Bioinformatics, 34, 2018, i656–i663
doi: 10.1093/bioinformatics/bty554
ECCB 2018

Off-target predictions in CRISPR-Cas9 gene editing using deep learning

Jiecong Lin and Ka-Chun Wong*

Department of Computer Science, City University of Hong Kong, Kowloon Tong, Hong Kong SAR

*To whom correspondence should be addressed.

Abstract

Motivation: The prediction of off-target mutations in CRISPR-Cas9 is a hot topic due to its relevance to gene editing research. Existing prediction methods have been developed; however, most of them just calculated scores based on mismatches to the guide sequence in CRISPR-Cas9. Therefore, the existing prediction methods are unable to scale and improve their performance with the rapid expansion of experimental data in CRISPR-Cas9. Moreover, the existing methods still cannot satisfy enough precision in off-target predictions for gene editing at the clinical level.
Results: To address it, we design and implement two algorithms using deep neural networks to predict off-target mutations in CRISPR-Cas9 gene editing (i.e. deep convolutional neural network and deep feedforward neural network). The models were trained and tested on the recently released off-target dataset, CRISPOR dataset, for performance benchmark. Another off-target dataset identified by GUIDE-seq was adopted for additional evaluation. We demonstrate that convolutional neural network achieves the best performance on CRISPOR dataset, yielding an average classification area under the ROC curve (AUC) of 97.2% under stratified 5-fold cross-validation. Interestingly, the deep feedforward neural network can also be competitive at the average AUC of 97.0% under the same setting. We compare the two deep neural network models with the state-of-the-art off-target prediction methods (i.e. CFD, MIT, CROP-IT, and CCTop) and three traditional machine learning models (i.e. random forest, gradient boosting trees, and logistic regression) on both datasets in terms of AUC values, demonstrating the competitive edges of the proposed algorithms. Additional analyses are conducted to investigate the underlying reasons from different perspectives.
Availability and implementation: The example code are available at https://github.com/MichaelLinn/off_target_prediction. The related datasets are available at https://github.com/MichaelLinn/off_target_prediction/tree/master/data.
Contact: kc.w@cityu.edu.hk

1 Introduction

CRISPR-Cas9 is a well-sought technology for precise gene editing (Cong *et al.*, 2013; Esvelt *et al.*, 2013; Mali *et al.*, 2013b; Ran *et al.*, 2013). With single-guide RNA and Cas9 protein, specific genomic fragments are able to be inserted, deleted or replaced (Al-Attar *et al.*, 2011; Chen *et al.*, 2017a; Hsu *et al.*, 2013; Klann *et al.*, 2017; Shalem *et al.*, 2014; Shibata *et al.*, 2017; Zhu, 2015). Therefore, CRISPR-Cas9 holds the potential to edit and renovate the harmful genes for personalized therapy (Kang *et al.*, 2017; Liang *et al.*, 2015; Manguso *et al.*, 2017; Wu *et al.*, 2013). Recently, a pathogenic gene mutation was corrected in human embryos (Ma *et al.*, 2017). Moreover, CRISPR can help us analyse the genetic interactions and the relationships between genetic variations and phenotypes (Cox *et al.*, 2015; Doudna and Charpentier, 2014; Hsu *et al.*, 2014; Shapiro *et al.*, 2018; Shen *et al.*, 2017; Smith *et al.*, 2015). There is no doubt that CRISPR-Cas9 will be critically important in the coming years.

Although specific fragments of DNA are aimed, sgRNA can sometimes influence other regions and incur off-target mutations (Chen *et al.*, 2017b). CRISPR-Cas9 can tolerate mismatches in sgRNA-DNA at different positions in a sequence-dependent manner; it is sensitive to the number, positions and distribution of mismatches (Hsu *et al.*, 2013; Kim *et al.*, 2015; Zhang *et al.*, 2015).

Off-target mutations can lead to genomic instability and disturb the normal gene functions; it is still a major problem when applying CRISPR-Cas9 gene editing to clinical applications (Cho *et al.*, 2014;

Corrigan-Curay *et al.*, 2015; Fu *et al.*, 2013; Hsu *et al.*, 2013; Mali *et al.*, 2013a; Pattanayak *et al.*, 2013). Consequently, we still need accurate off-target prediction methods for complementary purposes.

Most of the existing off-target prediction methods just calculate scores based on the positions of the mismatches to the guide sequence (Haeussler *et al.*, 2016; Xu *et al.*, 2017). The score of each base pair in sgRNA-DNA is derived using the statistical analysis of the mismatch effects based on previous gene editing experiments. For example, CFD (cutting frequently determination) score is derived by emulating a large number of sgRNAs with single-bp (single base pair) replacements, deletions or insertions with reference to the validated sgRNAs in MOLM13 cells; it calculates the percentage activity rates of different mutation sites based on LFC (log fold change) value (Doench *et al.*, 2016). CROP-IT score method (Singh *et al.*, 2015) grades the putative off-target sgRNA sequences by dividing each 23 bp sequence into three regions with different weights; it also proposes the penalty scores for the consecutive mismatched sites. CCTop score (Stemmer *et al.*, 2015) and MIT score (Hsu *et al.*, 2013) only considered the positions and counts of the mismatched sites of sgRNA-DNA as the features to score the potential off-targets.

In light of the above, their performance is vulnerable to experimental variation. Most importantly, the existing methods cannot take advantage of the growing CRISPR-Cas9 data for continuous self-learning. In addition, most of the existing methods do not consider the potential relationships between mismatched and matched sites, which may affect the off-target activity in CRISPR-Cas9 gene editing (Xu *et al.*, 2017).

The recent application of deep learning to sequence-based problems in genomics signifies its applicability (Zeng *et al.*, 2016). Examples of deep learning applications include alternative splicing predictions, binding target predictions for regulatory proteins, protein secondary structure and biomedical image analysis. In particular, recurrent neural networks (RNN), convolutional neural networks (CNN) and long short-term memory (LSTM) have been demonstrated successful (Almagro Armenteros *et al.*, 2017; Hou *et al.*, 2017; Jurtz *et al.*, 2017).

However, there is not any deep learning application for off-target prediction for CRISPR-Cas9 gene editing so far. The past successes of deep neural networks in molecular genetics inspire us to extend the applications to off-targets prediction in CRISPR-Cas9.

In this article, we took advantage of deep learning and developed two deep neural networks models to address the current problems including feedforward neural network (FNN) and CNN for off-target predictions.

The adaption of CNN from computer vision to genetic sequence can be accomplished by considering each sgRNA-DNA sequence pair as an image. Instead of processing 2-dimensional image with colour channels, we consider a genomic sequence as a $4{\times}L$ matrix where 4 is the number of the nucleotide types and L is the length of sequence. Therefore, we adopted a new encoding method to transfer each sgRNA-DNA sequence pair with length of 23 into a 4×23 matrix.

The following major contributions are made:

1. We develop a feasible sequence encoding method that converts each sgRNA-DNA sequence pair into a matrix with the shape of 4×23 as a convolutional input and make the first attempt to apply deep FNN and deep CNN to off-targets prediction in CRISPR-Cas9 gene editing.
2. We have tested a series of deep neural networks with different architectures and constructed deep CNN for the off-target prediction that outperforms the current state-of-art prediction methods on both the CRISPOR dataset and GUIDE-seq dataset.

2 Materials and methods

2.1 Sequence encoding

For encoding, the complementary base is designed to represent the original base in sgRNA; for instance, we can use A, G, C, T to represent both sgRNA and target DNA sequence in CRISPR-Cas9. Therefore, each base in the sgRNA and target DNA can be encoded as one of the four one-hot vectors [1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0] and [0, 0, 0, 1].

As a result, every sgRNA-DNA sequence pair can be represented by a 4×23 matrix where 23 is the length of the sequence which includes the 3-bp PAM adjacent to the 20 bases. To encode the mutated information in sgRNA-DNA, we derived a 4-length vector to encode the mismatched base pairs by implementing OR operator on two one-hot vectors of base-pairing. The code matrix of sgRNA-DNA will be directly fed into CNN-based models for training and testing, while the vectorization of the encoding matrix will be used as the input of traditional machine-learning-based models and deep FNN.

2.2 Neural network models

Figure 2 and the following description gives a summary of the basic architectural structure of CNN used: the input is a code matrix (e.g. Figure 1) with shape of 23 (sequence length) × 4 (size of nucleotides vocabulary).

The first layer of our network is a convolutional layer, which is designed for extracting sgRNA-DNA matching information using 40 filters of different sizes (10 for each of the sizes 4×1, 4×2, 4×3 and 4×5). To preserve the integrity of every base pair code in gRNA-DNA, the size of scanning step for each filter is set to 4 in the dimension of base pair. Thus it gives a $1 \times 23 \times 40$ feature map from this layer.

The second layer is a batch normalization (BN) layer, which is designed for reducing internal covariate shift in the neural network (Sergey and Christian, 2015). It further prevents smaller changes to the parameters to amplify and thereby allows higher learning rates than the opposite case. Moreover, ReLU (Glorot *et al.*, 2011) is used as the activation function for each neuron in this layer.

The third layer is a global max-pooling layer connected with the previous BN layer. Each of these max-pooling windows only outputs the maximum value of all of its respective BN layer outputs. The size of each pooling window used in standard CNN (CNN_std) is 1×5; other sizes are also tested in the following experiments. Accordingly, it gives a $1 \times 5 \times 40$ feature map to the next layers. The function of this global max-pooling can be thought as calling whether the mismatches modelled by the respective BN layer exist in the input sequence or not.

The following layers are two fully connected dense layers with the sizes of 100 and 23, respectively. A dropout layer is used on the last dense layer to randomly mask portions of the output to avoid overfitting; the probability used in CNN_std to drop a unit is 0.15 (Srivastava *et al.*, 2014). The final output layer consists of two neurons corresponding to the two classification results. Those two neurons are fully connected to the previous layers.

The architecture of FNN used for off-target prediction consists of input layer, several hidden layers and output layer. The input of FNN model is a vector with the length of 92, the vectorization of 4×23 matrix. The activation function is softmax which is able to

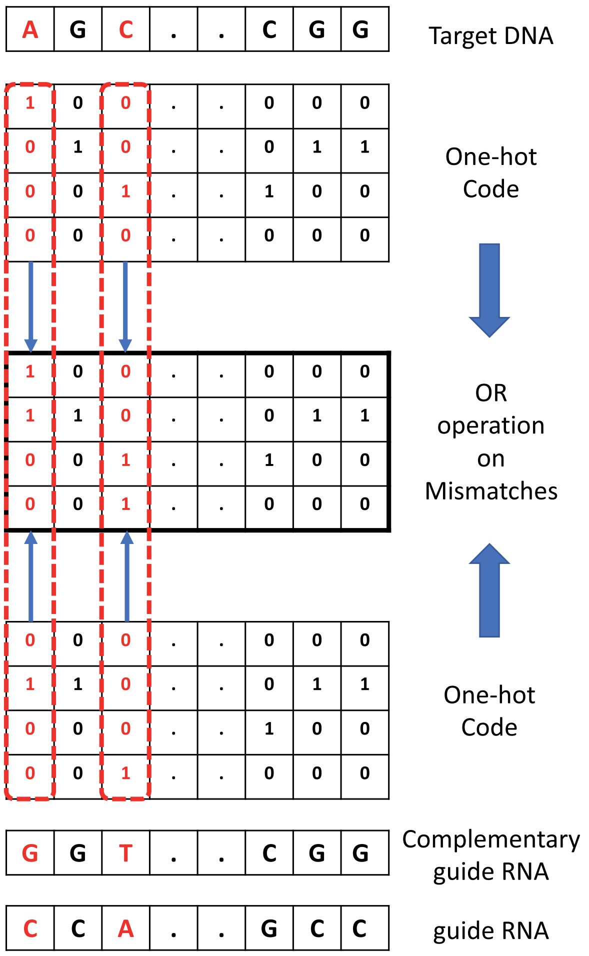


Fig. 1. An example on how to encode a sgRNA-DNA sequence pair. The table with thick borders in the middle of the figure shows the final code matrix of a sgRNA-DNA sequence pair, which can be used as the input for CNN modelling

convert each neuron output into probability. Accordingly cross-entropy (Lih-Yuan, 2006) is chosen as the loss function for our FNN.

2.3 Current prediction algorithms

To the best of our knowledge, there are four most recent off-target prediction algorithms including CFD score, MIT score, CROP-IT score and CCTop score. Note that MIT-web score is a website version of the MIT-score. In order to study whether the potential off-target is validated or not, these methods calculate scores based on the positions or identities of the mismatches to the guide RNA sequence. Higher score means this sequence is more likely to be an active off-target than the control. Initially, systematic testing of the effect of mismatches led to a weight for each possible nucleotide change at each position. The score of MIT website is based on these data but reduced to one weight per position. CCTop and CROP-IT, respectively, designed a heuristics approach based on the distances of the mismatches to the PAM (protospacer adjacent motif). Moreover, all scoring methods except CCTop and CFD propose a penalty score for consecutive mismatches (Haeussler *et al.*, 2016).

CFD score is based on the biggest cleavage data up to date. These data are obtained by infecting cells with a lentiviral library containing thousands of guides for all possible nucleotide mismatches and single-bp indels at all possible positions. The core idea is to calculate the percent activity rates for all sgRNAs sharing the same nucleotide mutation at the same position with independence assumption.

Haeussler et al. compared the performance of these four algorithms (Haeussler *et al.*, 2016). According to the experiment, the state-of-art off-target prediction method, CFD score, performed the best with an area under the ROC curve (AUC) of 0.91. The MIT score as calculated by CRISPOR website is slightly less discriminative than CFD score with an AUC of 0.87.

2.4 Datasets

2.4.1 CRISPOR dataset

The main dataset that we used for training, validation and testing was from CRISPOR (Haeussler *et al.*, 2016). Haeussler et al. also provided a tool for guide RNA selection in 120 genomes (including plants and emerging models or organisms) and pre-calculated results for all human coding exons. There are 26 034 putative off-targets including 143 validated off-targets identified by CRISPOR. Each of these off-targets has a mismatch count of up to four with one of the PAMs: NAG/NGA/NGG and a minimum modification frequency of 0.1%.

Owing to the data unbalance, we stratified 5-fold cross-validation to evaluate the performance of our deep neural networks (i.e. FNN and CNN) where each fold contains roughly the same proportions of class labels.

2.4.2 GUIDE-seq dataset

GUIDE-seq is the most rigorous framework for genome-wide identification of off-target effects to date (Tsai *et al.*, 2015). GUIDE-seq with Cas9 and 10 different sgRNA targeted at various endogenous human genes in either U2OS or HEK293 human cell lines. Guide RNAs used in GUIDE-seq targets the following sites: VEGFA site 1, VEGFA site 3, VEGFA site 2, FANCF, HEK293 site 2, HEK293 site 3 and HEK293 site 4, in which 28 off-targets with a minimum modification frequency of 0.1% among 403 potential off-targets identified by GUIDE-seq.

We use this dataset to independently evaluate and compare the performance between deep neural network models and the current state-of-art off-targets prediction methods. GUIDE-seq dataset was excluded from the CRISPOR dataset used for training.

2.5 Experiments

Two different sets of experiments were carried out. The first experiment was designed for model selection through comparing the performance of deep neural networks with different architectures as tabulated in Table 1 under stratified 5-fold cross-validation on the CRISPOR dataset.

All these neural network models were trained and validated on the CRISPOR dataset. The architectures of FNN and CNN with the best performance under 5-fold stratified cross-validation were adopted in the subsequent experiments. We constructed three different FNN models by varying the number of the hidden layers with fixed total number of neurons. As for CNN model selection, we constructed six CNN by varying one of the following parameters: BN, drop-out layer and the window size of pooling layer. The architecture of standard convolution neural network (CNN_std) served as a control as depicted in Figure 2.

The second experiment is designed to compare the performance among deep neural networks models, existing off-target prediction algorithms, and three traditional machine learning methods [i.e. random forest (RF), gradient boosting trees (GBT) and logistic regression (LR)].

For both FNN and CNN models, we use Adam algorithms (Kingma and Ba, 2014) to optimize the cross-entropy loss function. Mini-batch gradient descent is adopted for optimization which can further reduce the gradient variance. The size of batch is 100 for both models. Each model was run for 200 epochs (epoch = full pass over the training set) with learning rate at 0.0001.

For RF, we used 100 CART (classification and regression tree) as the individual classifiers. The maximum depth of each CART is confined to 3, avoiding model over-fitting.

CART is also used for GBT as the individual estimator. The loss function used for training is logistic loss function and the number of estimators is 200.

In our second experiment, both GUIDE-seq dataset and CRISPOR dataset were used for performance evaluation. The performance measurements used to assess the performance of our models are ROC (Receiver Operating characteristic) curve and its AUC value.

The traditional machine learning models were implemented in Python 2.7.13 using scikit-learn library (Pedregosa *et al.*, 2011). The neural network models were implemented in TensorFlow 1.4.1 (Abadi *et al.*, 2016) and one Tesla K80 GPU was used for training and testing.

Table 1. The naming convention code and brief description of the variants of CNN and FNN models compared in this work

Model	Code	Architecture
FNN	FNN_2layer	Using 2 hidden layers with 250 × 40 neurons
FNN	FNN_3layer	Using 3 hidden layers with 50 × 20×10 neurons
FNN	FNN_4layer	Using 4 hidden layers with 25 × 10×10 × 4 neurons
CNN	CNN_std	The basic structure as depicted in Figure 2
CNN	CNN_nbn	Without using BN layer
CNN	CNN_nd	Without drop-out layer
CNN	CNN_np	Without using max-pooling layer
CNN	CNN_pwin3	Using max-pooling layer of window size 3
CNN	CNN_pwin7	Using max-pooling layer of window size 7

Note: The descriptions of all CNN models are relative to CNN_std model depicted in Figure 2.

3 Results

We designed experiments to address the following questions:

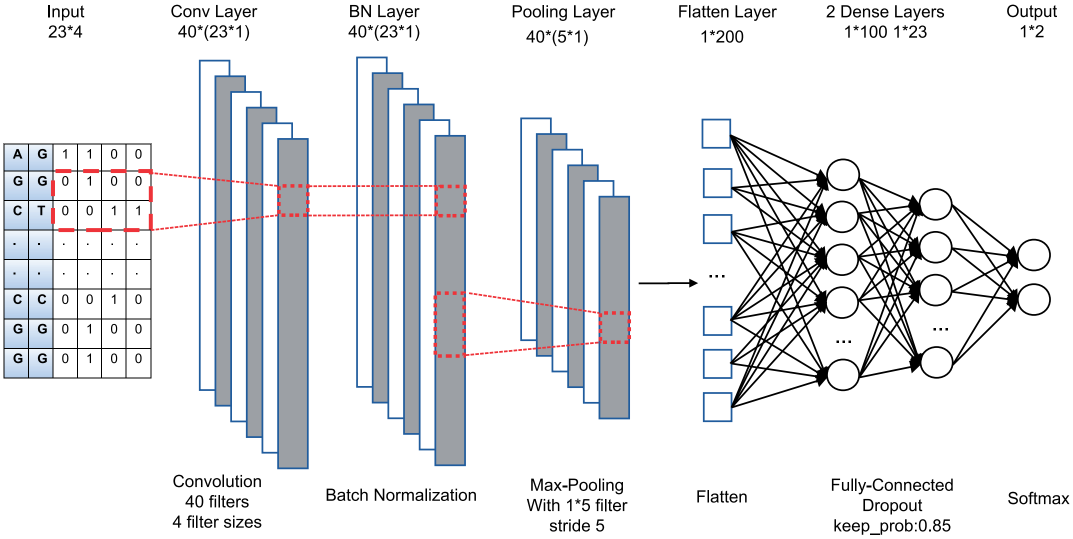
- What are the relative performance of the proposed neural network model architectures? → Section 3.1
- How does final Deep Neural Network models compare with current state-of-the-art off-target predictions and traditional machine learning models? → Sections 3.2 and 3.3
- What are the generalization performance on GUIDE-seq dataset if the models are trained on CRISPOR dataset? → Section 3.4

3.1 Model selection

In Table 2, we compare the performance of different model architectures trained on the CRISPOR dataset. FNN_3layer and CNN_std achieved the best performance predicting off-targets under stratified 5-fold cross-validation with the mean AUCs of 0.970 and 0.972, respectively.

Based on the validation results of the CNN models (i.e. CNN_std, CNN_np, CNN_pool_win3 and CNN_pool_win3) used for pooling layer testing, we found that pooling layer significantly increased performance for off-targets prediction. In particular, the CNN with pooling layer of window size 5 (CNN_std) achieved the best performance; such observation emphasizes the need to use a pooling layer with befitting window size to extract the mismatches in sgRNA-DNA.

Comparing CNN_std with CNN_nbn, we see that BN improves the performance. This is expected since BN can stabilize the training process and decrease the risk of overfitting. The performance of CNN_nd comparing with the CNN_std shows that drop-out layer can slightly improve the generalization performance.



A	G	1	1	0	0
G	G	0	1	0	0
C	T	0	0	1	1
.	.	.	.	.	.
.	.	.	.	.	.
C	C	0	0	1	0
G	G	0	1	0	0
G	G	0	1	0	0

Fig. 2. The architecture of standard deep CNN (CNN_std) for off-target prediction. The input of this deep neural network is the encoded sgRNA-DNA sequence with length 23. The convolutional layer consists of 40 filers including 10 for each of the sizes 4 × 1, 4 × 2, 4 × 3 and 4 × 5. The BN layer is used to normalize the output of the convolutional layer to speed up learning and avoid over-fitting. The global max-pooling layer applies a filter with window size 5 to the previous layers. The outputs of max-pooling layer are joined together into one vector by flattening. Each neurons in the flatten layer is fully connected to the first dense layer. Two dense layers consist of 100 and 23 neurons, respectively. The second dense layers with a drop-out layer is fully connected to two output neurons to predict whether the input pair is off-target or not as binary class probabilities. The neurons in output layer and dense layers use softmax function as the activation function, while all the neurons in other layers use ReLU as the activation function

Table 2. Performance comparisons for different architectures under stratified 5-fold cross-validation on CRISPOR dataset

Model	Min_AUC	Max_AUC	Mean_AUC	Var_AUC
FNN_2layer	0.852	0.891	0.842	0.010
FNN_3layer	**0.963**	0.977	**0.970**	**0.005**
FNN_4layer	0.951	0.960	0.954	0.009
CNN_std	0.954	**0.983**	**0.972**	**0.010**
CNN_nbn	0.929	0.973	0.954	0.022
CNN_nd	0.953	0.974	0.969	0.013
CNN_np	0.720	0.981	0.899	0.093
CNN_pool_win3	0.632	0.979	0.903	0.137
CNN_pool_win7	0.943	0.983	0.967	0.015

Bold values signifies: CNN_std achieved the highest Mean_AUC (0.972) and highest Max_AUC (0.983) under stratified 5-fold cross-validation in predicting off-targets among all neural network based models, FNN_3layer also accomplished the competitive performance (Mean_AUC = 0.970) with the highest Min_AUC (0.963) and the lowest AUC variance (0.005).

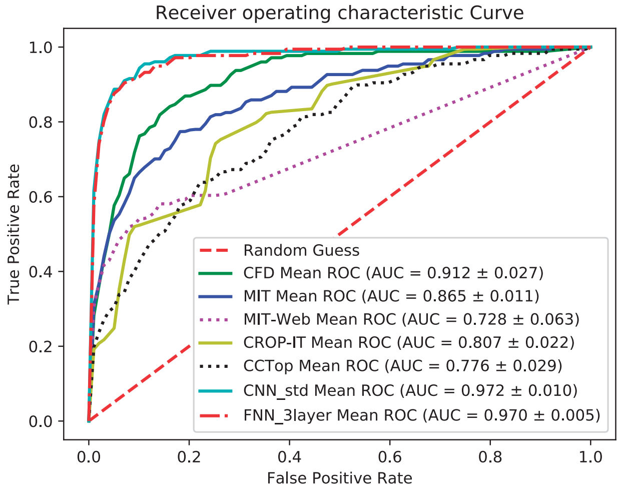


Fig. 3. ROC curves of two deep learning models and five state-of-the-arts prediction methods under stratified 5-fold cross-validation on CRISPOR dataset

Comparing FNN with CNN, we found that the average Mean_AUC of CNN models is higher than that of FNN models while FNNs had lower average standard deviance. Since the comparison between CNN_std and FNN_3layer is too close to call, we decided to use both of them for the rest of the experiment.

3.2 Comparison to current algorithms

The performance of FNN_3layer and CNN_std were compared with the current off-targets prediction models: CFD score, MIT score, MIT web score, CROP-IT score and CCTop score under stratified 5-fold cross-validation on CRISPOR dataset.

Figure 3 shows the AUC values and the ROC curves of the current prediction methods and two deep learning models, CNN_sd and FNN_3layer. It is hardly surprising that CFD score achieved the best performance with AUC of 0.912 among all current off-targets prediction models, since the experiments from Haeussler et al. have already proved that CFD score was the best prediction model on CRISPOR dataset.

However, our two deep learning models, CNN_std and FNN_3layer, achieved much better performance than all current off-targets prediction models in both ROC curves and AUC values. The AUCs of our CNN and FNN models are roughly 5.8% higher than CFD score under stratified 5-fold cross-validation on CRISPOR dataset, reaching 0.972 and 0.970, respectively.

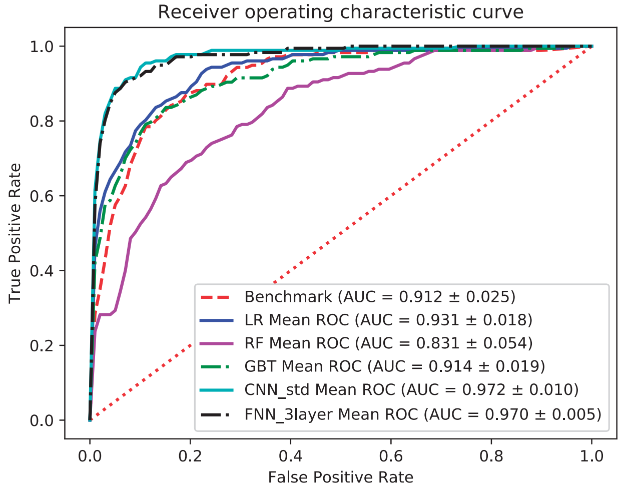


Fig. 4. ROC curves of two deep learning models (i.e. FNN_3layer and DNN_std) and three traditional machine learning models including LR, RF and GBT. The ROC curve and AUC value of CFD score were regarded as the state-of-arts benchmark on the figure

Furthermore, Figure 3 shows that the ROC curves of both deep learning models completely covered the ROC curve of CFD score. These results reveal that our deep learning approaches have competitive edges over the existing methods on the CRISPOR dataset.

3.3 Comparison to traditional machine learning models

Since the current off-target prediction models rely on the fixed scores to represent the mismatched information to evaluate the potential off-target sites, they do have the ability to improve their performance by training. Therefore, we decided to implement some traditional machine-learning models including LR, RF, and GBT for further comparison. Three traditional and two deep learning models were trained and tested on CRISPOR dataset under stratified 5-fold cross-validation.

Figure 4 shows that our deep learning models remained the best off-targets predictors, achieving top two AUC values among all models. Moreover, the standard deviations of two deep learning models are the lowest among all models; it reveals that our deep learning models are more stable than the traditional machine learning methods and current prediction models on the CRISPOR dataset.

In addition, we found that LR and GBT achieved slightly better performance with AUCs of 0.931 and 0.914 than CFD score. The observations confirm that machine-learning-based method still have good potential in off-target predictions for CRISPR-Cas9 gene editing.

3.4 Performance on GUIDE-seq dataset

To compare the generalization performance among current state-of-art models, deep learning models and three traditional machine learning models, we trained FNN_3layer, CNN_std and three traditional machine learning models on the whole CRISPOR dataset and compared their performance with current state-of-art prediction model, CFD score, on the GUIDE-seq dataset.

Figure 5 shows that CNN_std achieved the highest AUC valued at 0.881 among all prediction models. LR achieved the second best

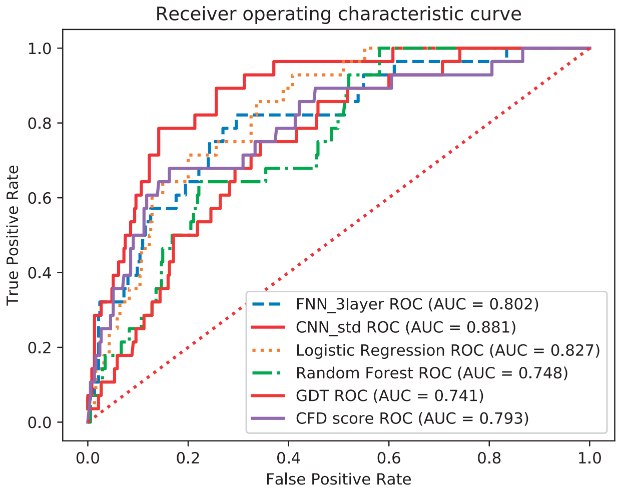


Fig. 5. ROC curves of deep learning models, CFD score and three traditional machine learning models on GUIDE-seq dataset

performance with AUC of 0.82. Surprisingly, the AUC of the current state-of-art prediction, CFD score, only reached 0.793 and ranked fourth. Considering the shapes of ROC curves, CNN_std's ROC curve is on the top of other ROC curves; it implies CNN_std always achieve the highest true positive rate among all prediction models when the false positive rate is fixed.

These results demonstrate that CNN_std has the best generalization performance among current prediction models, three traditional machine learning models and deep FNN. Moreover, CNN_std outperformed the current state-of-the-art prediction model, CFD score, on both CRISPOR dataset and GUIDE-seq dataset.

4 Discussion

In this article, we have introduced a new encoding method for transforming each sgRNA-DNA sequence pair into a matrix with the shape of 4 × 23 that can be used as the input of CNN. Second we have made the first attempt to apply deep neural networks to off-target predictions in CRISPR-Cas9 gene editing; it provided us a deep CNN-based off-target prediction model, achieving competitive performance on both CRISPOR dataset and GUIDE-seq dataset shown in Table 3. In addition, the experiments of deep neural networks' constructions identified the model with the best performance in off-target prediction after varying the layer types, size of pooling windows, BN and convolutional layer designs.

The comparison of the generalization performance for FNN and CNN showed that our CNN trained on the CRISPOR dataset generalized much better than the deep forward neural network on the GUIDE-seq dataset although they have achieved roughly the same performance under stratified 5-fold cross-validation on the CRISPOR dataset. Possible reasons include: first, the convolutional layer can be thought as a mismatch site scanner; four different sizes of scanning window used in this layer can capture the locations and the density of the mismatches in a certain range according to the size of the convolutional kernel. Through training, the scanning windows with different sizes can be iteratively adjusted and weighted. Such a mechanism is similar to the CROP-IT which scores the potential off-targets by dividing the sequence into regions with different scoring weights. However, deep feedforward neural can only assign weights to the whole sgRNA-DNA pair. Second, the drop-out layer and BN are advantageous to the generalization performance.

Table 3. Performance comparison among deep learning models, traditional machine learning models and CFD score

Model	Guide-seq dataset	CRISPOR dataset	
	AUC	Mean AUC	Var AUC
CNN_std	**0.881**	**0.972**	0.010
FNN_3layer	0.802	0.970	**0.005**
CFD score	0.793	0.912	0.027
Logistic regression	0.827	0.931	0.018
Gradient boosting trees	0.741	0.914	0.019

Significance of bold values and the comment on value "0.010": On both CRISPOR and GUIDE-seq datasets, CNN_std achieved the highest Mean_AUC (0.881 and 0.972) among current off-target predictions methods and machine-learning based models. Although FNN_3layer had lower variance (0.005) than CNN_std (0.010) under stratified 5-fold cross validation on CRISPOR dataset, the AUC of CNN_std (0.881) is 9.8% higher than FNN_3layer's (0.802) on GUIDE-seq dataset, which means CNN_std has much better generalization than FNN_3layer does. Above all, the defects cannot obscure the virtues, CNN_std is a preferable off-target predictions model.

Table 4. Performance comparison among different architectures under stratified 5-fold cross-validation on CRISPOR dataset

Model	Mean_AUC	Var_AUC
FNN_3layer	0.970	**0.005**
CNN_std	**0.972**	0.010
CROP-IT score	0.807	0.022
CFD score	0.912	0.027
MIT score	0.865	0.011
CCTop score	0.776	0.029
MIT-web score	0.728	0.063

Bold values signifies: FNN_3layer and CNN_std achieved the best performance with the mean AUC of 0.970 and 0.972 in predicting off-targets under stratified 5-fold cross-validation. Moreover, these two deep neural network based models (i.e. CNN_std and FNN_3layer) obtained more stable performances because of the lowest variances (0.010 and 0.005) than the other off-target predictions methods under stratified 5-fold cross validation on CRISPOR dataset.

Comment on value "0.010": The Mean_AUC of CNN_std (0.972) is the higher than that of FNN_3layer's (0.970) while FNN_3layer had lower standard deviance (0.005) than CNN_std (0.010). Since the comparison between CNN_std and FNN_3layer is too close to call, we decided to use both of them for further experiments on GUIDE-seq dataset.

We compared the performance of the deep neural networks with three traditional machine-learning based algorithms (i.e. LR, RF and GBT). Although those three machine learning algorithms have been widely adopted and achieved great success, our final deep convolutional performed significantly better than those three machine learning approaches on both CRISPOR dataset and GUIDE-seq dataset. The observation emphasized that deep learning can automatically elucidate the features which are important in complex sequences for off-target predictions. In contrast, for traditional machine learning algorithm, we had to exhaustively hand-craft features for performance gain.

We noted that we also compared the performance of the final deep CNN and deep FNN with other existing prediction methods in Table 4. We found that our CNN outperforms significantly than the other approaches including the state-of-the-art off-target prediction method, CFD score, on the CRISPOR dataset. It is not surprising

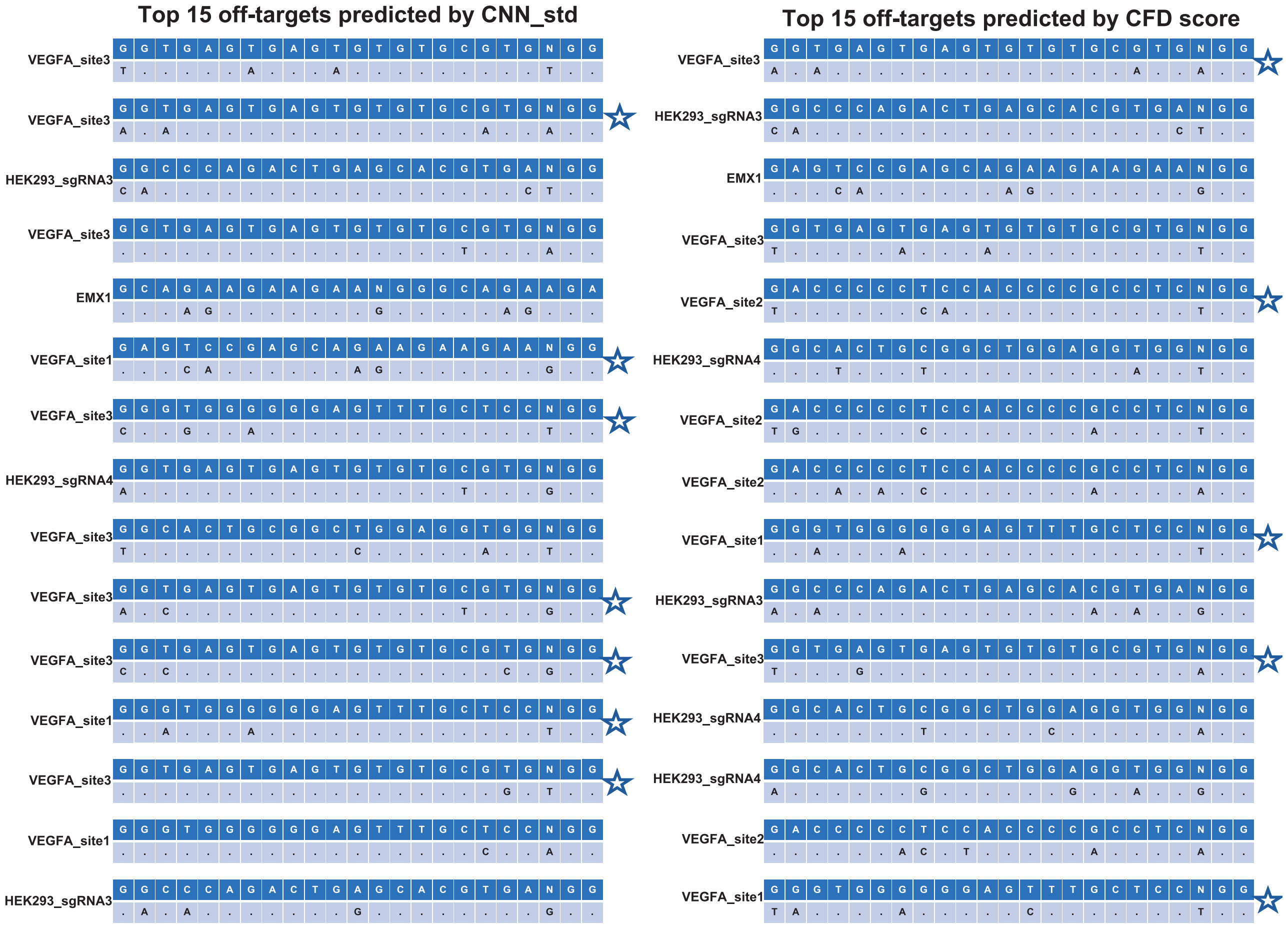


Fig. 6. 15 off-targets with the highest score predicted by final CNN and CFD score, respectively, on GUIDE-seq dataset. The sgRNA-DNA sequence marked with star is the true off-target

because most of the existing scoring methods generated various hand-crafted features for prediction, including the identity of mismatch, the position of the mismatch, the penalty of the nearby mismatches and different segmentations of sgRNA-DNA sequence. All these features can be captured from the original sgRNA-DNA sequence by convolutional scanning with max pooling windows in CNN. Moreover, the information of matched base sites can also be captured by convolutional neurons as influence factors for the off-target prediction. Therefore, comparing with the existing prediction approaches, our deep CNN have its own competitive edges.

In addition, we compared the generalization performance between our final CNN and the best state-of-art off-target prediction method, CFD score, on the GUIDE-seq dataset. Table 3 shows that the performance of CNN_std is still significantly better than the CFD score. For further analysis, we looked into the prediction results of the CFD score and CNN_std and selected the top 15 sgRNA-DNA sequence pairs with the highest prediction scores for each prediction model in Figure 6. We observe that there are 7 true off-targets among the top 15 off-targets predicted by CNN_std while there are only 5 true off-targets among the top 15 predicted by CFD score. Furthermore, the results shows that CFD have bad performance on predicting single-bp mismatches off-targets because there is not any true off-target with single-bp mismatch among the top 15 predicted off-targets of CFD score. Similar results could be observed for other top *k*-values.

5 Conclusion

We presented that deep neural networks are able to accurately predict the off-targets of CRISPR-Cas9 gene editing. To our knowledge, this is the first time that deep neural networks are designed and implemented for off-target predictions. Our final CNN, CNN_std, obtained the best performance on both CRISPOR dataset and GUIDE-seq dataset, outperforming the current state-of-art off-target prediction methods and three traditional machine learning algorithms including LR, RF and GBT. We discussed and attributed its performance successes to the neural network layer designs which are general enough to self-learn and capture sequence features. We believe that such intelligent approaches can contribute to CRISPR-Cas9 off-target predictions or other similar problems in a rigorous manner.

Acknowledgements

The work described in this article was substantially supported by three grants from the Research Grants Council of the Hong Kong Special Administrative Region [CityU 21200816], [CityU 11203217] and [CityU 11200218].

Conflict of Interest: none declared.

References

Abadi,M. *et al.* (2016). Tensorflow: large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv: 1603.04467.

Al-Attar,S. *et al.* (2011) Clustered regularly interspaced short palindromic repeats (CRISPRs): the hallmark of an ingenious antiviral defense mechanism in prokaryotes. *Biol. Chem.*, **392**, 277–289.

Almagro Armenteros,J.J. *et al.* (2017) Deeploc: prediction of protein subcellular localization using deep learning. *Bioinformatics*, **33**, 3387–3395.

Chen,F. *et al.* (2017) Targeted activation of diverse CRISPR-Cas systems for mammalian genome editing via proximal CRISPR targeting. *Nat. Commun.*, **8**, 14958.

Chen,J.S. *et al.* (2017) Enhanced proofreading governs CRISPR–Cas9 targeting accuracy. *Nature*, **550**, 407.

Cho,S.W. *et al.* (2014) Analysis of off-target effects of CRISPR/CAS-derived RNA-guided endonucleases and nickases. *Genome Res.*, **24**, 132–141.

Cong,L. *et al.* (2013) Multiplex genome engineering using CRISPR/Cas systems. *Science*, **339**, 819–823.

Corrigan-Curay,J. *et al.* (2015) Genome editing technologies: defining a path to clinic. *Mol. Ther.*, **23**, 796–806.

Cox,D.B.T. *et al.* (2015) Therapeutic genome editing: prospects and challenges. *Nat. Med.*, **21**, 121–131.

Doench,J.G. *et al.* (2016) Optimized sgrna design to maximize activity and minimize off-target effects of CRISPR-Cas9. *Nat. Biotechnol.*, **34**, 184–191.

Doudna,J.A. and Charpentier,E. (2014) The new frontier of genome engineering with CRISPR-Cas9. *Science*, **346**, 1258096.

Esvelt,K.M. *et al.* (2013) Orthogonal Cas9 proteins for RNA-guided gene regulation and editing. *Nat. Methods*, **10**, 1116.

Fu,Y. *et al.* (2013) High-frequency off-target mutagenesis induced by crispr-cas nucleases in human cells. *Nat. Biotechnol.*, **31**, 822–826.

Glorot,X. *et al.* (2011) Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, Ft. Lauderdale, FL, USA, pp. 315–323.

Haeussler,M. *et al.* (2016) Evaluation of off-target and on-target scoring algorithms and integration into the guide RNA selection tool CRISPOR. *Genome Biol.*, **17**, 148.

Hou,J. *et al.* (2017) Deepsf: deep convolutional neural network for mapping protein sequences to folds. *Bioinformatics*, **34**, 1295–1303.

Hsu,P.D. *et al.* (2013) Dna targeting specificity of RNA-guided Cas9 nucleases. *Nat. Biotechnol.*, **31**, 827–832.

Hsu,P.D. *et al.* (2014) Development and applications of CRISPR-Cas9 for genome engineering. *Cell*, **157**, 1262–1278.

Jurtz,V.I. *et al.* (2017) An introduction to deep learning on biological sequence data–examples and solutions. *Bioinformatics*, **33**: 3685–3690.

Kang,X.J. *et al.* (2017) Addressing challenges in the clinical applications associated with CRISPR/Cas9 technology and ethical questions to prevent its misuse. *Protein Cell*, **8**, 791–795.

Kim,D. *et al.* (2015) Digenome-seq: genome-wide profiling of crispr-cas9 off-target effects in human cells. *Nat. Methods*, **12**, 237–243.

Kingma,D.P. and Ba,J. (2014) Adam: A method for stochastic optimization. arXiv preprint arXiv: 1412.6980.

Klann,T.S. *et al.* (2017) CRISPR–Cas9 epigenome editing enables high-throughput screening for functional regulatory elements in the human genome. *Nat. Biotechnol.*, **35**, 561.

Liang,P. *et al.* (2015) CRISPR/Cas9-mediated gene editing in human tripronuclear zygotes. *Protein Cell*, **6**, 363–372.

Lih-Yuan,D. (2006). The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation, and machine learning. *Technometrics*, **48**, 147–148.

Ma,H. *et al.* (2017) Correction of a pathogenic gene mutation in human embryos. *Nature*, **548**, 413–419.

Mali,P. *et al.* (2013a) Cas9 transcriptional activators for target specificity screening and paired nickases for cooperative genome engineering. *Nat. Biotechnol.*, **31**, 833–838.

Mali,P. *et al.* (2013b) RNA-guided human genome engineering via Cas9. *Science*, **339**, 823–826.

Manguso,R.T. *et al.* (2017) In vivo CRISPR screening identifies Ptpn2 as a cancer immunotherapy target. *Nature*, **547**, 413.

Pattanayak,V. *et al.* (2013) High-throughput profiling of off-target DNA cleavage reveals RNA-programmed Cas9 nuclease specificity. *Nat. Biotechnol.*, **31**, 839–843.

Pedregosa,F. *et al.* (2011) Scikit-learn: machine learning in python. *J. Mach. Learn. Res.*, **12**, 2825–2830.

Ran,F.A. *et al.* (2013) Double nicking by RNA-guided CRISPR Cas9 for enhanced genome editing specificity. *Cell*, **154**, 1380–1389.

Sergey,I. and Christian,S. (2015). Batch normalization: accelerating deep network training by reducing internal covariate shift. In: Francis,B. and David,B. (eds.) *Proceedings of the 32nd International Conference on International Conference on Machine Learning*. Vol. 37, pp. 448–456.

Shalem,O. *et al.* (2014) Genome-scale crispr-cas9 knockout screening in human cells. *Science*, **343**, 84–87.

Shapiro,R.S. *et al.* (2018) A CRISPR–Cas9-based gene drive platform for genetic interaction analysis in *Candida albicans*. *Nat. Microbiol.*, **3**, 73.

Shen,J.P. *et al.* (2017) Combinatorial CRISPR–Cas9 screens for de novo mapping of genetic interactions. *Nat. Methods*, **14**, 573.

Shibata,M. *et al.* (2017) Real-space and real-time dynamics of CRISPR-Cas9 visualized by high-speed atomic force microscopy. *Nat. Commun.*, **8**, 1430.

Singh,R. *et al.* (2015) Cas9-chromatin binding information enables more accurate CRISPR off-target prediction. *Nucleic Acids Res.*, **43**, e118–e118.

Smith,C. *et al.* (2015) Efficient and allele-specific genome editing of disease loci in human ipscs. *Mol. Ther.*, **23**, 570–577.

Srivastava,N. *et al.* (2014) Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, **15**, 1929–1958.

Stemmer,M. *et al.* (2015) Cctop: an intuitive, flexible and reliable CRISPR/Cas9 target prediction tool. *PLoS One*, **10**, e0124633.

Tsai,S.Q. *et al.* (2015) Guide-seq enables genome-wide profiling of off-target cleavage by crispr-cas nucleases. *Nat. Biotechnol.*, **33**, 187–197.

Wu,Y. *et al.* (2013) Correction of a genetic disease in mouse via use of CRISPR-Cas9. *Cell Stem Cell*, **13**, 659–662.

Xu,X. *et al.* (2017) CRISPR-Cas9 cleavage efficiency correlates strongly with target-sgrna folding stability: from physical mechanism to off-target assessment. *Sci. Rep.*, **7**, 143.

Zeng,H. *et al.* (2016) Convolutional neural network architectures for predicting dna–protein binding. *Bioinformatics*, **32**, i121–i127.

Zhang,X.-H. *et al.* (2015) Off-target effects in CRISPR/Cas9-mediated genome engineering. *Mol. Ther. Nucleic Acids*, **4**, e264.

Zhu,L.J. (2015) Overview of guide RNA design tools for CRISPR-Cas9 genome editing technology. *Front. Biol.*, **10**, 289–296.