Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

OXFORD

Sequence analysis

AttCRISPR : an spacetime interpretable model for sgRNA efficiency prediction

Corresponding Author [1,*], Co-Author [2] and Co-Author [2,*]

[1]Department, Institution, City, Post Code, Country and
[2]Department, Institution, City, Post Code, Country.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: More and more higher specificities Cas9 variants are developed to avoid the off-target effect, which bring a significant volume of experimental data. Conventional machine learning performance poorly on these datasets, while model based on deep learning are often lack of interpretability, which makes it difficult for researchers to understand its decisions. Moreover, neither the deep learning based model with existing structure can not satisfy enough precision at such huge datasets.
Results: To overcome it, we design and implement AttCRISPR, a deep learning based model to predict the on-target activity. Our model was trained and tested on the biggest dataset, DeepHF dataset, as far as we know for performance evaluation. AttCRISPR achieves the best performance on DeepHF dataset, yielding an average spearman value of 0.99%, 0.99%, 0.99% (corresponding to WT-SpCas9, eSpCas9(1.1), SpCas9-HF1) under tenfold shuffled validation. In addition, another advantage of AttCRISPR over other well-perform methods is that it is intrinsic interpretable and does not rely on other post hoc explanations techniques. In this paper, We design a set of algorithm to reveal the biological significance of the decision maked by AttCRISPR from the global and local perspectives at sgRNA overall level and nucleotide level.
Availability: The example code are available at https://github.com/South-Walker/AttCRISPR
Contact: xlm@xiaoliming96.com
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Clustered regularly interspaced short palindromic repeats (CRISPR) / CRIPSR associated protein 9 (Cas9) systems is preferred over other biological research and human medicine technologies now, beacuse of it's efficiency, robustness and programmability. Cas9 nucleases can be directed by short guide RNAs (sgRNAs) to introduce site-specific DNA double-stranded breaks in target, so to enable editing site-specific within the mammalian genome (Jinek *et al.*, 2012; Cong *et al.*, 2013; Mali *et al.*, 2013). CRISPR/Cas9, to a large extent, has developed genetic therapies at the cellular level, while there are still severe medical disadvantage even now which has greatly hindered the further clinical application of the CRISPR/Cas9 systems. One of these disadvantage is due to point mutations caused by off-target effects (Rubeis and Steger, 2018; Kang *et al.*, 2016; Ishii, 2017; Liang *et al.*, 2015). To overcome this disadvantage, a solution is to engineer CRISPR / Cas9 with higher specificities. That's why more and more higher specificities Cas9 variants, such as enhanced SpCas9 (eSpCas9(1.1)), Cas9-High Fidelity (SpCas9-HF1) (Ishii, 2017; Slaymaker *et al.*, 2016), hyper-accurate Cas9 (HypaCas9) (Kleinstiver *et al.*, 2016), been developed and bring a significant volume of experimental data, that is to say researchers have to face the difficulty of analyzing such huge and heterogeneous data.

The activity of chosen sgRNA sequence determines the success of genome editing, however distinctly fluctuant behaviors have been observed for the performance of different sgRNAs, even in the same Cas9 system. Some optimum sgRNAs can hit almost all targers alleles, while anothers don't even show activity (Wang *et al.*, 2019). This fact indicates that it's meaningful to explore an efficient approaches to guide sgRNA design.

In practice, there have been a number of application and toolkit applied in this task. In the earlier studies, methods in silico are categorized into three types: (1) alignment-based, (2) hypothesis-driven and (3) learning-based (Chuai *et al.*, 2018). Recently we noticed that the last type of method

seems to be getting more attention because of huger and huger data set (Liu *et al.*, 2019a).

Learning-based method, which designed to predict the on-target activity (or off-target probability) of sgRNAs, is essentially a computational model built by machine learning algorithm, not only conventional machine learning but also deep learning algorithm. In general, the single or multiple base sequences (vary according to the task) and biological features are represented as a multi-dimensional vector $X \in \mathbb{R}^d$, and $d = l + b$, where l refers to the length of base sequences and b refers to the number of biological features, these methods can be represented as

$$y = Score(X) \tag{1}$$

where $Score(\cdot)$ depends on the algorithm being selected, and y denotes the predicted value. Some studies on HT_ABE and HT_CBE have shown that deep learning based models often outperformed conventional machine learning, when the number of sgRNAs in the data set reached a certain level (Song *et al.*, 2020; Kim *et al.*, 2018, 2019). Per contra, conventional machine learning algorithms, such as linear regression, logistic regression and the decision tree, are often more interpretable due to the fewer parameters and clearer mathematical assumptions. In short, what was needed for developer is to trade-off accuracy and interpretability. Muhammad Rafid *et al.* consider deep-learning models as black boxs and believe they lack interpretability, motivated by the empirical assertion, they turn to build a model based conventional machine learning to compete with state-of-the-art deep learning models (Muhammad Rafid *et al.*, 2020). On the other hand, input perturbation based feature importance analysis become a preferred components to reveal the importance of features in deep learning models. Liu *et al.* use a sliding window of length 2 to extract dimeric as input and rank the position of dimeric by contribution to final output (Liu *et al.*, 2019b). One regret is that subject to the processing of the input sgRNA sequence, their analysis can not exactly on the nucleotide class. Further, SHAP, one of the most prominent of model explain techniques, has been widely used to understand the decision made by the model. Wang *et al.* develope DeepHF, a deep learning based model, and use Deep SHAP to revealed nucleotide contributions (Wang *et al.*, 2019). Deep SHAP is a compositional approximation of SHAP values since it is challenged to computate SHAP values exactly, especially for a complex deep neural networks (Lundberg and Lee, 2017). In our understanding, the method based on input perturbation often requires better generalization ability of the model (even for artificial ridiculous noise data). Moreover, recent work indicates that model explain techniques, which based post hoc explanations techniques and input perturbations, could be fooled to generate meaningless explanations instead of reflecting the underlying biases (Slack *et al.*, 2019), in other word, they could be unreliable and misleading, even on model with excellent performance. In addition, as far as we know, all the interpretable deep learning models today can only analyze the preference of on-target activity (or off-target probability) on specific nucleotide species and position for example, the G adjacent to PAM has a positive effect on sgRNA activity as Wang *et al.* report, which we'll call the first-order preference in our paper, due to it's similar to the total differential of y in Equation (1) as following

$$dy = \sum_{i}^{d} a_i dx_i \tag{2}$$

$$a_i = \frac{\partial}{\partial x_i} Score(X) \tag{3}$$

where x_i denotes the i-th dimension of the vector X, a_i indicates how dramatically the function changes as x_i changes in a neighborhood of X. Similarly the first-order preference indicates how dramatically the function changes as x_i changes, in other word, the importance of x_i. However

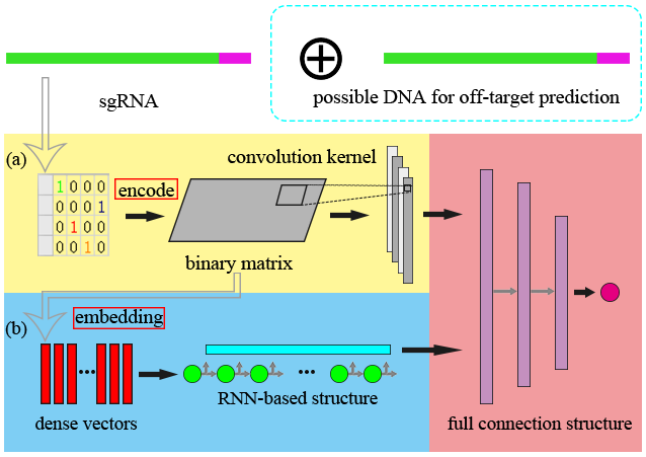

Fig. 1. Two categories of the deep learning models used in sgRNA related task. (a) Model work in spatial domain. In spatial domain, the base sequence is encoded into a binary matrix (or a binary image). Since convolution has great advantages in extracting spatial features, the convolutional neural network is an excellent tool in spatial domain. (b) Model work in temporal domain. In temporal domain, the base sequence (represented by the binary matrix) is embedded into a sequance of high-dimensional vector, in which the recurrent neural network perform better. In addition, we note that the last layers of neural network are usually full connection structure (not necessarily), which greatly increases the difficulty of understanding the decisions of model.

they didn't analyzed the preference for position i nucleotide at the overall level, which we'll call the second-order preference in our paper due to its calculation is based on first-order preference,

Specifically, we use a vector A_i to build the first-order original preference at position i within sgRNA sequence, then define A as the first-order combine preference matrix (or just first-order preference), which means $\tilde{A}$ can be expressed linearly by A as follow

$$\tilde{A} = BA \tag{4}$$

where the weight matrix $B \in \mathbb{R}^{l \times l}$ is learned through attention mechanism. Then the predicted value can be expressed as

$$y = \sum_{i}^{l} W_i \tilde{A}_i \cdot X_{ei} \tag{5}$$

where $W \in \mathbb{R}^l$ denotes a non-negative weight vector, W_i denotes the weight of the i-th position, X_{ei} denotes the embedding vector of the nucleotide at i-th position. Assume that in Equation (4) the first-order original preference at each position is relatively independent and weight matrix B is independent of any first-order original preference, then the total differential of the first-order preference at i-th position $\tilde{A}_i$ could be written as

$$d\tilde{A}_i = \sum_{j}^{l} B_{ij} E dA_j \tag{6}$$

where E is the identity matrix, and we call B as the second-order preference matrix, it can explain how a particular pattern containing two nucleotides affects the base sequence, for example, how does the degree of preference for the G adjacent to PAM be affected by a T at 2 bases upstream?

The previous work (even based on conventional machine learning algorithms with high interpretability) was limited to dimers which can only work on the adjacent base pairs (Liu *et al.*, 2019b) or have to perform complex feature engineering (Muhammad Rafid *et al.*, 2020). In light of the above, we believe it is essential to develope a model which can not only

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

match deep learning based model in performance, but also be comparable to conventional machine learning algorithms in interpretability.

Deep neural network has shown its power in the study of CRISPR/Cas9 and its improved Systems (Liu *et al.*, 2019a). Most of the deep neural network existing are the combination of recurrent neural network (RNN), convolutional neural network (CNN), fully connected neural network (FNN), and their variants. As Figure 1 show, We found that the deep learning models used in sgRNA on-target activity (even for off-target effect) prediction tasks in recent years can be divided into the following two categories according to the encoding approach of the sgRNA sequence (sgRNA-DNA sequence pair, for off-target effect prediction):

1. Methods in spatial domain. Some previous studies have used model based CNN to predict gRNA on-target activity or off-target effects (Lin and Wong, 2018; Kim *et al.*, 2018; Chuai *et al.*, 2018). They process sgRNA base sequence inputs with the help of one-hot encoding idea. In other words, they regard it as two-dimensional image data, and use convolution layer to extracte potential features in spatial domain, It is worth noting that Zhang *et al.* adds bidirectional gated recurrent unit (BGRU, in short), a RNN variant, after pooling layer of classic CNN network (Zhang *et al.*, 2020). Our explanation is that BGRU assists CNN to extract spatial features in one dimension, under this belief it belong to this category.
2. Methods in temporal domain. Although RNN-based network have been shown effective to improve the performance of the model with temporal sequential input, especially in natural language processing and sequential recommendation(Huang *et al.*, 2018), RNN be not used for gRNA activity prediction, until recently (Wang *et al.*, 2019; Liu *et al.*, 2020, 2019b). They consider the nucleotides(can also dimer or polymer) in the sgRNA sequence as word, and the sgRNA sequence itself as a sentence (from 5' to 3'), then a trainable matrix (could be either supervised or unsupervised) is used to project the word to the dense real-valued space. This technology is called embedding, which generates the base embedding. RNN further encoding the base embedding into a sequence of hidden state vector. Specially, Liu *et al.* use RNN and CNN in parallel to extract features in base embedding (Liu *et al.*, 2019b). However, base embedding is not spatially interpretable (different from one-hot encode), and they have no way to further explore the correlation between CNN and RNN output. Almost all RNN based models used in sgRNA on-target activity or off-target effect flatten the hidden state vector into a one-dimensional vector as the input of the fully connected layer. It is a pity that the temporal sequential dependency of hidden state vector are rarely noticed. To summarise, RNN has limited representation power in capturing spatial feature. Furthermore, hidden state vector representation is usually hard to understand and explain.

Attention mechanism has demonstrated its power in Natural Language Processing, Statistical Learning, Speech and Computer Vision (Chaudhari *et al.*, 2019). It makes model tends to focus selectively on parts of the input which is help in performing the task effectively. Previous observation have shown that Cas9 preferentially binds sgRNAs containing purines but not pyrimidines (Wang *et al.*, 2014) and multiple thymine in the spacer impairing sgRNA activity (Wu *et al.*, 2014), that is to say some specific nucleotide and base position need more attention compared to others. The above is the premise of introducing attention mechanism. Strictly speaking, we are not the first to bring attention mechanisms into this field. The most similar approach to ours is the work based on transformer by Liu *et al.*. They use transformer, a components based on attention mechanism, instead of RNN to improve the ability of temporal feature extraction, hence, enhance the performance of their model (Liu *et al.*, 2019b; Vaswani *et al.*, 2017). In our work,the interpretability benefit from attention mechanism is more focused. In this paper, our main contributions are as follows:

- Present a novel deep-learning model, which can extract potential feature representation of sgRNA sequence in both spatial and temporal domain parallelly. Finally, the ensemble learning method is used to combine the two to achieve better performance than other models. It doesn't belong to any of the above categories of existing approaches.
- Introduce attention mechanism into our model. As a result, It does not need post hoc explanations techniques based on input perturbation to explain itself. It is intrinsic interpretable in both temporal and spatial domains. In the spatial domain it's at the nucleotide level, in our word, first-order preference, while at the overall level in the temporal domain, in our word, second-order preference. Thus, it is transformed from a black box to an intrinsically interpretable model with the performance of deep learning based model.
- Throught ablation analysis and testing a series of possible network structure, we find there are multiple components and strategy can improve the performance of our model and constructed AttCRISPR, which could outperform the current state-of-the-art tool on DeepHF dataset.

2 Materials and methods

2.1 Sequence encoding and embedding

For encoding process, we use the complementary base to represent the original base in sgRNA, as others do. Further, we use one-hot encode strategy, which's is to say, we encoding each base in sgRNA into a four-dimensional vector (encode A,T,G,C into [1,0,0,0], [0,1,0,0], [0,0,1,0], [0,0,0,1], respectively), called one-hot vector. Then a sgRNA can be considered as a matrix $X_{oh} \in \mathbb{R}^{l\times 4}$, named one-hot matrix (a little sparse, since an one-hot vector is zero in all but one dimension). We believe it is meaningful to regard X_{oh} as a binary image, therefore, it is used as an input of convolutional neural network, which performs well in the image field. Meanwhile, as mentioned above, one-hot matrix is a little sparse.

To facilitate the training process, we can mapping each one-hot vector into a dense real-valued high-dimensional space, which is called embedding. Concretely speaking, at the matrix level, the formula is as follows

$$X_e = X_{oh}E_m \tag{7}$$

where X_e named embedding matrix, $E_m \in \mathbb{R}^{4\times m}$ is a trainable transformational matrix, m refers to the dimension of embedding space. We believe it is also meaningful to regard nucleotides in the sgRNA sequence as word, and the sgRNA sequence itself as a sentence, Guided by this belief E_m is the word embedding matrix and X_e is the sentence embedding in natural language processing (NLP). Therefore, X_e is used as an input of recurrent neural network (or its variant), which performs well in the NLP field. In the area of NLP, there have been many methods based on pre-training and unsupervised learning been developed to determine the E_m. However, we believe that sgRNA sequence is not a natural language in the true sense, the method above does not apply to it. So E_m will be trained along with the model as a whole.

Due to each element of X_{oh} is interpretable (representing whether there is a corresponding nucleotide type at the corresponding location), we call X_{oh} the spatial input, and the convolutional neural network work on X_{oh} is method in spatial domain. In other hand, different from X_{oh}, X_e can only be explained in the first dimension (representing the embedding vector of corresponding nucleotide type), and embedding vector is difficult for humans to understand. That's why we call X_e the temporal input, and the recurrent neural network (or its variant) work on X_e is method in temporal domain.

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

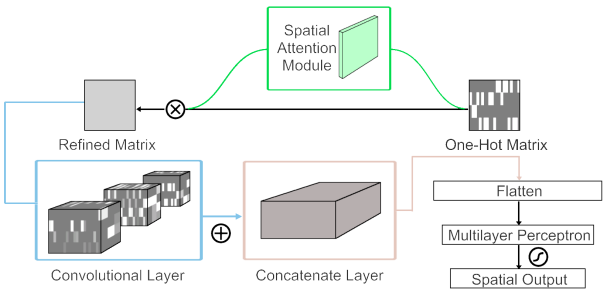

Fig. 2. The architecture of spatial domain method in AttCRISPR. The input of the method is encoded sgRNA sequence X_{oh}, a 21×4 one-hot matrix. Then refine it through a spatial attention module, which could tell us the importance of a specific matrix element (or just say, pixel). A simple convolutional neural network followed is applied to extract potential feature representation of sgRNA sequence. In the last step, we flatten the output of convolutional neural network into a one dimensional vector and use a multilayer perceptron with sigmoid activation function to achieve the spatial output y_s.

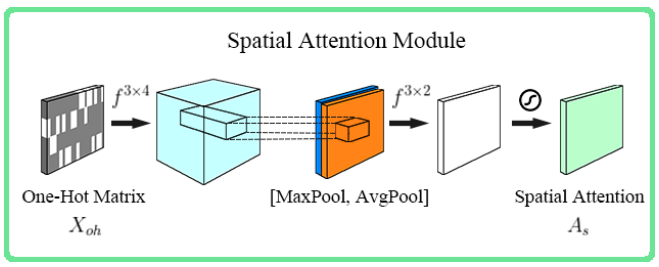

Fig. 3. Details of spatial attention module. A convolution layer is used to generate multi-channel map from X_{oh}. Then concatenated the output of both max-pooling and average-pooling method and forward it to the last convolution layer. A sigmoid function is used to map the final result to a range of zero to one at last, which generates the spatial first-order preference matrix A_s.

2.2 Neural network architecture

Based on the categorization above, we assume that method in spatial domain and in temporal domain are heterogeneous, which can satisfy the diversity premise of ensemble learning. Based on assumption above and ensemble learning, we follow the stacking strategy to develop AttCRISPR which can extract potential feature representation of sgRNA sequence in both spatial and temporal domain parallelly. Further, we apply attention mechanisms in both spatial and temporal domain respectively to enhance the interpretability of AttCRISPR.

2.2.1 Method and attention in spatial domain

As Figure 2 show, method in spatial domain relies on the convolutional neural network. As previously mentioned, sgRNA sequence has been encoded into a 21×4 one-hot matrix X_{oh}, and we regard X_{oh} as a binary image. Then, convolution kernels with different size are used to extract potential spatial features just like what other do in computer vision. According to the foregoing, the Spatial Attention Module proposed by Woo *et al.* can be applied in our method in spatial domain, which was used to improve the performance of CNN in vision tasks.

As Figure 3 show, Given an one-hot map X_{oh}, spatial attention module generating spatial attention matrix $A_s \in \mathbb{R}^{l\times4}$ with the same as X_{oh} in shape. Each element of A_s is constrained to a range of zero to one, implemented by a sigmoid function, which reflects the importance of the corresponding elements of X_{oh}. The overall spatial attention process can be summarized as:

$$\begin{cases} X_{mc} = f^{3\times4}(X_{oh}) \\ A_s = \sigma(f^{3\times2}([AvgPool(X_{mc}); MaxPool(X_{mc})])) \\ X_{rf} = A_s \otimes X_{oh} \end{cases} \tag{8}$$

where $f^{p\times q}$ represents a convolution operation with the filter size of $p \times q$, $p, q \in \mathbb{Z}^+$, X_{mc} is a multi-channel map generated by X_{oh}, σ denotes the sigmoid function, AvgPool denotes the average-pooling operation, MaxPool denotes the max-pooling operation, $\otimes$ denotes element-wise multiplication. Spatial attention matrix A_s formally conforms to our proposed definition of first-order preference (each element of X_{oh} is multiplied by the corresponding element of A_s), in other word, element of A_s reveal how important the corresponding elements in X_{oh} is. We think it can reveal the preference of the scoring function at each position. For instance, follow the encoding rules above, train spatial domain part of AttCRISPR with the WT-SpCas9 dataset. Then take the average of all spatial attention matrix, and the element in the first row and third column are closer to 1, which means when calculate the final score, G typically may have a important contribution at first position within sgRNA sequence. In fact, this corresponds to some early studies concerning the Human (hU6) promoter, which is believed to require G as the first nucleotide of its transcript (Cong *et al.*, 2013; Jinek *et al.*, 2012; Mali *et al.*, 2013).

2.2.2 Method and attention in temporal domain

As Figure 4 show, temporal domain part of AttCRISPR relies on the recurrent neural network (or its variant). As previously mentioned, we mapping each one-hot vector into a dense real-valued high-dimensional space follow the Equation (7), which generates the embedded matrix X_e. And we regard X_e as a sequential data, or temporal data. Recurrent neural network (or its variant) has showed outstanding performance in the tasks with temporal data (for instance, natural language processing, sequential recommendation). That's why we prefer to use it to extract potential temporal features. To be precise, we prefer the architecture of encoder-decoder which has been proven to be effective in Seq2Seq task. Two main differences we have to face are that sgRNA is not a natural language in the traditional sense, and we don't have to translate it to other sequence. To accommodate them, the embedded matrix X_e is used as input of both the encoder and decoder, and the sequence of decoder is to build the first-order preference of sgRNA sequence $\tilde{A}$. As mentioned above, the predicted value y should satisfy Equation (5).

On this basis, we apply the idea of attention mechanism which has been widely used in natural language processing tasks to AttCRISPR in the method of temporal domain, and name it Temporal Attention Module (Vaswani *et al.*, 2017; Luong *et al.*, 2015; Bahdanau *et al.*, 2014). As the idea of Vaswani *et al.*, Temporal Attention Module satisfies the following equation

$$Attention(Q, K, V) = align(Q, K)V \tag{9}$$

where the $align(\cdot)$ is put forward by Luong *et al.*. Q, K, V are queries, keys and values matrix accordingly in the paper of Vaswani *et al.*.

As Figure 5 show, in our attention module they are calculated by the following equation

$$\begin{cases} K_i = Encoder(X_{ei}, \theta_E, K_{i-1}) \\ Q_i = Decoder(X_{ei}, \theta_D, Q_{i-1}) \\ V = K \end{cases} \tag{10}$$

where vector K_i, Q_i denotes the i-th row of the matrix K and Q accordingly, $Encoder(\cdot)$ and $Decoder(\cdot)$ are independent GRU units, θ_E and θ_D denote all the related parameters of GRU networks accordingly. In the actual implementation, we apply the bidirectional GRU networks

for better performance, and for the sake of conciseness, we show a conventional GRU network here. The function $align(\cdot)$ is as follows

$$B = align(Q, K) \tag{11}$$

$$B_i = softmax(Q_i K^T) \otimes G_i \tag{12}$$

$$G_{ij} = \begin{cases} exp(\frac{(i-j)^2}{-2\sigma}) & , |i-j| \leqslant \sigma \\ 0 & , |i-j| > \sigma \end{cases} \tag{13}$$

where the matrix $B \in \mathbb{R}^{l \times l}$ is the second-order preference we need, and vector B_i denotes the i-th row of the matrix B. $G \in \mathbb{R}^{l \times l}$ is the damping matrix base on Gaussian function. Since a simple belief that the closer the base is to the i-th position, the more it affects the i-th position normally, we use the damping matrix G to constrain the network learning. σ represents a threshold of length, any base over this length from the position i is not considered to be affected. Further, if we think of the values matrix as a vector form of the first-order preference A in Equation (5), we can reach the following equation

$$\tilde{A} = BV \tag{14}$$

according to above, values matrix V comes from the hidden states of a bidirectional GRU networks, which is usually hard to understand and explain. While B is the second-order preference matrix obtained by the attention mechanism, in our belief, the j-th dimension of B_i, denoted as B_{ij}, can reveal the effect of the base at position j on position i in the biological sense.

2.2.3 Model ensemble following stacking strategy

As Wang *et al.*; Li and Yu; Wang *et al.* have shown, some indirect sgRNA features, which can't be obtained directly by deep learning, including position accessibilities of secondary structure, stem–loop of secondary structure, melting temperature, and GC content are strongly associated with sgRNA activity. It's worth noting that in the work of Wang *et al.*, hand-crafted biological features didn't be normalized. Since the wide range of data distribution, we believe it makes sense to normalize it, and we preprocess they based on the standard score (Z-Score).

Then we use a simple full connection network to extract the indirect features, and call the output of fully connection network y_{bio}. As mentioned above, we assume that method in spatial domain and in temporal domain are heterogeneous, which can satisfy the diversity premise of ensemble learning. That's why we follow the stacking strategy, integrate the methods in time domain and space domain. Specifically, the y_{bio}, the spatial output y_s and the temporal output y_t we got earlier are concatenated and then weighted averaging is performed through a full connection layer as follow

$$y = W[y_{bio}; y_s; y_t] \tag{15}$$

where, y is the final prediction value of AttCRISPR, W is the weight learned by the full connection network. In the actual implementation, we freeze the network in the spatial domain and temporal domain at first, in order to make our network focused on learning the weight W. Later in the process, in the fine tuning of AttCRISPR, the parameters of the entire network are adjusted.

2.3 Datasets and current prediction method

The dataset we used for training, validation and testing is built by Wang *et al.* (Wang *et al.*, 2019). We extracted 55604, 58617, 56888 sgRNAs with activity (represented by insertion/deletion (indel)) for WT-SpCas9, eSpCas9(1.1) and SpCas9-HF1 respectively, from its source data. As far as we know, the best performance on this dataset is the DeepHF of Wang *et al.* currently, which base on deep learning and bidirectional LSTM architecture. DeepHF achieved Spearman correlation coefficients

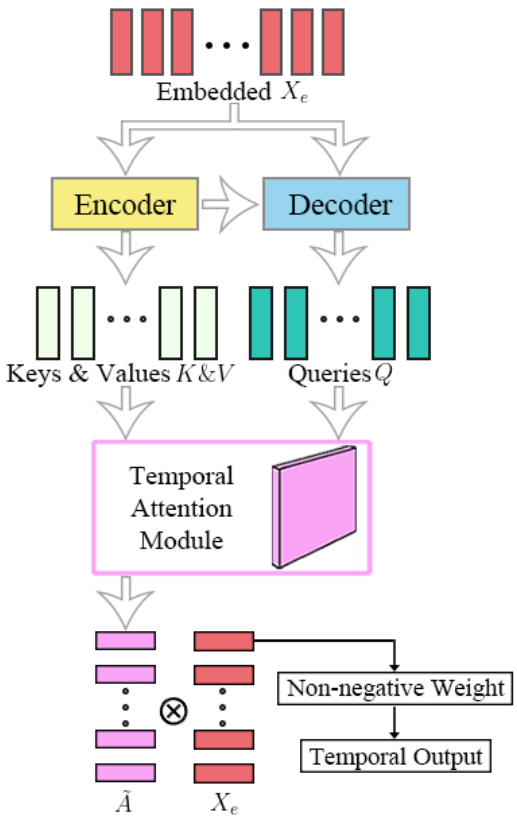

Fig. 4. The architecture of temporal domain method in AttCRISPR. The input of the method is embedding sgRNA sequence X_e, a $21 \times e$ embedding matrix, where e is the dimension of a nucleotide embedding vector. Then Keys K, Values V and Queries Q is generated through a classic encoder-decoder structure which is need for temporal attention module. Next, the temporal attention module generates the first-order preference $\tilde{A}$, a $21 \times e$ matrix (or a vector set). Each of the row vectors in matrix $\tilde{A}$ represents the base preference of sgRNA at the corresponding position, we use their dot product with the corresponding row vector in embedded X_e to build the score of corresponding position. Hence, a full connection layer is used to weighted average them and achieve the temporal output y_t.

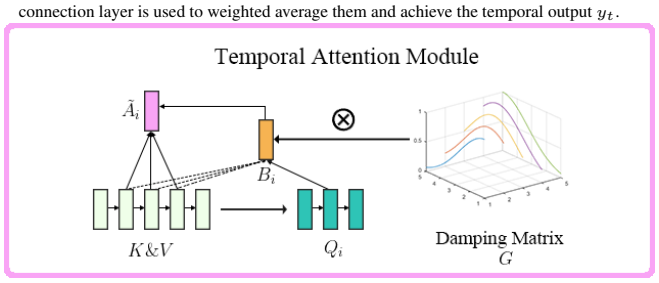

Fig. 5. Details of temporal attention module. To generate the first order preference vector in the i position of sgRNA $\tilde{A}_i$. First, the i-th row vector of the querys matrix Q_i is multiplied by the transpose of the keys mateix K^T, and apply a softmax function to obtain a preliminary weights vector on the values. Second, to favor the alignment points near i, the weights vector obtained is multiplied element-by-element by the i-th row vector of the damping matrix G, as Equation (12) has shown. G_{ij} can be regarded as the result of place a Gaussian distribution centered around i, then sampling the position j (a scaling factor is used to ensure the sum of G_I is l). Then we achieve the second-order preference matrix B, it is also a weights matrix on the values. So the first-order preference matrix $\tilde{A}$ come from the product of B and values matrix V. Although it's not shown above, it is also worth noting that, a full connection layer with L2 constraints is used to generates a transformation matrix in order to ensure $\tilde{A}_i$ and $X_{e\,i}$ are in the same vector space. Temporal attention module generates the temporal first-order preference matrix $\tilde{A}$ and the temporal second-order preference matrix B.

Table 1. The methods compared in our work and brief description of these methods

Method	Neural	End2end	Description
CNN*	Yes	Yes	Naive convolutional neural network
RNN*	Yes	Yes	Bidirectional long short-term memory neural network
XGBoost*	No	Yes	Extreme Gradient Boosting regression tree
MLP*	Yes	Yes	Multilayer perceptron
DeepHF*	Yes	No	Bidirectional long short-term memory neural network (with hand-crafted biological features)
CRISPRpred(SEQ)#	No	No	A conventional machine learning pipeline
SpAC	Yes	Yes	Spatial AttCRISPR
TAC	Yes	Yes	Temporal AttCRISPR
EnAC	Yes	Yes	Ensemble AttCRISPR (without hand-crafted biological features)
StAC	Yes	No	Standard AttCRISPR

Note: The method with superscript of * and # is reported by Wang *et al.* and Muhammad Rafid *et al.* respectively. Specially, CRISPRpred(SEQ) take another set of hand-crafted sequence-based feature to improve performance.

of 0.867, 0.862 and 0.860 for WT-SpCas9, eSpCas9(1.1) and SpCas9-HF1 respectively.

Another tool worth noting is CRISPRpred(SEQ), which is based on conventional machine learning and need complicated human feature engineering exercise (Muhammad Rafid *et al.*, 2020). They also test their tool with this dataset and achieved Spearman correlation coefficients of 0.838, 0.830 and 0.821 for WT-SpCas9, eSpCas9(1.1) and SpCas9-HF1 respectively (without any hyperparameter tuning).

Wang *et al.* implemente a tenfold shuffled validation to evaluate the stability of the performance of DeepHF. To be more specific, each set is shuffled and divided into three parts, 76.5%, 15% and 8.5% of the relevant data was used as the training, test and validation set respectively in a single experiment. The experiment is repeated ten times with the results recorded and averaged finally. In order to make the comparison apples to apples, we will follow this strategy to design experiments.

2.4 Experiments

Two different sets of experiments are carried out in our work. The first one is designed for ablation analysis of AttCRISPR. We compare the performance of end2end method (without any hand-crafted biological features) in both spatial and temporal domain. Furthermore, we test the ensemble method based on the same strategy, which proved that the ensemble of method in both spatial and temporal domain can significantly improve the performance.

The second experiment is designed to compare the performance of AttCRISPR and other current prediction method. In order to make the comparison apples to apples, we reduce the dimensionality of the same hand-crafted biological features as DeepHF's, which has been shown to enhance the predictability of a deep-learning model greatly, with a multilayer perceptron. Then follow the Equation (15) to achieve the final prediction value. AttCRISPR (with the hand-crafted biological features) performs better on all three data sets than DeepHF, the current state-of-the-art.

Our baselines have a comprehensive coverage of the methods tested in these datasets. In Table 1, we annotate some properties of these baselines

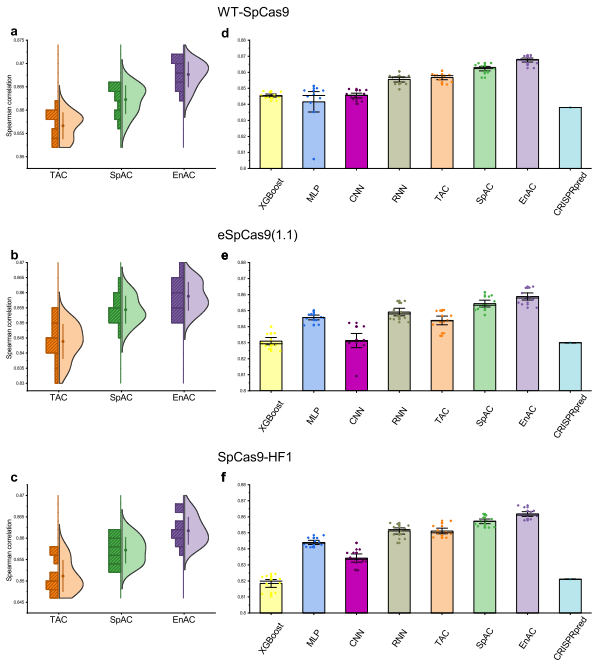

Fig. 6. In the absence of hand-crafted biological features, performance of different algorithms for sgRNA activity prediction. (a)-(c) The performance of Temporal AttCRISPR, Spatial AttCRISPR and Ensemble AttCRISPR. The half-violin plots show the mean and distribution of the Spearman correlation coefficient between predicted and measured sgRNA activity scores over all tests. (e)-(g) In the absence of hand-crafted biological features, the performance of all prediction methods in these datasets as far as we know. The $mean \pm s.d.$ of the Spearman correlation coefficient between predicted and measured sgRNA activity scores are shown in the bar plots.

(is/isn't neural models, is/isn't end2end models) All the experiments were carried out in Python 3.6 using Keras 2.2.4 and one GeForce RTX 2080Ti Super was used for training and testing if needed.

3 Results

We designed experiments to address the following questions:

- In the absence of hand-crafted biological features, whether the stacking of method in spatial domain and temporal domain can get better performance then use these method alone? → Section 3.1
- How does AttCRISPR perform compared to current state-of-the-art methods, covering both conventional machine learning and deep-learning models? → Section 3.2
- How can researcher understand the decisions make by AttCRISPR locally and globally, based on attention mechanisms? → Section 3.3

3.1 Model building and stacking

In Table 2, we list the performance of methods in spatial or temporal domain and the stacking of methods. Temporal AttCRISPR, TAC for short, achieved Spearman correlation coefficients of 0.857, 0.844, 0.851 in above three dataset. Spatial AttCRISPR, SpAC for short, corresponds to 0.862, 0.854, 0.857. In the absence of hand-crafted biological features. Ensemble AttCRISPR achieve the best performance of our knowledge, corresponds to 0.868, 0.859, 0.862.

In addition, in Table 2, the performance of other methods without using hand-crafted biological features, are also recorded. Regardless of the method we developed, RNN reported by Wang *et al.*, which can be

Table 2. Performance comparisons for different methods in the absence of hand-crafted biological features (take Spearman correlation coefficient as evaluation index)

Method	WT-SpCas9	eSpCas9(1.1)	SpCas9-HF1
XGBoost*	0.845	0.831	0.818
MLP*	0.842	0.846	0.844
CNN*	0.846	0.831	0.834
RNN*	0.856	0.849	0.851
TAC	0.857	0.844	0.851
SpAC	0.862	0.854	0.857
EnAC	**0.868**	**0.859**	**0.862**
CRISPRpred(SEQ)#	0.838	0.830	0.821

Table 3. Performance comparisons for methods before and after integrating with hand-crafted biological features (take Spearman correlation coefficient as evaluation index)

	WT-SpCas9		eSpCas9(1.1)		SpCas9-HF	
Method	Mean	Std Dev ($\times 10^{-3}$)	Mean	Std Dev ($\times 10^{-3}$)	Mean	Std Dev ($\times 10^{-3}$)
RNN*	0.856	3.33	0.849	5.00	0.851	4.11
EnAC	0.868	2.66	0.859	4.66	0.862	3.19
DeepHF*	0.867	**2.37**	0.862	4.24	0.860	3.21
StAC	**0.872**	2.55	**0.867**	**3.71**	**0.867**	**2.65**

Note: The method with superscript of * and # is reported by Wang *et al.* and Muhammad Rafid *et al.* respectively. In the tables, we use the results reported in the relevant paper as the performance of the method directly.

categorized as the method in temporal domain, is the most predictive with Spearman correlation coefficients of 0.856, 0.849, 0.851. It's obvious that the ensemble AttCRISPR is better at prediction (Figure 6 a-c). Furthermore, the prediction ability of models could be boosted by addition of other hand-crafted biological features, which can't be obtained directly by sequence information.

Hence, the second experiment is designed to compare the performance of standard AttCRISPR (hand-crafted biological features are used to improve performance of ensemble AttCRISPR) and current state-of-the-art method, DeepHF (the RNN mentioned above integrated with hand-crafted biological features).

3.2 Comparison to current methods

In order to make the comparison apples to apples, we follow Equation (15) to modify the ensemble method and design the control experiment using the same strategy to validate the conclusion that integrating with hand-crafted biological features can improve the predictive performance of methods. What's more, we compare the standard AttCRISPR and current state-of-the-art method, DeepHF, and show the results in Table 3.

As shown in Table 3, in the absence of hand-crafted biological features, AttCRISPR has significant advantages over DeepHF in predictability. Further, integration with the hand-crafted biological features can also improve the performance of AttCRISPR, and achieve Spearman correlation coefficients of 0.872, 0.867 and 0.867 for WT-SpCas9, eSpCas9(1.1) and SpCas9-HF1 respectively. Meanwhile, current state-of-the-art method, DeepHF achieve 0.867, 0.862 and 0.860 respectively. After integrating with biological features, the performance gap between AttCRISPR and DeepHF is shortened, while AttCRISPR still has better performance. In addition, we also compare the standard deviation of data

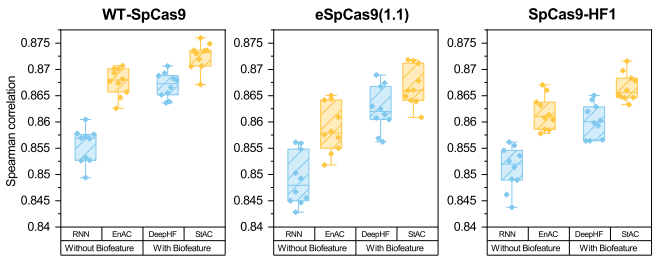

Fig. 7. Performance comparisons for methods before and after integrating with hand-crafted biological features, where DeepHF is the RNN integrated with hand-crafted biological features, and StAC is the EnAC integrated with hand-crafted biological features. The box plot show the mean and distribution of Spearman correlation coefficient between predicted and measured sgRNA activity scores over all tests.

obtained in ten tests, which are shown in Table 3. It reveal that AttCRISPR is more stable than DeepHF.

3.3 Interpretability of the AttCRISPR

To some extent, the attention module in the AttCRISPR can help us to understand the decisions it makes. In the following sections, we will analyze the insight into activity of sgRNA bring through the attention mechanism at both global and local levels.

In the following content, when we use the word, global, we are referring to rules thats are common throughout the data set. If assumed the method with strong generalization ability, it might mean that these rules is common in the corresponding Cas9 system. On the other hand, local refers to the rules applicable to a certain sgRNA. We believe that the locally interpretable is very helpful in the design and optimization of sgRNA.

3.3.1 Globally interpretable

At global level, a important question we want AttCRISPR to answer is which nucleotide it prefers at each position on the sequence. In fact, Wang *et al.* has already answered this question in detail with the DeepSHAP method. Leaving aside nucleotides, just consider the importance of each position, the method of Liu *et al.* goes some way to answering this question.

3.3.2 Locally interpretable

Text

4 Discussion

Text

5 Conclusion

Text

Acknowledgements

Text

Funding

text.

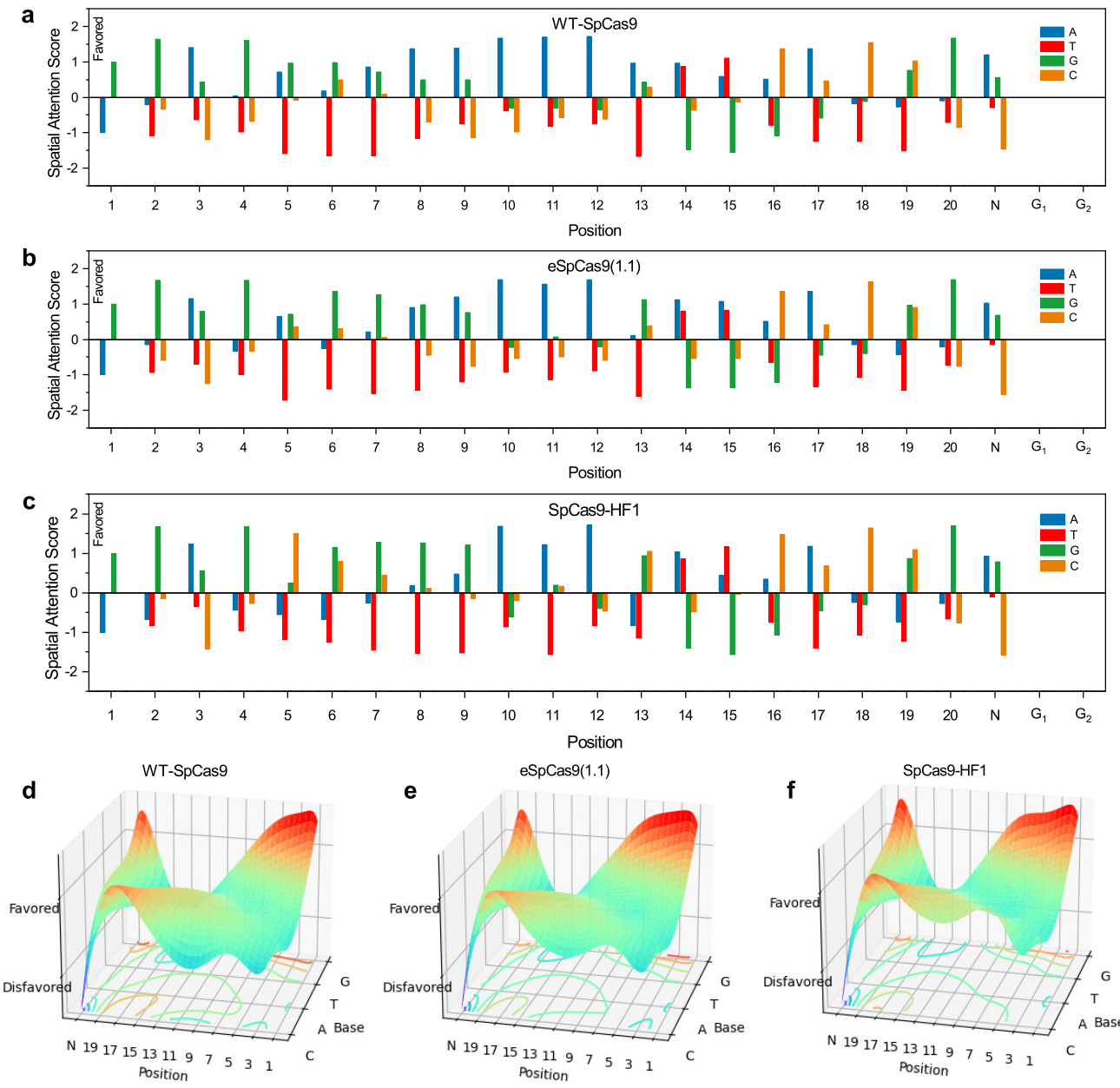

Fig. 8. wait2write

References

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv: Computation and Language*.

Chaudhari, S., Polatkan, G., Ramanath, R., and Mithal, V. (2019). An attentive survey of attention models. *arXiv: Learning*.

Chuai, G., Ma, H., Yan, J., Chen, M., Hong, N., Xue, D., Zhou, C., Zhu, C., Chen, K., Duan, B., *et al.* (2018). Deepcrispr : optimized crispr guide rna design by deep learning. *Genome Biology*, **19**(1), 1–18.

Cong, L., Ran, F. A., Cox, D. D., Lin, S., Barretto, R. P. J., Habib, N., Hsu, P., Wu, X., Jiang, W., Marraffini, L. A., *et al.* (2013). Multiplex genome engineering using crispr/cas systems. *Science*, **339**(6121), 819–823.

Huang, J., Zhao, W. X., Dou, H., Wen, J., and Chang, E. Y. (2018). Improving sequential recommendation with knowledge-enhanced memory networks. pages 505–514.

Ishii, T. (2017). Reproductive medicine involving genome editing: clinical uncertainties and embryological needs. *Reproductive Biomedicine Online*, **34**(1), 27–31.

Jinek, M., Chylinski, K., Fonfara, I., Hauer, M., Doudna, J. A., and Charpentier, E. (2012). A programmable dual-rna-guided dna endonuclease in adaptive bacterial immunity. *Science*, **337**(6096), 816–821.

Kang, X., He, W., Huang, Y., Yu, Q., Chen, Y., Gao, X., Sun, X., and Fan, Y. (2016). Introducing precise genetic modifications into human 3pn embryos by crispr/cas-mediated genome editing. *Journal of Assisted Reproduction and Genetics*, **33**(5), 581–588.

Kim, H. K., Min, S., Song, M., Jung, S., Choi, J. W., Kim, Y., Lee, S., Yoon, S., and Kim, H. (2018). Deep learning improves prediction of crispr-cpf1 guide rna activity. *Nature Biotechnology*, **36**(3), 239–241.

Kim, H. K., Kim, Y., Lee, S., Min, S., Bae, J. Y., Choi, J. W., Park, J., Jung, D., Yoon, S., and Kim, H. (2019). Spcas9 activity prediction by deepspcas9, a deep learning–based model with high generalization performance. *Science Advances*, **5**(11).

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

Kleinstiver, B. P., Pattanayak, V., Prew, M. S., Tsai, S. Q., Nguyen, N. T., Zheng, Z., and Joung, J. K. (2016). High-fidelity crispr–cas9 nucleases with no detectable genome-wide off-target effects. *Nature*, **529**(7587), 490–495.

Li, Z. and Yu, Y. (2016). Protein secondary structure prediction using cascaded convolutional and recurrent neural networks. *ArXiv*, **abs/1604.07176**.

Liang, P., Xu, Y., Zhang, X., Ding, C., Huang, R., Zhang, Z., Lv, J., Xie, X., Chen, Y., Li, Y., *et al.* (2015). Crispr/cas9-mediated gene editing in human tripronuclear zygotes. *Protein & Cell*, **6**(5), 363–372.

Lin, J. and Wong, K. (2018). Off-target predictions in crispr-cas9 gene editing using deep learning. *Bioinformatics*, **34**(17).

Liu, G., Zhang, Y., and Zhang, T. (2019a). Computational approaches for effective crispr guide rna design and evaluation. *Computational and structural biotechnology journal*, **18**, 35–44.

Liu, Q., He, D., and Xie, L. (2019b). Prediction of off-target specificity and cell-specific fitness of crispr-cas system using attention boosted deep learning and network-based gene feature. *PLoS computational biology*, **15**(10), e1007480–e1007480. 31658261[pmid].

Liu, Q., Cheng, X., Liu, G., Li, B., and Liu, X. (2020). Deep learning improves the ability of sgrna off-target propensity prediction. *BMC Bioinformatics*, **21**(1), 51.

Lundberg, S. and Lee, S. (2017). A unified approach to interpreting model predictions. pages 4768–4777.

Luong, M., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *arXiv: Computation and Language*.

Mali, P., Yang, L., Esvelt, K. M., Aach, J., Guell, M., Dicarlo, J. E., Norville, J. E., and Church, G. M. (2013). Rna-guided human genome engineering via cas9. *Science*, **339**(6121), 823–826.

Muhammad Rafid, A. H., Toufikuzzaman, M., Rahman, M. S., and Rahman, M. S. (2020). Crisprpred(seq): a sequence-based method for sgrna on target activity prediction using traditional machine learning. *BMC Bioinformatics*, **21**(1), 223.

Rubeis, G. and Steger, F. (2018). Risks and benefits of human germline genome editing: An ethical analysis. *Asian Bioethics Review*, **10**(2), 133–141.

Slack, D., Hilgard, S., Jia, E., Singh, S., and Lakkaraju, H. (2019). Fooling lime and shap: Adversarial attacks on post hoc explanation methods. *arXiv: Learning*.

Slaymaker, I. M., Gao, L., Zetsche, B., Scott, D. A., Yan, W. X., and Zhang, F. (2016). Rationally engineered cas9 nucleases with improved specificity. *Science*, **351**(6268), 84–88.

Song, M., Kim, H. K., Lee, S., Kim, Y., Seo, S.-Y., Park, J., Choi, J. W., Jang, H., Shin, J. H., Min, S., Quan, Z., Kim, J. H., Kang, H. C., Yoon, S., and Kim, H. H. (2020). Sequence-specific prediction of the efficiencies of adenine and cytosine base editors. *Nature Biotechnology*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. pages 5998–6008.

Wang, D., Zhang, C., Wang, B., Li, B., Wang, Q., Liu, D., Wang, H., Zhou, Y., Shi, L., Lan, F., *et al.* (2019). Optimized crispr guide rna design for two high-fidelity cas9 variants by deep learning. *Nature Communications*, **10**(1), 4284–4284.

Wang, S., Peng, J., Ma, J., and Xu, J. (2016). Protein secondary structure prediction using deep convolutional neural fields. *Scientific Reports*, **6**.

Wang, T., Wei, J. J., Sabatini, D. M., and Lander, E. S. (2014). Genetic screens in human cells using the crispr/cas9 system. *Science*, **343**(6166), 80–84.

Woo, S., Park, J., Lee, J., and Kweon, I. S. (2018). Cbam: Convolutional block attention module. pages 3–19.

Wu, X., Scott, D. A., Kriz, A. J., Chiu, A. C., Hsu, P., Dadon, D. B., Cheng, A. W., Trevino, A. E., Konermann, S., Chen, S., *et al.* (2014). Genome-wide binding of the crispr endonuclease cas9 in mammalian cells. *Nature Biotechnology*, **32**(7), 670–676.

Zhang, G., Dai, Z., and Dai, X. (2020). C-rnncrispr: Prediction of crispr/cas9 sgrna activity using convolutional and recurrent neural networks. *Computational and structural biotechnology journal*, **18**, 344–354.

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture