Prediction and functional characterization of transcriptional activation domains

Saloni Mahatma*
Plant and Microbial Biology Department
North Carolina State University
Raleigh, USA
smahatm@ncsu.edu

Lisa Van den Broeck*
Plant and Microbial Biology Department
North Carolina State University
Raleigh, USA
0000-0003-0226-0757

Nicholas Morffy
Department of Biology
Duke University
Durham, USA
0000-0003-3170-2032

Max V Staller
Center for Computational Biology
University of California, Berkeley
Berkeley, USA
0000-0001-9094-5697

Lucia C Strader
Department of Biology
Duke University
Durham, USA
0000-0002-7600-7204

Rosangela Sozzani
Plant and Microbial Biology Department
North Carolina State University
Raleigh, USA
0000-0003-3316-2367

***Abstract*—Gene expression is induced by transcription factors (TFs) through their activation domains (ADs). However, ADs are unconserved, intrinsically disordered sequences without a secondary structure, making it challenging to recognize and predict these regions and limiting our ability to identify TFs. Here, we address this challenge by leveraging a neural network approach to systematically predict ADs. As input for our neural network, we used computed properties for amino acid (AA) side chain and secondary structure, rather than relying on the raw sequence. Moreover, to shed light on the features learned by our neural network and greatly increase interpretability, we computed the input properties most important for an accurate prediction. Our findings further highlight the importance of aromatic and negatively charged AA and reveal the importance of unknown AA properties. Taking advantage of these most important features, we used an unsupervised learning approach to classify the ADs into 10 subclasses, which can further be explored for AA specificity and AD functionality. Overall, our pipeline, relying on supervised and unsupervised machine learning, shed light on the non-linear properties of ADs.**

Keywords—Multilayer neural network, transcriptional activation domains, feature importance, unsupervised clustering

I. Introduction

Transcription factors (TFs) can promote gene expression by binding to the DNA and subsequently recruiting transcriptional machinery. Generally, TFs bind short DNA sequences called motifs through conserved DNA-binding domains. In addition, TFs contain repression or activation domains (ADs) that bind corepressors or coactivators, respectively. Such interactions recruit the transcription machinery to initiate transcription and/or lead to chromatin-modifying activity.

Repression domains and ADs are less conserved and more challenging to identify within the TF's protein sequence. A major limitation in predicting ADs is that they are intrinsically disordered regions, which have no specific three-dimensional (3D) structure and thus conformational heterogeneity. Currently, due to the lack of conservation, *in silico* prediction of ADs is largely missing.

In the last decade, neural networks have been deployed in biological sciences to address various challenges [1], [2]. A multitude of prediction methods, including neural networks, have been developed to predict intrinsically disordered regions within protein sequences [3]. Such methods indicate which protein region contains a potential AD. However, since intrinsically disordered regions are highly abundant in eukaryotic proteins [4], these methods are not exclusively predicting ADs leading to ambiguous and inaccurate AD predictions. Recently, an AD prediction neural network was designed and trained using a large-scale random peptide dataset [5]. However, this neural network captures one subclass of activation domains and, as a consequence, less than 65% of an independent set of experimental validated ADs were correctly identified [6]. As such, improved prediction models are needed to gain a deep molecular understanding of AD activity.

Here, we designed a rigorous neural network to predict ADs based on amino acid chain properties and intrinsically disordered descriptors rather than the raw sequence. Leveraging a random peptide dataset, we obtained 91.95% accuracy on the test dataset. To further gain a deeper understanding of the molecular actions of ADs, we used what our neural network learned to identify the most important features that correlate with AD predictions. These features include known features such as charged residues and the fraction of negative residues, and novel attributes, including the number of valine and glycine residues. These most important and predictive features were used to classify ADs into 10 different subtypes using an AI-

*Denotes equal contribution. This work was supported by the National Science Foundation (NSF) (PGRP BIO-2112058, IOS-2112056, 2112057).

based unsupervised approach. Overall, our AD predictions, key important sequence features, and AD type classification greatly deepens our understanding of the TF function.

II. Methods

A. Feature calculations

We used a large-scale balanced dataset containing random peptides, of which 37923 and 37922 were experimentally confirmed as ADs and non-ADs, respectively [5]. This experimental dataset contained amino acid sequences of length 30 and their respective AD score, which represents their ability to activate transcription.

We calculated a feature matrix for each AA sequence of size 26x41 that captures the amino acid properties, sequence, and structure of each 30 AA sequence. AD function depends on sequence properties, such as hydrophobic residues, therefore we included 11 AA properties. These 11 properties were calculated by counting the number of amino acids of each side chain class (Table 1) for a window of size 5 across the entire sequence length with a stride of 1 resulting in a 26x11 matrix (matrix-1).

TABLE I. Amino acid properties that are used as input data for our neural network.

Side Chain	Amino Acids
Aliphatics	I, V, L, A
Aromatics	W, F, Y
Polars	R, K, D, E, Q, N, Y
Branching	V, I, T
Charged	K, R, H, D, E
Negatives	D, E
Phosphorylatable	S, T, Y
Hydrophobics	W, F, L, V, I, C, M
Positives	K, R, H
Sulfur containing	M, C
Tinys	G, A, S, P

A previous study [5] has shown that the sequences having AD function have no conserved sequence or structure. To incorporate low dependency of sequence structure on the AD function, we calculated the number of occurrences of each AA in a window of size 5 across the entire sequence length with a stride of 1 resulting in a 26x20 matrix. We concatenated this matrix horizontally resulting in matrix-1 resulting in 26x31. An additional 8 properties were calculated using localcider [8] across the same window size and stride to calculate matrix-1 (Table 2). This matrix was concatenated horizontally with matrix-1 resulting in a 26x39 matrix. Last, we calculated the final 2 properties (kappa and omega) associated with intrinsically disordered proteins for the entire 30 AA sequence without using a sliding window approach. Since it was not calculated using a window of size 5, it has a dimension of 1x2. To merge these 2 properties with matrix-1 of size 26x39, we duplicated the values of 1x2 values for 26 rows to obtain a 26x2 matrix and horizontally concatenated it with matrix-1, creating the resultant feature matrix of a sequence of size 26x41.

TABLE II. Intrinsically disordered properties that are used as input for the neural network.

Properties	Meaning
Positive	Fraction of residues that are positively charged
Hydropathy	Mean hydropathy
Hydropathy ww	Wimley and White hydropathy
NCPR	Net charge per residue
FCR	Fraction of charged residues
Charge	The absolute mean net charge
Negative	Fraction of residues that are negatively charged
Promoting	Fraction of residues predicted to be ‘disorder promoting’
Omega	Patterning between charged/proline and all other residues
Kappa	Distribution of oppositely charged residues

Using the window size 5 and stride 1, we also included a predicted secondary structure from AlphaFold [7] as one of the features to create a second matrix of size 26x42 (matrix-2). AlphaFold provides residue-by-residue confidence scores of the predicted secondary structure of a protein sequence. A very low confidence score is generally associated with a lack of secondary structure. We approximated the confidence score in each window by calculating the mean confidence score in that window. We concatenated this feature in matrix-1 horizontally resulting in a 26x42 size for matrix-2. This feature matrix was used for further analysis.

We scaled each property (z-score standardization) in 26 windows over the entire dataset to make the mean of features equal to 0 and unit standard deviation. The standardization was followed by min-max normalization so that the values of properties lie between 0 and 1. The scaled input was used for training, validation, and testing of the neural network. We used a training-validation-test split ratio of 70:20:10.

B. Neural network architecture

To classify protein sequence as an AD, we used a neural network architecture that contains: (i) two convolutional neural network layers to extract and compress sequence information from the input, (ii) an attention layer to selectively focus on the features that are more important for the prediction, (iii), two bidirectional long short-term memory (biLSTM) layers to capture the interdependence of the sub-sequences in a sequence, and (iv) a dense layer to connect to the output layer (Fig. 1).

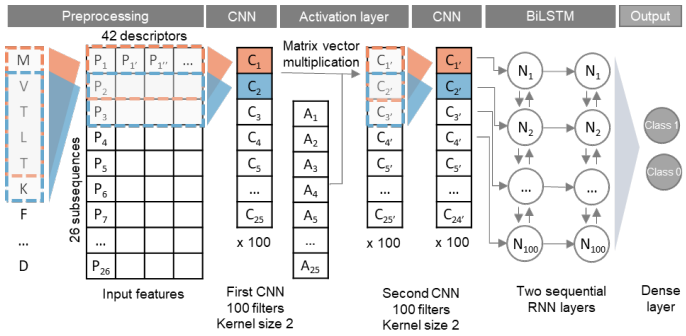

Fig. 1. Neural network architecture.

To evaluate the performance of the neural network, the accuracy, precision, recall, and F1 score of each class individually were calculated as follows:

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total predictions}} \quad (1)$$

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \quad (2)$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad (3)$$

$$\text{F1 score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

To identify the optimal architecture with the highest F1 score, the neural network's performance on the test dataset was evaluated for different activation functions of the convolutional layer, the inclusion of an attention layer at different positions, and the inclusion of AlphaFold prediction probabilities in the input feature matrix. The neural networks were consistently trained for 20 epochs upon comparison of the best F1 score (Table 3). The neural network with the highest F1 score on the test dataset was chosen for further analyses, including benchmarking. The test dataset was not used during training or validation making it unbiased for benchmarking.

TABLE III. THE NEURAL NETWORK'S PERFORMANCE WITH DIFFERENT ARCHITECTURES, PARAMETERS, AND INPUTS USING TEST DATASET.

Activation function	**F1-score**
Sigmoid	0.9115
TanH	0.9127
Relu	0.9149
Gelu	0.9195
Feature Matrix	**F1-score**
26 x 41	0.9161
26 x 42 (including AlphaFold)	0.9195
NN Architecture	**F1-score**
Conv1D-Dropout-Conv1D-Dropout-BiLSTM-BiLSTM-Dense	0.9112
Conv1D-**Attention**-Dropout-Conv1D-Dropout-BiLSTM-BiLSTM-Dense	0.9159
Conv1D-Dropout-**Attention**-Conv1D-Dropout-BiLSTM-BiLSTM-Dense	0.9118
Conv1D-Dropout-Conv1D-**Attention**-Dropout-BiLSTM-BiLSTM-Dense	0.9170
Conv1D-Dropout-Conv1D-Dropout-**Attention**-BiLSTM-BiLSTM-Dense	0.9179
Conv1D-Dropout-Conv1D-Dropout-BiLSTM-**Attention**-BiLSTM-Dense	0.9172

Both L1 and L2 regularization were included into the first convolutional layer of the model, controlling the model complexity. The following hyperparameters were used for upon testing the different neural network architectures, parameters, and inputs, as well as for the final network (Table 4):

TABLE IV. HYPERPARAMETERS USED FOR THE FINAL NEURAL NETWORK.

Hyperparameters	**Values**
Optimizer	Adam
Learning rate	1e-3
Dropout probabilities	0.3
Kernel rate	2
Filters	100
Batch size	64

C. *Feature importance*

To find the importance of each 42 calculated properties in the feature matrix for model prediction, we used SHapley Additive exPlanations (SHAP) [9]. We used GradientExplainer to obtain the SHAP value associated with each feature. SHAP values indicate the impact of features on the model output. We performed this analysis on both classes separately using the feature matrix of sequences that were classified with the highest 2% probability score.

D. *Unsupervised clustering*

To find the correspondence between the classes and potential subclusters, we performed a Principal Component Analysis (PCA) with 10 components and plotted it using T-distributed Stochastic Neighborhood Embedding (t-SNE) for both classes. To identify clusters within the sequences classified to have AD function, we used unsupervised clustering. Specifically, we first selected the most important features identified by SHAP as those that scored above 50% of the top scoring feature, which led to 12 features. These 12 features were calculated for the entire 30 AA sequence without using a sliding window approach. Next, we performed a PCA with 3 components, which equals 70.216% of the variance in the dataset, on these most important features. K-means clustering was performed on the three components of PCA to obtain 10 clusters and plotted it using t-SNE with a *perplexity* of 250 and *early exaggeration* of 4 for class 1 (sequence classified as AD). We used a higher value of perplexity as it defines a clearer shape and distance between the cluster.

III. Results

A. Designing a neural network for predicting activation domains

To predict AD activity, the amino acid properties, such as side chain properties, and disordered sequences properties, such as charge and hydrophobicity patterning, are required. Thus, to accurately predict ADs, we opted to calculate a feature matrix, which consists of a total of 42 calculated properties (see Methods). To explore hidden patterns and the structure of our feature matrix prior to training, we performed a PCA followed by t-SNE plotting (Fig. 2A-C). The two classes were not linearly separable in the t-SNE plot, which substantiates the motivation to implement a neural network to learn complex features to predict AD function (Fig. 2A-C). To this end, we used the calculated feature matrix as the input dataset (see Methods). Since amino acid sequences are sequential data where the function depends on the properties but also the position of the amino acids, we employed convolutional and Bi-LSTM layers in our neural network architecture (see Methods).

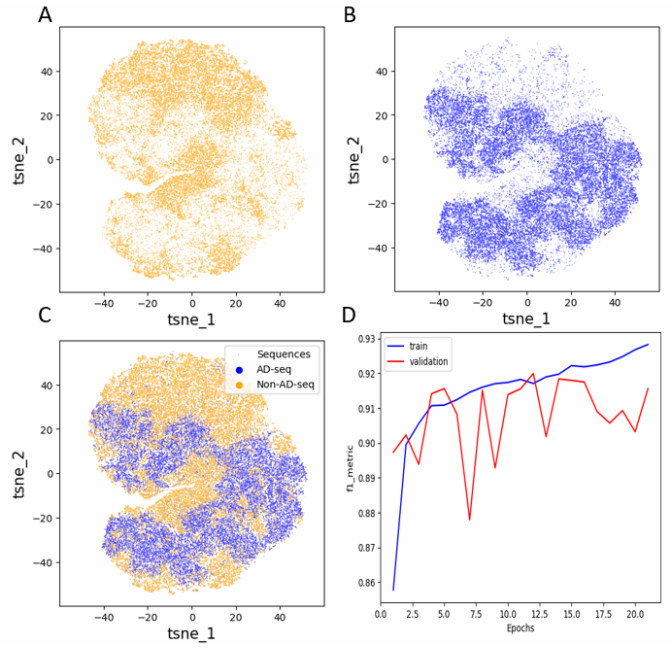

Fig. 2. Features as input for a sequential multi-layer neural network. (A-C) The 42 quantified properties were clustered using principal component analysis (PCA) and t-SNE for class 1 (A), class 0 (B), and both (C). (D) Neural network performance F1 score.

We trained neural networks with two different sets of feature matrices, where one set contains an additional feature that represents the secondary structure of the sequence, an indicator of the presence of an intrinsically disordered region [7], [10]. In addition, we tested different positions of the activation layer in our neural network architecture and different activation functions used during training (see Methods). We compared performance and selected the neural network architecture and hyperparameters with the highest F1 score. We found that the neural network with 2 convolutional layers followed by an attention layer and 2 Bi-LSTM layers performed the best with an F1 score of 91.95% on the test dataset (Fig. 2D).

B. Benchmarking

To assess the computational performance of our model, we compared our neural network to ADpred, a recently published AD prediction neural network [5]. In addition to a different architecture, a key difference between the two models is the preprocessing of the input data. While ADpred performs one-hot-encoding of the sequence and includes one secondary structure feature, our approach computes several side chain properties and properties associated with intrinsically disordered proteins. To show that using a feature matrix as input can alleviate the challenges associated with predicting ADs (i.e. the lack of a conserved sequence), we used our test dataset and compared the predictions from our neural network with ADpred's predictions (Fig. 3). We used ADpred's trained model, which was trained on the same dataset. ADpred computes a prediction score for each AA and recommends classifying an AD when the prediction scores are abovc e ≥ 0.8 over at least 10 continuous AA. Using these criteria, as well as considering a less stringent classifiation relying on only 1 AA above the recommended prediction threshold of 0.8, our neural network outperformed ADpred for accuracy, recall, and F1 score. While, our neural network did not have a higher precision than the benchmark. Taken together, we showed that computing sequence descriptors and properties advance the model's ability to predict ADs.

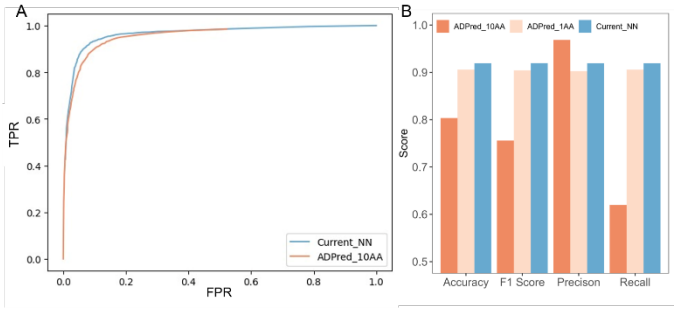

Fig. 3. The neural network's performance. Comparison of the performance of our neural network (Current_NN) and ADpred using the test dataset. ADpred's performance was evaluated using a 0.8 prediction threshold for at least 1 AA (ADPred_1AA) and 10 consecutive AAs (ADPred_10AA).

C. Identifying sequence properties key for activation domain predictions

Because the primary sequence is not critical for AD functionality and ADs are generally intrinsically disordered, it is challenging to find rules that define ADs. Such a set of rules would greatly benefit the identification and classification of ADs across species. Moreover, this set of rules will aid in the understanding of the mode of action of ADs, how they bind to interaction partners, and how they allow for context-specific interactions. The idea here is that our neural network can capture complex linear and non-linear correlations between the input features and the predictions that are generally missed by other computational models. To find the input features that impact our model's prediction the most, we used an approach that quantifies the importance of input features. Specifically, we computed a SHAP (SHapley Additive exPlanations) value for each feature of the sequences with the highest 2% prediction probability score for both classes (i.e. ADs and non-ADs). For each

prediction, SHAP assigned an importance value to each input feature. As expected, we found properties known to be important for AD activity, including hydrophobics, charged residues, aromatics, and the fraction of negative residues (Fig. 4). Consistently, the counts of tryptophan (W), an aromatic residue, and Aspartic Acid (D), an acidic residue, were also found to be key for proper AD prediction. Surprisingly, valine (V) and glycine (G) residues, two aliphatic residues, appeared to be important indicators for AD prediction (Fig. 4). Overall, by using this approach, we were able to identify the properties most important for AD prediction, and, importantly, we greatly improved the interpretability of our neural network.

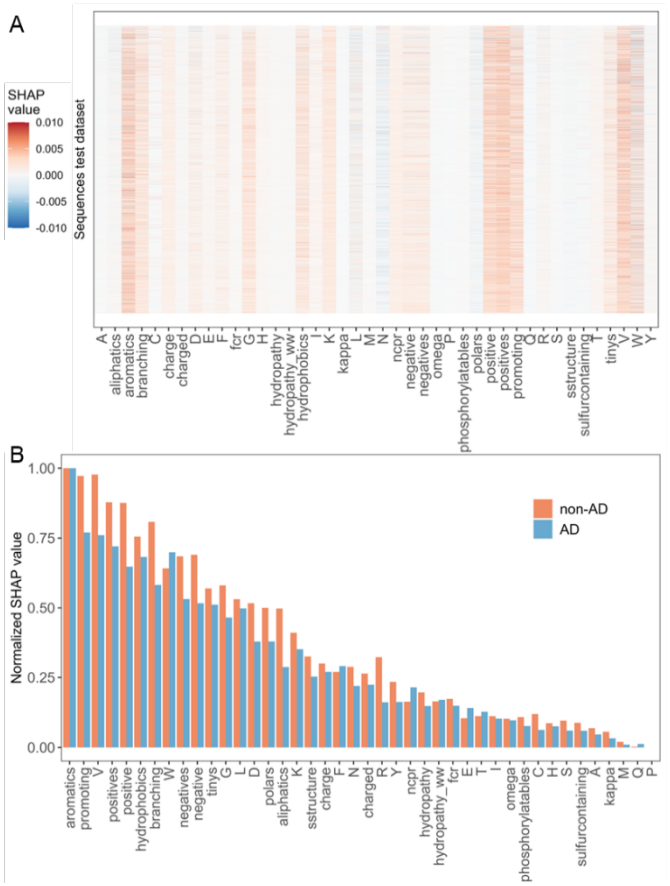

Fig. 4. Importance of input features. (A) The SHAP values averaged across the 26 subsequences for each input feature. The SHAP values were calculated for the test dataset classified as AD. (B) Normalized SHAP values ranked from most important to least important for both classes (AD and non-AD sequences). The SHAP values were averaged across the 26 subsequences and the test dataset sequences.

D. Classifying activation domains using unsupervised learning

In literature ADs have been classified arbitrarily depending on their enrichment in acidic residues, glutamine, or proline. These three different subclasses of ADs have been shown to differ in functionality and transcriptional activation strength. The acidic ADs are the most prominent class. However, even among the acidic ADs, it has been shown that leucine, aromatic, and negative amino acids play an important role. Thus, classifying ADs based on the enrichment of solely acidic residues, glutamines, or prolines is ambiguous. A less biased generalizable classification approach is needed. To overcome this challenge, we used our findings and interpretation of our neural network to perform unsupervised classification of ADs. The idea here is that each AD class will cluster spatially separately upon unsupervised clustering. To find the clusters among the sequences that have AD function, we opted to use the 12 most important features identified using our neural network while the weakly predictive features would be disregarded. We reasoned that the latter features would interfere, while the most important features contained the strongest signal for clustering. The unsupervised clustering was then projected onto a t-SNE analysis (see Methods). The t-SNE plot revealed that the ADs can indeed be further divided into subclasses (Fig. 5). Using this unsupervised classification approach, we visually imposed 10 clusters and thus 10 subclasses of ADs (Fig. 5). These AD subclasses have sufficient divergent features to form distinct clusters and may have their own functionality.

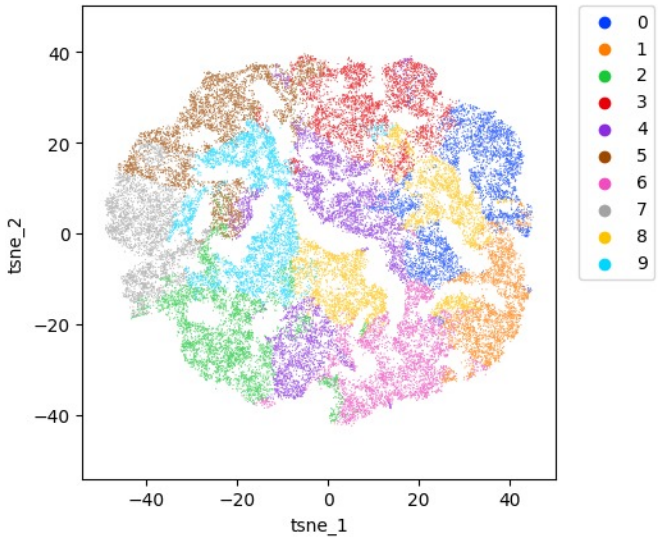

Fig. 5. AD classification. The ADs were divided into 10 subclasses based on k-means clustering of the 2D t-SNE output. T-SNE was performed on a 3-component PCA of the 12 most important features. The colors in the t-SNE plot correspond to k-means cluster memberships.

IV. Conclusions and Perspectives

The identification of effector domains, including transcriptional ADs, would greatly improve our understanding of the transcriptional activity, reconstructing gene regulatory networks, and identifying novel transcription factors. However, predicting ADs is currently challenging as a result of the lack of sequence conservation across ADs and the lack of available predictive tools. In this study, we developed a multilayer neural network containing convolutional, activation, and recurrent layers to learn sequence motifs, location, and order. To increase the interpretability of our neural network and evaluate the learned features, we computed the importance of each input feature during prediction. We found the importance of known, but also novel sequence properties. Taken together, features that correlate with AD function included the presence of aromatic, hydrophobic, positive, and negative residues, the fraction of residues predicted to be 'disorder promoting', and the number

of valine, tryptophan, glycine, leucine, and glutamine residues. This set of important features or rules was further used to classify the ADs in an unsupervised manner. For the ADs in our random peptide dataset, we imposed 10 subclasses. These subclasses could be correlated with or even functionally driving cell-type specificity, plant development, or plant response specificity. Whether or not members of the different subclasses are coregulated with specific cell types, throughout development, or upon a plant response remains to be explored potentially through single cell sequencing. We envision that a sequential pipeline of training a neural network, capturing its learned features, and unsupervised classification could be applied to ADs from specific species. As such, a species-specific and highly specialized network and set of rules for AD identification could be generated.

ACKNOWLEDGMENT

We thank Matthew Murray and Antonio Alonso-Stepanova for their valuable work and input at the start of this project.

REFERENCES

[1] G. Eraslan, Ž. Avsec, J. Gagneur, and F. J. Theis, "Deep learning: new computational modelling techniques for genomics.," Nat. Rev. Genet., vol. 20, no. 7, pp. 389–403, Jul. 2019, doi: 10.1038/s41576-019-0122-6.

[2] L. V. den Broeck et al., "Functional annotation of proteins for signaling network inference in non-model species," Res. Sq., Nov. 2022, doi: 10.21203/rs.3.rs-2201240/v1.

[3] M. Necci, D. Piovesan, CAID Predictors, DisProt Curators, and S. C. E. Tosatto, "Critical assessment of protein intrinsic disorder prediction.," Nat. Methods, vol. 18, no. 5, pp. 472–481, May 2021, doi: 10.1038/s41592-021-01117-3.

[4] M. Necci, D. Piovesan, CAID Predictors, DisProt Curators, and S. C. E. Tosatto, "Critical assessment of protein intrinsic disorder prediction.," Nat. Methods, vol. 18, no. 5, pp. 472–481, May 2021, doi: 10.1038/s41592-021-01117-3

[5] A. Erijman et al., "A High-Throughput Screen for Transcription Activation Domains Reveals Their Sequence Features and Permits Prediction by Deep Learning.," Mol. Cell, vol. 78, no. 5, pp. 890-902.e6, Jun. 2020, doi: 10.1016/j.molcel.2020.04.020.

[6] N. F. C. Hummel, A. Zhou, B. Li, K. Markel, I. J. Ornelas, and P. M. Shih, "The trans-regulatory landscape of gene networks in plants," BioRxiv, Oct. 2022, doi: 10.1101/2022.10.23.513368.

[7] J. Jumper et al., "Highly accurate protein structure prediction with AlphaFold.," Nature, vol. 596, no. 7873, pp. 583–589, Aug. 2021, doi: 10.1038/s41586-021-03819-2.

[8] A. S. Holehouse, R. K. Das, J. N. Ahad, M. O. G. Richardson, and R. V. Pappu, "CIDER: Resources to Analyze Sequence-Ensemble Relationships of Intrinsically Disordered Proteins.," Biophys. J., vol. 112, no. 1, pp. 16–21, Jan. 2017, doi: 10.1016/j.bpj.2016.11.3200.

[9] S. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," arXiv, 2017, doi: 10.48550/arxiv.1705.07874.

[10] R. J. Emenecker, D. Griffith, and A. S. Holehouse, "Metapredict V2: An update to metapredict, a fast, accurate, and easy-to-use predictor of consensus disorder and structure," BioRxiv, Jun. 2022, doi: 10.1101/2022.06.06.494887.