrator:* We feed the aligned feature vector $\boldsymbol{h}_g$ as input to the Transformer. To enhance the model's representational power, we adopt multi-head attention. Meanwhile, we introduce residual connections and layer normalization in each sub-layer. The final output representation of each sub-layer is given by

$$\boldsymbol{h}_3 = LN(\boldsymbol{h}_g + MultiHead(\boldsymbol{h}_g)) \tag{4}$$

The modules incorporates feed-forward neural network (*FFN*) layers to enhance its nonlinear modeling capacity.

$$FFN(\boldsymbol{h}_3) = ReLU(\boldsymbol{h}_3 \cdot \boldsymbol{W}_1 + \boldsymbol{b}_1) \cdot \boldsymbol{W}_2 + \boldsymbol{b}_2 \tag{5}$$

where $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ are learnable parameter matrices, and $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ are learnable bias vectors.

Finally, the representation of RNA's global features is:

$$\boldsymbol{h}_t = LN(\boldsymbol{h}_3 + FFN(\boldsymbol{h}_3)) \tag{6}$$

### D. Prediciton Module

We concatenate the outputs of the Transformer and CNN. Then We employ a MLP with activation functions for binary classification and the loss is cross-entropy loss:

$$\boldsymbol{h}_{final} = \boldsymbol{h}_c \oplus \boldsymbol{h}_t \tag{7}$$

$$\hat{y} = Softmax\left(ReLU\left(MLP\left(\boldsymbol{h}_{final}\right)\right)\right) \tag{8}$$

$$Loss = -ylog\left(\hat{y}\right) + \left(1-y\right)log\left(1-\hat{y}\right) \tag{9}$$

## IV. EXPERIMENTS AND RESULT DISCUSSION

### A. Experimental Settings

*1) Datasets:* The dataset consists of 24 RBP binding site datasets derived from HITS-CLIP, PAR-CLIP, and iCLIP experiments which is originally used in GraphProt [4].

*2) Evaluation Metrics and Baselines:* We used AUC as evaluation metric. Baselines include GraphProt [4], Deepnet-rbp [11], iDeepE [8], RNASSR [9] and DeepPN [12].

TABLE I: Performance of our model and other baselines. The best results are highlighted in bold.

| RBP | Graph-Prot | Deepnet-rbp | iDeepE | RNASSR | DeepPN | Ours |
|---|---|---|---|---|---|---|
| ALKBH5 | 68.0% | 71.4% | 75.8% | 77.1% | 66.0% | **80.1%** |
| C17ORF85 | 80.0% | 82.0% | 83.0% | 88.9% | 83.7% | **89.6%** |
| C22ORF28 | 75.1% | 79.2% | 83.7% | 86.5% | 78.5% | **86.7%** |
| CAPRIN1 | 85.5% | 83.4% | 89.3% | 92.4% | 88.6% | **92.5%** |
| AGO2 | 76.5% | 80.9% | 88.4% | 89.0% | 86.8% | **89.1%** |
| ELAVL1H | 95.5% | 96.6% | 97.9% | 98.3% | 97.8% | **98.7%** |
| SFRS1 | 89.8% | 93.1% | 94.6% | 95.3% | 93.6% | **95.4%** |
| HNRNPC | 95.2% | 96.2% | 97.6% | 98.0% | 97.7% | **98.2%** |
| TDP43 | 87.4% | 87.6% | 94.5% | 95.2% | 93.6% | **95.4%** |
| TIA1 | 86.1% | 89.1% | 93.7% | 94.8% | 92.8% | **95.0%** |
| TIAL1 | 83.3% | 87.0% | 93.4% | 94.6% | 92.6% | **94.7%** |
| Ago1-4 | 89.5% | 88.1% | 91.5% | 93.7% | 91.2% | **93.8%** |
| ELAVL1B | 93.5% | 96.1% | 97.1% | 98.0% | 97.6% | **98.1%** |
| ELAVL1A | 95.9% | 96.6% | 96.4% | 97.7% | 96.7% | **97.9%** |
| EWSR1 | 93.5% | 96.6% | 96.9% | 97.0% | 95.4% | **97.1%** |
| FUS | 96.8% | 98.0% | 98.5% | 98.6% | 97.7% | **98.7%** |
| ELAVL1C | 99.1% | **99.4%** | 98.8% | 99.1% | **99.4%** | 99.2% |
| IGF2BP1-3 | 88.9% | 87.9% | 94.7% | 97.0% | 92.8% | **97.1%** |
| MOV10 | 86.3% | 85.4% | 91.6% | **94.0%** | 90.4% | **94.0%** |
| PUM2 | 95.4% | 97.1% | 96.7% | 97.9% | 95.2% | **98.1%** |
| QKI | 95.7% | **98.3%** | 97.0% | 98.1% | 97.5% | 97.6% |
| TAF15 | 97.0% | 98.3% | 97.6% | 98.4% | 97.4% | **98.7%** |
| PTB | 93.7% | **98.3%** | 94.4% | 94.9% | 93.8% | 95.0% |
| ZC3H7B | 82.0% | 79.6% | 90.7% | 91.7% | 89.8% | **92.1%** |
| Average | 88.7% | 90.2% | 93.1% | 94.4% | 91.9% | **94.9%** |

*3) Experimental Setup:* We ran experiments on a Ubuntu server with NVIDIA GTX 3090 GPU with memory of 24 GB. We utilized a learning rate of 0.001, and a batch size of 128, and trained the model for 300 epochs. The original dataset includes both a training and a test set. Following by [9], we randomly split the training dataset: 90% for training and 10% for validation. Our source code is available on Github[1].

### B. The Effectiveness of DARE

The performance is detailed in Table I. DARE outperforms baselines with an average AUC of 94.9% across 24 datasets. DARE achieves a 3% improvement over the second-best on the ALKBH5 dataset and 0.7% on the C17ORF85 dataset. The results can demonstrate DARE's advantage in small data set.

Comparing baselines in Table I, GraphProt, relying more on hand-crafted features, performs the worst. DeepPN and iDeepE, although deep learning models, are surpassed by RNASSR and DARE, both of which consider RNA secondary structure. DARE incorporate global and local RNA features, leads to its superior performance. While Deepnet-rbp accounts for primary, secondary, and tertiary RNA structure, it performs well on some datasets but poorly overall, potentially due to inaccuracies in secondary and tertiary structure prediction.

In summary, DARE excels on 21 datasets and demonstrates strong performance on others, affirming its efficacy in enhancing RNA-protein binding site prediction.

### C. Ablation Studies

In order to evaluate the contribution of each component of DARE, we conduct an ablation study, the results of which are

[1]https://github.com/scu-kdde/Bioinfo-DARE-2023

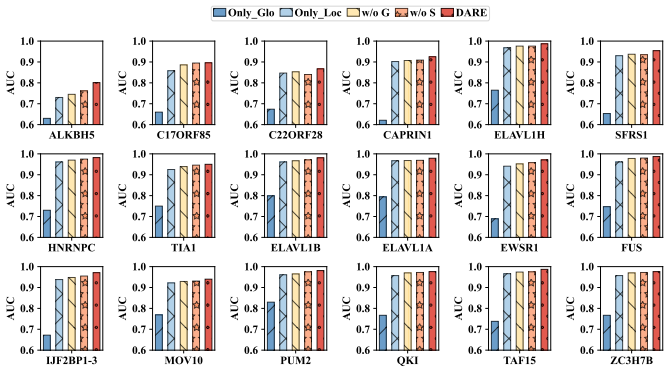


Fig. 2: Analysis of the impact of different variants on DARE.

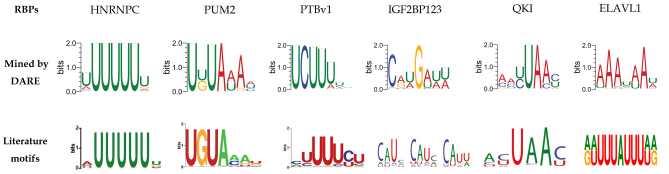


Fig. 3: Samples of detected motifs.

shown in Figure 2. Specifically, these variants are constructed as follows: **Only_Glo** denotes that only the global feature extraction module is retained; **Only_Loc** denotes that only the local feature extraction module is retained; **w/o G** denotes the removal of the GCN's module for secondary structural feature extraction; **w/o S** denotes the removal of the module for the nucleotide semantic information that is spliced with the GCN.

The results reveal that removing the localized feature modules has the most substantial impact on performance, while the removal of other components also exerts some influence, underscoring the significance of each component.

### *D. Case Study*

Following DeepBind [7], we extract motifs using neural network parameters, considering convolutional filters as "motif detectors". To fully synthesize the information, we combine the outputs of Transformer and CNN to extract k-mer candidate motifs. We employ the MEME tool to visualize motifs. Figure 3 shows motifs extracted for HNRNPC, PUM2, PTBv1, IGF2BP123, QKI, and ELAVL1 proteins. Our predicted motifs closely align with experimentally validated results from the CISBP-RNA database [14], reinforcing DARE's biological interpretability and predictive accuracy.

## V. Conclusion

In this study, we introduce DARE, a deep learning model for predicting RNA and protein binding sites. By leveraging the capabilities of both CNN and Transformer, DARE effectively captures global and local RNA information, enhancing prediction accuracy and interpretability. We validate the effectiveness of DARE through numerous experimental results. Additionally, we evaluate the model's biological interpretability by generating motif plots, aligning well with existing literature. In future work, we consider integrating additional bioinformatic data, such as RNA 3D structure and chemical modifications, to enhance predictive performance and provide a more comprehensive understanding of RNA-protein binding sites.

## References


[1] S. Gerstberger, M. Hafner, and T. Tuschl, "A census of human RNA-binding proteins," *Nature Reviews Genetics*, vol. 15, no. 12, pp. 829–845, 2014.

[2] C. He, L. Duan, H. Zheng, and et al., "Graph convolutional network approach to discovering disease-related circrna-mirna-mrna axes," *Methods*, vol. 198, pp. 45–55, 2022.

[3] I. R. Mackenzie, R. Rademakers, and M. Neumann, "TDP-43 and FUS in amyotrophic lateral sclerosis and frontotemporal dementia," *The Lancet Neurology*, vol. 9, no. 10, pp. 995–1007, 2010.

[4] D. Maticzka, S. J. Lange, F. Costa, and et al., "Graphprot: Modeling binding preferences of RNA-binding proteins," *Genome Biology*, vol. 15, no. 1, p. R17, 2014.

[5] I. Dotu, S. I. Adamson, B. Coleman, and et al., "SARNAclust: Semi-automatic detection of RNA protein binding motifs from immunoprecipitation data," *PLOS Computational Biology*, vol. 14, no. 3, pp. 1–25, 2018.

[6] H. Ma, H. Wen, Z. Xue, and et al., "RNANetMotif: Identifying sequence-structure RNA network motifs in RNA-protein binding sites," *PLOS Computational Biology*, vol. 18, no. 7, pp. 1–27, 2022.

[7] B. Alipanahi, A. Delong, M. T. Weirauch, and et al., "Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning," *Nature Biotechnology*, vol. 33, no. 8, pp. 831–838, 2015.

[8] X. Pan and H.-B. Shen, "Predicting RNA–protein binding sites and motifs through combining local and global deep convolutional neural networks," *Bioinformatics*, vol. 34, no. 20, pp. 3427–3436, 2018.

[9] Z. Liu, F. Luo, and B. Du, "RNA secondary structure representation network for RNA-proteins binding prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 1, 2021, pp. 362–370.

[10] T. Aviv, Z. Lin, G. Ben-Ari, and et al., "Sequence-specific recognition of RNA hairpins by the SAM domain of Vts1p," *Nature Structural & Molecular Biology*, vol. 13, no. 2, pp. 168–176, 2006.

[11] S. Zhang, J. Zhou, H. Hu, and et al., "A deep learning framework for modeling structural features of RNA-binding protein targets," *Nucleic Acids Research*, vol. 44, no. 4, pp. e32–e32, 2015.

[12] J. Zhang, B. Liu, Z. Wang, and et al., "DeepPN: A deep parallel neural network based on convolutional neural network and graph convolutional network for predicting RNA-protein binding sites," *BMC Bioinformatics*, vol. 23, no. 1, p. 257, 2022.

[13] R. Lorenz, S. H. Bernhart, C. Höner zu Siederdissen, and et al., "Viennarna package 2.0," *Algorithms for Molecular Biology*, vol. 6, pp. 1–14, 2011.

[14] D. Ray, H. Kazan, K. Cook, and et al., "A compendium of RNA-binding motifs for decoding gene regulation," *Nature*, vol. 499, no. 7457, pp. 172–177, 2013.