Title: Protein RNA Binding Site Prediction Using Deep Learning

Author: Adnan Muneer

Course: COE 589: Special Topics in Computer Systems and Applications

---

1. Introduction:

RNA-binding proteins (RBPs) play a crucial role in the regulation of RNA after transcription. A key aspect of post-transcriptional gene regulation involves identifying RBP binding sites and understanding their interactions. Many computational approaches have been employed to model RBP binding preferences, yet obtaining a complete three-dimensional representation remains a significant challenge. Recently, deep learning has emerged as a powerful tool for predicting RBP binding sites and modeling structural preferences. By leveraging deep learning, researchers can analyze complex structural features embedded in RNA sequences and profiles, helping to identify new binding sites and motifs. One of the main challenges lies in fully characterizing the three-dimensional structure of RBPs' binding targets. Deep learning techniques can extract and predict these intricate features, offering new insights into binding site preferences. Among the datasets used for RNA-protein binding site predictions are CLIPSEQ_ELAVL1 and ALKBH5_Baltz2012. The CLIPSEQ_ELAVL1 dataset, derived from CLIP-seq experiments, identifies RNA regions bound by ELAVL1 (HuR), a protein that stabilizes mRNAs, particularly those with AU-rich elements (AREs) in their 3' UTRs. This dataset is valuable for studying post-transcriptional gene regulation, especially in contexts like cancer and immune responses. Meanwhile, the ALKBH5_Baltz2012 dataset centers on ALKBH5, an RNA demethylase responsible for removing N6-methyladenosine (m6A) from mRNAs, offering insights into epitranscriptomic regulation with potential relevance to cancer and reproductive biology.

---

2. **Model Architectures:**

1. **Model 1: CLIPSEQ_ELAVL1** trained from the Scratch. For the first model (CLIPSEQ_ELAVL1), we utilized a Convolutional Neural Network (CNN) with essential layers such as Conv2D and MaxPooling. A dropout layer was implemented to minimize overfitting, while a Flatten layer was used to convert the two-dimensional data into one-dimensional format for further processing. The Dense layer acted as a fully connected neural network (MLP). This architecture was chosen primarily to handle the issue of overfitting that often arises with artificial neural networks (ANNs) during training. The input data, arranged in a 200 x 4 matrix format, was well-suited for CNNs, which excel in processing spatially correlated data within matrices.

2. **Model 2**: **ALKBH5_Baltz2012** trained from the Scratch. A CNN architecture was also used to develop the second model (ALKBH5_Baltz2012). The layers in this model included a sequence of Conv2D followed by MaxPooling and Dropout, and another Conv2D paired with Dropout, before Flattening the data. A dropout layer was added at multiple points to reduce overfitting. Like Model 1, this model also employed a 200 x 4 matrix for input data, leveraging CNN's strength in capturing spatial relationships within such structured data.

3. **Model 3: Fine-tuning CLIPSEQ_ELAVL1 on ALKBH5_Baltz2012** The third model was constructed by fine-tuning the pre-trained CLIPSEQ_ELAVL1 model on the ALKBH5_Baltz2012 dataset. We used the previously trained Model 1 as a base, adding extra layers to further refine the model. The output from the Dense layer of the base model was passed to new layers, including Conv2D, MaxPooling, Dropout, and Flatten, followed by a final Dense layer. All layers from the base model were set as trainable, allowing the entire network, including the pre-trained sections, to be updated during training. This approach was also aimed at mitigating overfitting and was applied to data structured in a 200 x 4 matrix, an arrangement that suits CNNs for their proficiency in processing matrix data.

All three models were compiled using the Adam optimizer, with categorical cross-entropy as the loss function. Evaluation metrics included accuracy, precision, recall, and AUC score.

---

1. **Results during Training**

	Training				Validation			
	Precision	***Accuracy***	***AUC***	***Recall***	***Precision***	***Accuracy***	***Recall***	***AUC***
Model 1	83.67	94.75	98.56	74.84	83.90	94.71	74.53	98.81
Model 2	81.28	97.88	99.66	72.19	58.30	58.35	55.19	54.62
Model 3	84.63	99.45	96.34	74.97	58.52	58.14	55.64	61.29

2. Model Evaluation:

Evaluation on Test Data				
	Precision	***Recall***	***Accuracy***	***AUC***
Model 1	83.06	73.66	92.21	97.19
Model 2	57.29	54.10	53.38	58.26
Model 3	58.74	55.37	57.25	60.78

5. Additional Analysis (if applicable):

During the project, a notable insight was the stark contrast in performance between training and test data across models, particularly with Model 2, which exhibited high training metrics but a drastic decline in test accuracy (from 97.88% to 53.38%). After Fine tuning of Model 1, we got considerably good performance. Model 1 generalized very well (See Fig 1), However overfitting can be seen for Model 2 and Model 3 (See figure2 and figure 3). The area under the curve (AUC) of all three models are shown in Figure 4, Figure 5, and Figure 6.

Figure 1: Training Vs Validation accuracy

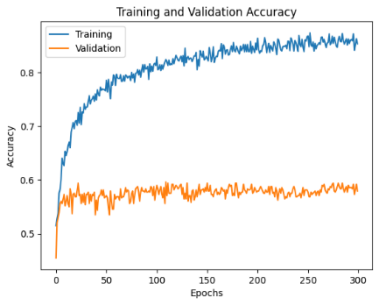

Figure 2: Training Vs Validation accuracy

Figure 3: Training Vs Validation accuracy

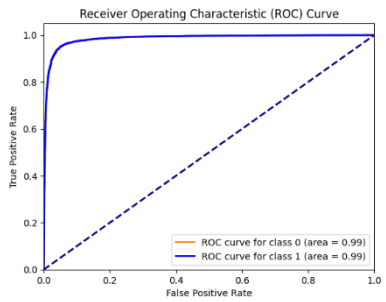

Figure 4: AUC Curve

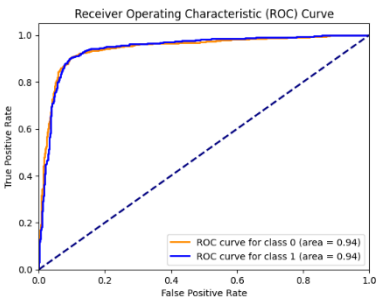

Figure 5: AUC Curve

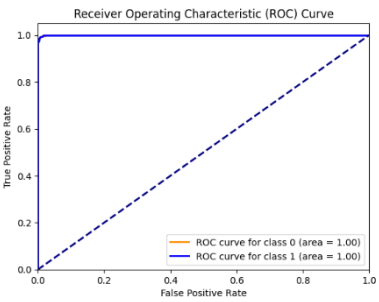

Figure 6: AUC Curve

6. Conclusion:

In this evaluation, **Model 1** emerged as the best performer, exhibiting high accuracy (92.21%), precision (83.06%), recall (73.66%), and AUC (97.19%) across training, validation, and test data, indicating strong generalization capabilities. In contrast, **Model 2** and **Model 3** demonstrated significant overfitting, with Model 2 experiencing a dramatic drop in test accuracy to 53.38% and Model 3 showing only marginally better performance at 57.25% after finetuning.

References

[1] Lebedeva, S., Jens, M., Theil, K., Schwanhäusser, B., Selbach, M., Landthaler, M., & Rajewsky, N. (2011). Transcriptome-wide analysis of regulatory interactions of the RNA-binding protein HuR. *Molecular cell*, *43*(3), 340-352.

[2] Baltz, A. G., Munschauer, M., Schwanhäusser, B., Vasile, A., Murakawa, Y., Schueler, M., ... & Landthaler, M. (2012). The mRNA-bound proteome and its global occupancy profile on protein-coding transcripts. *Molecular cell*, *46*(5), 674-690.

[3] *Zhang, S., Zhou, J., Hu, H., Gong, H., Chen, L., Cheng, C., & Zeng, J. (2016). A deep learning framework for modeling structural features of RNA-binding protein targets. Nucleic acids research, 44(4), e32-e32.*