Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Deep learning improves the ability of sgRNA off-target propensity prediction

Qiaoyue Liu
University of Science and Technology Beijing
Xiang Cheng
University of Science and Technology Beijing
Gan Liu
University of Science and Technology Beijing
Bohao Li
University of Science and Technology Beijing
Xiuqin Liu (✉ mathlxq@163.com)
https://orcid.org/0000-0002-9855-8779

Methodology article

Keywords: sgRNA, off-target, deep learning, GloVe model

Posted Date: December 11th, 2019

DOI: https://doi.org/10.21203/rs.2.18444/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Version of Record: A version of this preprint was published on February 10th, 2020. See the published version at https://doi.org/10.1186/s12859-020-3395-z.

Deep learning improves the ability of sgRNA off-target propensity prediction

Qiaoyue Liu[1], Xiang Cheng[1], Gan Liu[1], Bohao Li[1], Xiuqin Liu[1*]

[*]Correspondence: mathlxq@163.com

[1] Department of information and computing science, University of Science and Technology Beijing, Beijing, 100083, China

Background

CRISPR/Cas9 system, as the third-generation genome editing technology, has been widely applied in target gene repair and gene expression regulation. Selection of appropriate sgRNA can improve the on-target knockout efficacy of CRISPR/Cas9 system with high sensitivity and specificity. However, when CRISPR/Cas9 system is operating, unexpected cleavage may occur at some sites, known as off-target. Presently, a number of prediction methods have been developed to predict the off-target propensity of sgRNA at specific DNA fragment. Most of them use artificial feature extraction operations and machine learning techniques to obtain off-target scores. With the rapid expansion of off-target data and the rapid development of deep learning theory, the existing prediction methods can no longer satisfy the prediction accuracy at the clinical level.

Results

Here, we propose a prediction method named CnnCrispr to predict the off-target propensity of sgRNA at specific DNA fragments. CnnCrispr automatically trains the sequence features of sgRNA-DNA pairs with GloVe model, and embeds the trained word vector matrix into the deep learning model including biLSTM and CNN with five hidden layers. We conducted performance verification on the data set provided by DeepCrispr, and found that the auROC and auPRC in the "leave-one-sgRNA-out" cross validation could reach 0.957 and 0.429

respectively (the pearson value and spearman value could reach 0.495 and 0.151 respectively under the same settings).

Conclusion

Our results show that CnnCrispr has better classification and regression performance than the existing states-of-art models.

Keywords: sgRNA, off-target, deep learning, GloVe model

1 Background

CRISPR/Cas9 system[1-4](Clustered regularly interspaced short palindromic repeats /CRISPR-associated 9 system), originally derived from the immune defense mechanism of archaea, it is one of the most popular gene editing technology in recent days. Compared with zinc-finger nucleases [5, 6](ZFNs) and transcription activator-like effector nuclease [6-8](TALENs) technologies, CRISPR/Cas9 system has a pellucid mechanism, simple operation and high efficiency, thus gradually replacing the earlier methods and presently being applied to the fields of biology and clinical medicine, *etc*.

CRISPR/Cas9 system requires three important components in the process of gene editing: Cas9 protein, guide RNA and PAM motif (protospacer adjacent motif)[9]. Among them, the guide RNA that recognizes a target DNA sequence through complementary base pairing is generally referred to as an sgRNA[10, 11] (single guide RNA, generally an RNA sequence of 20nt in length). The PAM[11-13] is a 3nt motif on the target sequence and a prerequisite for Cas9 protein cleavage at a specified site. A common type of PAM is NGG[14-16] (N represents any base of A, T, C, G). During the editing process, the Cas9 protein cleaves the target DNA at the

site three bases upstream of the PAM under the guidance of the sgRNA sequence, and performs subsequent gene editing operations : Introduction of an insertion/deletion (indel) base to cause mutation of a gene at a target position by nonhomologous end-joining (NHEJ); or utilization of the “donor template” provided by foreign DNA to recombine with a mutant target to achieve DNA-based editing of the genome by homology-directed repair (HDR)[17-19].

Some studies have found that when CRISPR/Cas9 system operates, several mismatch sites may appear in the complementary matching of sgRNA to the target DNA sequence, therefore resulting in unintended cleavage of the DNA sequence, which is called “off-target”[16, 20, 21]. Fu *et al*. [20]have confirmed that sgRNA allows 1-5 base mismatches during the guiding process, which in turn causes unintended sequences to be erroneously edited. The existence of off-target phenomenon has greatly hindered the clinical application and further promotion of CRISPR technology. How to assess the off-target propensity of specific sgRNAs and minimize the risk of off-target has become the focus of the CRISPR/Cas9 system study.

Presently, a variety of off-target detection methods have been developed, such as the GUIDE-Seq[22-24] method created by Tsai *et al*., which can effectively identify 0.1% of mutations in cells and predict the cleavage activity of the system based on sequencing results. The HTGTS[25] method utilizes fusion of known DNA double-strand breaks with other cleavage DNAs to detect DNA breaks by PCR amplification techniques and further detect off-target sites. On this basis, Frock *et al*.[26] further developed a higher throughput off-target detection method. The BLESS[27] techniques further speculate on off-target sites by detecting DNA double-strand breaks. However, this method is complicated to operate and it is impossible to detect a break site that has not occurred or has already been repaired. In addition, the IDLV [28, 29]

method can detect off-target sites within the genome-wide range without bias, but with an accuracy of only 1%.

The above detection methods cannot detect all off-target sites of specific sgRNA due to technical limitations. As the core of artificial intelligence, machine learning and deep learning can effectively analyze empirical data and provide important technical support for bioinformatics. To this date, machine learning has been gradually applied to off-target site prediction[14, 30], sgRNA activity prediction[14] and sgRNA design optimization[31, 32], *etc*. Various machine learning based sgRNA design models[30, 33-36] have been developed and put into application. Their main design idea is to introduce sgRNA sequence features and secondary structure features, rank all possible sgRNA for specific target DNA sequences by scores of off-target effect, and selecting the sgRNA with high cleavage efficiency and low off-target propensity.

The above machine learning methods were based on sequence features. At the time this paper is written, only three existing prediction models have introduced the idea of deep learning into the sgRNA off-target propensity prediction problem.

DeepCpf1[31], based on the convolutional neural network(CNN), introduced sgRNA sequence features and chromatin accessibility to predict the editing efficiency of sgRNA corresponding to Cpf1. This method does not have to construct the feature artificially, further simplifying the model, and is convenient for researchers to use. DeepCrispr[37] introduced four epigenetic features in addition to DNA sequence features and automatically extracts valid information using the principle of Auto-encoder. Several models including sgRNA target cleavage and off-target propensity prediction were established. However, it is still unknown whether the four

epigenetic characteristics will have a positive impact on the model prediction results. CNN_std[38] only used sequence features to construct two-dimensional input matrix by means of "XOR" coding design and utilized CNN for prediction. This deep learning method also received a higher accuracy in the CRISPOR dataset.

Most of the existing prediction methods are still based on machine learning methods and model prediction through complex manual feature extraction [39-44]. However, the internal mechanism of CRISPR gene editing technology is not presently clear and explicit. Manual design of sgRNA features may have a negative impact on the prediction results. Therefore, we would like to present CnnCrispr，a novel computational method for predication of sgRNA off-target cleavage propensity utilizing the deep learning method. In CnnCrispr, the GloVe embedding model was introduced to extract global and statistical information of input sequences by constructing the co-occurrence matrix of sgRNA and its corresponding DNA sequence. Further integrating with the deep neural network model, the off-target propensity of a given sgRNA at a specific DNA fragments can be predicted. We trained CnnCrispr with the data set used by DeepCrispr[37], and proved that CnnCrispr has a good competitive advantage in predicting sgRNA off-target propensity through performance comparison with four state-of-the-arts models, which is expected to be a potential tool for research on CRISPR technology.

2 Results

2.1 Model Structure and Prediction

In our initial conception, we combined biLSTM with CNN framework at the final prediction model and the model structure is shown in **Fig. 1**. We also constructed several similar but different models by removing different network parts to compare

the test results and choose the final prediction model. All pre-selected network frameworks for model selection are briefly described in **Table 1**.

The structure of the benchmark framework of CnnCrispr is described in detail below:

The first layer of CnnCrispr is an embedding layer, which is used for input of the vector obtained by GloVe model. Since the vector dimension of the GloVe model is set to 100, the input of embedding layer is a two dimensional matrix with the size of 16×100. We called the mittens package in Python to train the GloVe model on the basis of the realization of GloVe co-occurrence matrix.

The second layer is a biLSTM network, which is mainly used to extract the context features of input information. Five convolution layers are subsequently connected to the model, and each layer has a different kernel number and kernel size. Then the full connection layers are introduced behind the last convolution layer, has the sizes of 20 and 2 respectively.

In addition to the framework mentioned above, Batch Normalization and Dropout layers are added between each layers to prevent model overfitting. The parameters of the Dropout layer are set as 0.3. For the output layer, *softmax* and *sigmoid* function are used as activation functions respectively to obtain the prediction results of classification model and regression model.

In the training process, the initial learning rate was set as 0.01, and we used *Adam* algorithm to optimize the loss function. In addition, we set the batch size as 256 in consideration of the requirements of potential information extraction from negative data and avoiding the occurrence of over fitting. Too large of a batch size may

increase the risk of multiple occurrence of some positive data in a single batch during training, while too small of a batch size may reduce the training speed of a model and extend the training time.

Our experiment was divided into two parts. First, we compared the performance of different models. Then, the final prediction model was compared with the existing models with better performance to evaluate the practical application ability of our model. Detailed network descriptions are in Additional file 2.

2.2 Model Selection

Experimental data are from the attachment provided by DeepCrispr article, and the relevant data description is detailed in section 5.1. During the process of training, 20% of the data in the Hek293t and K562 data sets were selected randomly to compile the test sets (Hek293t test set, K562 test set and Total test set respectively). Different prediction models were obtained by training with all the remaining data, and the prediction performance of each model in the three test sets were evaluated. During the training process, we generated the batch training data using the data sampling method mentioned in **Section 5.6**.

We built two models for classification and regression prediction, respectively. The first three models mentioned in **Table 1** were trained in order to verify the influence of different parts on the prediction performance of the model. The structure of the benchmark model CnnCrispr is introduced in **Section 2.1**. And the model CnnCrispr_No_LSTM was obtained by removing the LSTM part from the basis of CnnCrispr, CnnCrispr_Conv_LSTM was obtained by adjusting the order of

Convolution layers and Recurrent layer on the basis of CnnCrispr. Among them, the purpose of the latter two models was mainly to illustrate whether CNN layer and RNN layer have improved the performance, as well as whether the order of the two frameworks will affect the performance.

We initially trained the three models mentioned above and obtained the prediction results. The model performance is shown in **Table 2**.

Due to the highly unbalanced nature of the data set, it was easy for the model to obtain a high auROC value. Therefore, we gave up the comparison of auROC values and focused on the comparison results of auPRC and Recall value on the test set. The results in **Table 2** were used to draw the histogram (**Fig. 2**), from which it can intuitively be seen that CnnCrispr has better predictive performance. Therefore, we took the CnnCrispr as the benchmark network framework and further well-tuned the network structure.

Based on CnnCrispr, the Dropout layer and Batch Normalization layer were removed respectively to verify the influence of the two parts on performance. A brief description of the network structure is given in **Table 1**. The recall value of CnnCrispr_No_Dropout was 0.810 in the total test set, which was a little lower than that of CnnCrispr, this showed that the Dropout layer does have improved performance and prevented over-fitting, although the degree of improvement is not very noticeable. However, after adding the Dropout layer, the training parameters of the model were greatly reduced, which further saved time for model training, hence we kept the Dropout layer in the final model. Then we trained the model without

the Batch Normalization layer several times and test it on the test set, but every time the entire test set were all classified as negative samples. This indicated that the model without the BN layer has lost its ability of classification prediction. Therefore, the BN layer is essential in the final model. In addition, we also mentioned the importance of BN layer for neural network model in **Section 5.5**, hence we reserved it in our final model.

2.3 Model Comparing

We selected four sgRNA off-target propensity prediction models for model comparison, namely CFD[33], MIT[16], CNN_std[38] and DeepCrispr[37].

CFD is short for Cutting Frequency Determination. As a scoring model for evaluating the off-target propensity of sgRNA-DNA interaction, CFD specified different scores for the location and type of mismatch between sgRNA and corresponding DNA sequence. When multiple mismatches appear in the sequence pair, the corresponding scores are multiplied to obtain the final score. For example, if the sgRNA-DNA sequence has a rG-dA mismatch in position 6 and a rC-dT mismatch in position 10, it will receive a CFD score of 0.67×0.87=0.583. *Haeussler et al.*[45] compared the performance of CFD with that of MIT, and proved that the prediction performance of CFD was slightly better than that of MIT in CRISPOR data set. CNN_std is a CNN-based sgRNA off-target propensity prediction model developed by *Jiecong Lin*. The combination of sgRNA and corresponding DNA sequences was encoded by "XOR" principle and predicted by multi-layer convolution network. DeepCrispr is a deep learning method which combines

sgRNA-DNA sequence information with genomic epigenetic characteristics as the input. DeepCrispr used the largest data set available to conduct model training and introduced the auto-encoder to automatically acquire potential features of the sgRNA-DNA sequence, which was a good attempt at deep learning in sgRNA related prediction problems.

In order to make a more comprehensive comparison with the four models above, we tested the performance of the classification and regression models in two test patterns. We downloaded the prediction models of CFD, MIT and CNN_std from relevant websites and obtained the prediction results on the same test set as Cnncrispr. Due to the fact that the training methods were consistent between CnnCrispr and DeepCrispr, we just used the test results given by DeepCrispr to make the comparison.

2.3.1 Test pattern 1 -- withheld 20% as an independent testing set

Consistent with the training method of “Model Selection” section, we randomly divided the data sets of each cell line in the proportion of 8:2. We compared the performance of CnnCrispr with the current preferable prediction models. **Fig. 3** shows the comparison results under the classification schema. CnnCrispr achieved an auROC value of 0.975 and an auPRC value of 0.679 at the total test set. Which were both higher than the value of CFD, MIT and CNN_std (there were similar trends in the Hek293t test set and K562 test set, CnnCrispr achieved the auROC of 0.971 and 0.995 on Hek293t test set and K562 test set, respectively. And auPRC of 0.686 and 0.688 on Hek293t test set

and K562 test set, respectively). The AUC values of ROC curve and PRC curve of CnnCrispr on the three test sets were all higher than those of CFD, MIT and CNN_std, which proved that CnnCrispr had more advanced prediction ability. In addition, the PRC curve obtained by CnnCrispr on the total test set and Hek293t test set completely contained the PRC curve obtained by the other three models, CFD, MIT and CNN_std, while on the K562 test set, only a small portion of the curve was covered by the CNN_std. Comprehensive comparison showed that the overall performance of the CnnCrispr was better than the other three models, and since the training and test sets were extremely unbalanced, the PRC curve and the area under it were more important measures for model evaluation, where CnnCrispr had a strong competitive advantage. In addition to the comparison with the above three models, we further compared the testing performance of CnnCprisr with DeepCrispr. Since the training methods and data sets were consistent, we directly compared the test results given in ref.[37], and the results are shown in **Table3**. The auROC values of DeepCrispr were slightly better than those of CnnCripr (shown more intuitively on Hek293t test set), but the auPRC values obtained by CnnCrispr on all three test sets were higher than those of DeepCrispr. By comprehensive comparison, the CnnCrispr had better performance than DeepCrispr under test pattern 1.

Unlike the classification schema, the Pearson correlation coefficient and Spearman rank correlation coefficient of the prediction results were mainly used as evaluation measures for regression schema. From the comparison

results, the Pearson correlation coefficient between CnnCrispr's prediction results and the real labels was strictly superior to the three comparison models(Since the Pearson coefficient was not selected as the evaluation measure in DeepCrispr, we only compared the Spearman values of CnnCrispr with DeepCrispr.).

The Pearson value of CnnCrispr on Hek293t test set reached 0.712(higher than 0.371 obtained by CFD, 0.153 obtained by MIT，0.33 obtained by CNN_std). In the entire test set, CnnCrispr also demonstrated its better predictive ability, with Pearson value reaching 0.682, higher than 0.343 of CFD, 0.150 of MIT and 0.321 of CNN_std. For Spearman correlation coefficient, the negative data in the test set was much larger than the positive data (about 250:1), therefore, a high Spearman value cannot be achieved. Nevertheless, the prediction ability of CnnCrispr was still better than those of the four models above (the test results of CnnCrispr on Hek293t, K562 and Total test set were 0.154, 0.160 and 0.134 respectively, while the Spearman correlation coefficients of CFD on the three test sets were 0.140, 0.143 and 0.128 respectively; Spearman correlation coefficients of MIT were 0.085, 0.084 and 0.086 respectively; Spearman correlation coefficients of CNN_std were 0.141, 0.144 and 0.132 respectively; Spearman correlation coefficients of DeepCrispr were 0.136, 0.126 and 0.133 respectively). In addition, we also compared the AUC values under ROC and PRC curves of the five models by referencing the CRISTA's evaluation method and considering the predicted results as the probability of the classification

labels. The auROC value and auPRC value obtained by CnnCrispr on the total test set were as high as 0.986 and 0.601 respectively, which were superior to 0.942 and 0.316 of CFD, 0.947 and 0.208 of CNN_std, and the same results were obtained on Hek293t and K562 test sets. Based on the above performance results, we concluded that CnnCrispr had better predictive performance.

2.3.2 Test pattern 2 – “Leave-one-sgRNA-out”

In order to examine the accuracy and generalization ability of CnnCrispr for the prediction of off-target propensity of new sgRNA, we set up the "leave-one-sgRNA-out" experiment, which is a good evaluation method for the prediction of off-target propensity. During the training, a sgRNAs and its corresponding off-target sequences(with true cleaved propensity or the potential sites obtained from whole genome) were completely extracted for model testing. According to the difference of sgRNAs, a total of 29 times of model training and performance evaluations were conducted. Through this 29-fold cross-validation method, we were able to comprehensively evaluate the generalization ability of CnnCrispr and avoid over-fitting or under-fitting of the model when predicting for some special sgRNAs.

For classification, CnnCrispr achieved an average auROC of 0.957, auPRC of 0.429, which were both higher than the results of the four models above (CFD achieved an average auROC of 0.903, auPRC of 0.319, MIT achieved an average auROC of 0.848, auPRC of 0.115, CNN_std achieved an average auROC of 0.925, auPRC of 0.303; and DeepCrispr achieved an average auROC

of 0.841, auPRC of 0.421). In the 29-fold cross validation, CnnCrispr's comprehensive competitive advantage was more significant, and the auPRC results were higher than results yielded by the other four models, which was of great significance to prevent the model from missing the actual off-target sites (see **Fig. 5**).

In order to make a more comprehensive evaluation, we also considered the distribution of the values of auROC and auPRC obtained by "29-fold" cross-validation, and drew the violin plot (Because we didn't get the test data of DeepCrispr, we could not draw the violin pot of it.). Violin plot is characterized by the kernel density estimation of the basic distribution, and the external shape of the violin plot is the kernel density estimation. First of all, **Fig. 6** shows that the auROC values of CnnCrispr were generally higher and the AUC values of CnnCrispr were more concentrated, 75% of the prediction results were greater than 0.9. On the other hand, there were obvious abnormal points in the prediction results of auROC by the other three models, indicating that they cannot play a good role in predicting the off-target propensity of individual sgRNA. In addition, the distribution of CnnCrispr's auROC values was more concentrated, while the auROC values of CFD and CNN_std had obvious discrete values (the whiskers on the lower side were longer).With the increase of auROC values, the horizontal distance of the violin plot plotted by CnnCrispr was larger, which showed that more auROC values were distributed on this interval, further indicating the good prediction performance of

CnnCrispr. For auPRC values, the median of prediction results obtained by CnnCrispr was significantly larger than the other three models, which showed that CnnCrispr had a higher overall score and 75% of auPRC values obtained by CnnCrispr were greater than 0.2. CnnCrispr was more distributed at higher scores, indicating that the overall predictive performance of CnnCrispr was indeed better than that of CFD and CNN_std(see **Fig. 6**).

We further compared the 29-fold cross-validation results in regression schema and organized the performance visualization results in **Fig. 5-6**. We first compared the average value of Pearson correlation coefficient and the Spearman correlation coefficient (see **Fig. 5**). CnnCrispr achieved a higher mean Pearson value and Spearman value, this showed that CnnCrispr had better fitting ability. Furthermore, we drew 29 sets of Pearson values and Spearman values into violin maps. As shown in **Fig. 6**, Pearson values obtained by CnnCrispr were more distributed in the high score range. In addition, the Spearman scores of all four models were lower, but despite this, the distribution of CnnCrispr scores was significantly better than that of the other three models. CnnCrispr had a higher probability of obtaining highly fitting prediction results for off-target propensity.

3 Discussion

Initially, RNN was applied to the related problems of sequence data. The biggest difference between it and other networks is that RNN can achieve some "memory function", which is the best choice for time series data analysis. We regarded sgRNA-DNA pair sequences

as data with time characteristics, that is, the context of sequences contains potential features which can be used for result prediction.

The convolution kernel size of the CNN was smaller than the input matrix, so the convolution operation can extract more local features -- which is consistent with the image processing. In fact, it is not necessary for each neuron to perceive the global image, but only need to perceive the local, and then integrate the local information at a higher level to obtain the global information. The parameter sharing mode of CNN can also greatly reduce the computation. Secondly, we set convolution kernels of different sizes for different levels in the convolution part, and used multiple convolution kernels to convolve the input images, so as to extract local features as comprehensively as possible in this way. In addition, GloVe method was based on the statistical information of global word co-occurrence to learn word vectors, so as to combine the advantages of statistical information with the local context window method. We used this method to replace the traditional "one-hot" representation method and made the input sequence of CnnCrispr have better characteristic representation ability.

In the process of model building, RNN and CNN were considered to be combined, and the excellent prediction ability of the final network model CnnCrispr was proved by comparing with the performance of different pre-selected models. The final network structure is shown in **Fig. 1**. After the GloVe model, the biLSTM was connected to extract context features, and the two-dimensional matrix information was further extracted by using 5 convolutional layers. In the output layer of the network, the model was divided into classification schema and regression schema by setting different activation functions

(*softmax* or *sigmoid* functions).

In "Model Selection" section, we also intuitively saw that the order of RNN and CNN had a great impact on the test performance, and the model CnnCrispr_Conv_LSTM cannot play a very good role in feature extraction and data prediction (see **section 2.2** and **Table 2**).We briefly analyze the following reasons: the RNN can fully extract the contextual text features of input sequences, while the convolution operation will initially break the internal connection of sequences and affect the function of RNN. Firstly, the RNN operation was carried out to extract the context features of the sequence, and then the CNN was used to extract the local features, and the local information was integrated at a higher level to obtain the global feature information, so as to improve the prediction ability of CnnCrispr. In comparison with the performance of the existing four state-of-the-arts prediction models, CnnCrispr had better prediction ability in highly unbalanced test sets from DeepCrispr. In the "leave-one-sgRNA-out" experiment, the mean auPRC of 0.471 and mean Pearson value of 0.502 were achieved, which showed that CnnCrispr has a better competitive advantage. In addition, CnnCrispr only used the sequence information between sgRNA and corresponding potential DNA segments, giving up the construction of artificial features, thus avoiding the introduction of invalid or interfering information and making the prediction results more convincing.

We hope that CnnCrispr can help clinical researchers narrow down the screening range of off-target site test and save researchers' time and energy.

Since 2014, the number of open source data sets and online resources available for studying the application of machine learning on CRISPR/Cas9 system has been increasing.

As of the day the composition is written, the data set used by the author for model training is the largest data set presently available. However, with the continuous development of biological research technology, the available open source data sets will gradually increase, which will further improve the generalization ability of CnnCrispr in the future.

4 Conclusion

In this paper, we built a novel sgRNA off-target propensity prediction model, CnnCrispr. With introduction of the GloVe model, CnnCrispr attempted new feature representation methods to embed sequence information into the deep learning model, combined RNN with CNN, and only used sequence information to predict the off-target propensity of sgRNA at specific sites. By comparison with existing prediction models, the superior prediction ability of CnnCrispr was further confirmed. Our model used deep learning to comprehend the automatic learning of sequence features between sgRNA and corresponding potential off-target site, avoiding the unknown influence of artificial feature construction process on model prediction results, which is a new attempt at deep learning in the direction of sgRNA off-target propensity prediction.

5 Materials and Methods

5.1 Data sources

So far, there is no public website to integrate off-target data, and most studies still use various detection methods to obtain the data of potential off-target sites and off-target propensity of specific sgRNA at specific site. We used the off-target data that has been published in the DeepCrispr article as our training data. The off-target data set contains a total of 29 sgRNA from two different cell types: 293-related cell lines

and K562t. For all 29 sgRNAs, a total of more than 650 positive data have been identified as off-target sites, and Guohui C *et al*.[37] obtained more than 160,000 possible loci across the whole genome similar to the corresponding sgRNA using *bowtie2*. The whole dataset was highly unbalanced, it was likely to affect model fitting precision in the process of training, we will further describe the concrete solution in "Sampling for Training data" section. For the classification model, the labels of off-target sites were set to "1", and the labels of other sites were set to "0".For the regression model, the labels of the off-target site were set to the targeting cleavage frequency detected by different detection assays and other sites were set to "0".

5.2 Data Preprocessing and Encoding

The sgRNA sequence and its corresponding DNA sequence are each composed of 23 bases. Considering that the combination of the two bases at the same site is a feature, since the DNA sequence is composed of four types of bases: A, C, G and T, there are altogether 4×4=16 possible situations for this feature. Therefore, we set a unique index value for each combination, and encoded the original sgRNA-DNA sequence with an index vector of 23 for subsequent GloVe model embedding (showed in **Fig. 1a**).

5.3 GloVe model for Data Embeding

As an unsupervised word representation method, the GloVe[46] model enables vectors to contain as many hidden features of input data as possible through vectorization of the original data, which can eliminate the disadvantages of the one-

hot coding method. In this model, the input vector was obtained by multiplying the one-hot coding of the original "vocabulary" by a trained weight matrix. When training the GloVe model, the co-occurrence matrix X should be firstly calculated according to the original "corpus" (here it refers to the original data set composed of our original sgRNA and corresponding off-target sequences). An element $x_{i,j}$ in matrix X is the sum of the times that the word w_j appears in the context box of the word w_i.

$X_i = \sum_{j=1}^{N} x_{i,j}$, represents the sum of the times that appear in the context box of word w_i for all words in the word list.

$P_{i,k} = \frac{x_{i,k}}{X_i}$, represents the probability that word w_k appears in the context box of word w_i.

$ratio_{i,j,k} = \frac{P_{i,k}}{P_{j,k}}$, represents the correlation of words w_i, w_j, w_k (**Table 4**). We can notice that the value of $ratio_{i,j,k}$ is calculated according to the co-occurrence times in the co-occurrence matrix $X = (x_{i,j})$. Now we hope to construct a word vector for each word and reproduce the value by using the word vector v_i, v_j, v_k and a specific function $g(\bullet)$. If such a word vector v_i can be found, it indicates that the word vector will definitely contain the information in the co-occurrence matrix.

In order to make the value of $g(v_i, v_j, v_k)$ as close as possible to $ratio_{i,j,k} = \frac{P_{i,k}}{P_{j,k}}$, we considered to build a cost function:

$$J=\sum_{i,j,k}^{N}\left(\frac{P_{i,k}}{P_{j,k}}-g\left(v_i,v_j,v_k\right)\right)^2,$$

and obtained the final cost function expression:

$$J=\sum_{i,j}^{N}f\left(x_{i,j}\right)\left(v_i{}^{\mathrm{T}}v_j+b_i+b_j-\log\left(x_{i,j}\right)\right)^2$$

through a series of hypothesis derivation.

In order to implement the GloVe embedding model, we called the *Python* extension package *mittens*, and used the GloVe model to train the preprocessed co-occurrence matrix. Finally, the embedded word vector representation was obtained. Global Vector integrated the global statistics of Latent Semantic Analysis (LSA) with the advantages of local context window. By integrating the aforementioned statistical information in its entirety, the training speed of the model can be accelerated and the relative weight of words can be controlled.

5.4 Recurrent Neural Network and LSTM Variant

As a special neural network model, the recurrent neural network (RNN) adds the horizontal connection of each neuron node in the same layer on the basis of the multi-layer feedforward network.

$$O_t=g(VS_t)$$
$$S_t=f(Ux_t+WS_{t-1})$$

RNN is mainly used to process time series. The output layer performs a full connection operation with the adjacent hidden layer. V is the connection matrix of the output layer, and g is used as an activation function to obtain the final output result. For the hidden layer at time t, neuron node first receives the network

input from that moment by weight U and receives the hidden layer output from time $t-1$ by weight W. And further operations on the sum of the two are taken. It can be seen that RNN has only one hidden layer, and the hidden layer is called multiple times during the training process to realize the information extraction function of the input information context. RNN implements the memory function but this memory function is limited because there will be a phenomenon of gradient disappearance or gradient explosion. Therefore, *Hochreiter* and *Schmidhuber*[47] proposed the LSTM network to solve the above mentioned problems by introducing a memory cell, the key to which is the cell state, which receives or rejects input information by a well-designed structure called a "gate". Although many LSTM variants have been proposed, we have only described the forward recursion of a most basic LSTM model. The basic formulas were given below:

$$\text{Input gate: } f_t = \sigma\left(W_f \cdot \left[h_{t-1}, x_t\right] + b_f\right)$$

$$\text{Forgetting gate: } i_t = \sigma\left(W_i \cdot \left[h_{t-1}, x_t\right] + b_i\right)$$

$$\tilde{C}_t = \tanh\left(W_C \cdot \left[h_{t-1}, x_t\right] + b_C\right)$$

$$\text{Output gate: } o_t = \sigma\left(W_o \cdot \left[h_{t-1}, x_t\right] + b_o\right)$$

For the model input at time t, the input gate controls the selection of information and adds it to next step. The value of the input gate is usually between $[0,1]$ as the degree of information selection. The setting of forgetting gate is mainly to prevent the introduction of too much information to relieve the burden of memory C. Therefore, some useless information is discarded by setting forgetting gate. As a deformed structure of RNN, LSTM adds memory units to each neuron in the hidden

layer. Through the setting of several controllable gates on the neuron, it can control the memory and forgetting degree of current information and previous information, thus achieve the function of long-term memory.

In order to get the characteristic representation of forward and backward information of RNA sequences, the bidirectional LSTM(showed in **Fig. 1c**), a variant of LSTM, was used, which consists of two parallel LSTM: one input forward sequence and one input reverse sequence. Here's how it works:

$$y_2 = g\left(VA_2 + V'A_2'\right)$$

And,

$$A_2 = f\left(WA_1 + Ux_2\right)$$
$$A_2' = f\left(W'A_3' + U'x_2\right)$$

For output y_2, its input mainly comes from the weighted sum of two parts, A_2 and A_2'. Therefore, two values should be stored in the hidden layer of LSTM, which are respectively A participating in forward calculation and A' participating in reverse calculation.

5.5 Convolution Neural Network and Batch Normalization

CNN (showed in **Fig. 1d**) mainly uses image data as network input, which avoids the complex process of feature extraction and data reconstruction that often occurs in traditional recognition algorithms. Therefore, it has great advantages in 2D image processing.

In the training of CnnCrispr, due to the small number of samples available for training and testing, the samples in the training set will be over-trained, which will lead to overfitting problem and the insufficient generalization ability of the model

in the test and other related data sets. Therefore, we need to reduce the over-fitting phenomenon of the model as much as possible by adjusting the model structure. Batch Normalization is a good tool for solving this problem. With the deepening of the network depth or in the training process, the distribution of the activated input value before the nonlinear transformation will gradually shift or change of the deep neural network, and the gradient of the low layer neural network will disappear in the backward propagation process, so that the convergence speed of the deep layer neural network will decrease. Batch Normalization forcibly transforms the input value distribution of any neuron into the standard normal distribution with mean equal to 0 and variance equal to 1, ensuring that the input value of each layer of network is uniformly distributed in the training process of the model, so as to avoid the gradient disappearance problem and further accelerate the training speed. In addition, the Batch Normalization layer, as an alternative operation to Dropout, avoids the inactivation of some input nodes and thus reduces the loss of effective information.

5.6 Sampling for Training data

The samples in the entire data set is extremely unbalanced: the number of negative samples is about 250 times that of positive samples. A highly unbalanced data set may make the gradient update process unstable, increase the difficulty of training, and even lead to the failure of model training. In order to solve this problem, we designed a data sampling method for model training: we set the batch size as m for model training, and divided the negative sample set into N subsets according

to this value (if the negative set has M samples, batch size is m, then N =[M / m]), and N random oversampling operations are further performed on the positive training set, therefore generating N training batches. In the data sampling process, almost all negative data is traversed, and the potential information of the negative data can be searched more reliably. However, one thing to note is that oversampling may over-emphasize the effects of some positive data on the training results, resulting in the model overfitting. So the choice of batch size is very crucial.

5.7 Model Evaluation and Performance Measures

In this paper, we investigated the classification and regression models for prediction of off-target propensity. For the classification model, the confusion matrix M of the predicted results can be obtained.

$$M = \begin{bmatrix} TP & FN \\ FP & TN \end{bmatrix}$$

Among them, TP (True Positive) and TN (True Negative) were expected prediction results, while FP (False Positive) and FN (False Negative) would reduce the prediction accuracy of our model.

For the prediction of off-target propensity, we should pay more attention to whether the actual off-target site is correctly predicted as "off-target" -- that is, whether the label value is 1. Therefore, ROC curve, PRC curve and Recall value were selected as the performance measures. The Recall value is defined as follows:

$$Recall = \frac{TP}{TP + FN}$$

The recall value can well describe the proportion of the actual off-target sites that

are correctly classified, that is, the larger the recall value is, the better the prediction ability of the model will be.

ROC and PRC curves[48] are widely used in classification model, it is important to note that since the prediction problem of category has highly unbalanced characteristics, compared with the AUC value under the ROC curve, the area under the PRC and its curve is more worthy of attention, the higher the value, proves the model has better performance in class imbalance problems.

For the regression model, the Pearson and Spearman correlation coefficients are mainly considered as metrics. The two measures are continuous variable correlation evaluation indicators, among them, the Pearson value pay more attention to whether it has a linear relationship between two variables, while Spearman value is a nonparametric statistical method to do the linear correlation analysis by using the rank of each variable.

Additional file

Additional file 1. Detailed comparison results for sgRNA off-target propensity prediction. (XLSX 20 kb)

Additional file 2. A detailed description of the model structure for model selection. (PDF 184 kb)

Abbreviations

CRISPR/Cas9 system: Clustered regularly interspaced short palindromic repeats /CRISPR-associated 9 system; GloVe: Globel vector; CNN: Convolutional neural network; PAM:

Protospacer adjacent motif; sgRNA: Single-guide RNA; RNN: Recurrent neural network; auROC: Area under the ROC curve; auPRC: Area under the PRC curve.

Acknowledgments

The authors are grateful to the anonymous reviewers for their insightful comments and suggestions.

Funding

This work was supported by grants from the Fundamental Research Funds for the Central Universities (No.FRF-BR-18-008B). The funders had no role in the design of the study, the collection, analysis, and interpretation of data and in writing the manuscript.

Availability of data and materials

All data used during this study are included in the published article "DeepCrispr" and its supplementary information files (DOI：10.1186/s13059-018-1459-4). We obtained 29 sgRNAs and their corresponding DNA sequences from its **additional file 2**.

Author Contributions

X.L and Q.L conceived and designed the experiments. G.L and B.L made the investigation. Q.L and X.C performed the experiments, Q.L wrote the paper. X.L revised the manuscript. We ensured that all authors had read and approved the manuscript, and ensured that this is the case.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

References

1. Devaki B, Michelle D, Rodolphe B: **CRISPR-Cas systems in bacteria and archaea: versatile small RNAs for adaptive defense and regulation**. *Annual Review of Genetics* 2011, **45**(45):273-297.
2. Terns MP, Terns RM: **CRISPR-based adaptive immune systems**. *Current Opinion in Microbiology* 2011, **14**(3):321-327.
3. Blake W, Sternberg SH, Doudna JA: **RNA-guided genetic silencing systems in bacteria and archaea**. *Nature* 2012, **482**(7385):331-338.
4. Ishino Y, ., Shinagawa H, ., Makino K, ., Amemura M, ., Nakata A, . **Nucleotide sequence of the iap gene, responsible for alkaline phosphatase isozyme conversion in Escherichia coli, and identification of the gene product**. *Journal of Bacteriology* 1987, **169**(12):5429-5433.
5. Miller JC, Holmes MC, Wang J, Guschin DY, Lee YL, Rupniewski I, Beausejour CM, Waite AJ, Wang NS, Kim KA: **Miller, J.C. et al. An improved zinc-finger nuclease architecture for highly specific genome editing. Nat. Biotechnol. 25, 778-785**. *Nature Biotechnology* 2007, **25**(7):778-785.
6. Wood AJ, Te-Wen L, Bryan Z, Pickle CS, Ralston EJ, Lee AH, Rainier A, Miller JC, Elo L, Xiangdong M: **Targeted genome editing across species using ZFNs and TALENs**. *Science* 2011, **333**(6040):307-307.
7. Dirk H, Haoyi W, Samira K, Lai CS, Qing G, Cassady JP, Cost GJ, Lei Z, Yolanda S, Miller JC: **Genetic engineering of human pluripotent cells using TALE nucleases**. *Nature Biotechnology* 2011, **29**(8):731-734.
8. Michelle C, Tomas C, Doyle EL, Clarice S, Feng Z, Aaron H, Bogdanove AJ, Voytas DF: **Targeting DNA double-strand breaks with TAL effector nucleases**. *Genetics* 2010, **186**(2):757-761.
9. Makarova KS, Haft DH, Rodolphe B, Brouns SJJ, Emmanuelle C, Philippe H, Sylvain M, Mojica FJM, Wolf YI, Yakunin AF: **Evolution and classification of the CRISPR-Cas systems**. *Nature Reviews Microbiology* 2011, **9**(6):467-477.
10. Elitza D, Krzysztof C, Sharma CM, Karine G, Yanjie C, Pirzada ZA, Eckert MR, J?Rg V, Emmanuelle C: **CRISPR RNA maturation by trans-encoded small RNA and host factor RNase III**. *Nature* 2011, **471**(7340):602-607.
11. Martin J, Krzysztof C, Ines F, Michael H, Doudna JA, Emmanuelle C: **A programmable dual-RNA-guided DNA endonuclease in adaptive bacterial immunity**. *Science* 2012, **337**(6096):816-821.
12. Mojica FJM, Díez-Villase?Or C, ., García-Martínez J, ., Almendros C, . **Short motif sequences determine the targets of the prokaryotic CRISPR defence system**. *Microbiology* 2009, **155**(3):733-740.
13. Sternberg SH, Sy R, Martin J, Greene EC, Doudna JA: **DNA interrogation by the CRISPR RNA-guided endonuclease Cas9**. *Nature* 2014, **507**(7490):62-67.

14. Cem K, Sevki A, Ritambhara S, Jeremy T, Mazhar A: **Genome-wide analysis reveals characteristics of off-target sites bound by the Cas9 endonuclease**. *Nature Biotechnology* 2014, **32**(7):677-683.
15. Zhang Y, Ge X, Yang F, Zhang L, Zheng J, Tan X, Jin ZB, Qu J, Gu F: **Comparison of non-canonical PAMs for CRISPR/Cas9-mediated DNA cleavage in human cells**. *Scientific Reports* 2014, **4**:5405.
16. Hsu PD, Scott DA, Weinstein JA, F Ann R, Silvana K, Vineeta A, Yinqing L, Fine EJ, Xuebing W, Ophir S: **DNA targeting specificity of RNA-guided Cas9 nucleases**. *Nature Biotechnology* 2013, **31**(9):827-832.
17. Lu XJ, Xue HY, Ke ZP, Chen JL, Ji LJ: **CRISPR-Cas9: a new and promising player in gene therapy**. *Journal of Medical Genetics* 2015, **52**(5):289-296.
18. Rouet P, ., Smih F, ., Jasin M, . **Introduction of double-strand breaks into the genome of mouse cells by expression of a rare-cutting endonuclease**. *Molecular & Cellular Biology* 1994, **14**(12):8096-8106.
19. Rouet P, ., Smih F, ., Jasin M, . **Expression of a site-specific endonuclease stimulates homologous recombination in mammalian cells**. *Proc Natl Acad Sci U S A* 1994, **91**(13):6064-6068.
20. Yanfang F, Foden JA, Cyd K, Maeder ML, Deepak R, J Keith J, Sander JD: **High-frequency off-target mutagenesis induced by CRISPR-Cas nucleases in human cells**. *Nature Biotechnology* 2013, **31**(9):822-826.
21. Vikram P, Steven L, Guilinger JP, Enbo M, Doudna JA, Liu DR: **High-throughput profiling of off-target DNA cleavage reveals RNA-programmed Cas9 nuclease specificity**. *Nature Biotechnology* 2013, **31**(9):839-843.
22. Tsai SQ, Zongli Z, Nguyen NT, Matthew L, Topkar VV, Vishal T, Nicolas W, Cyd K, A John I, Long P, Le: **GUIDE-seq enables genome-wide profiling of off-target cleavage by CRISPR-Cas nucleases**. *Nature Biotechnology* 2015, **33**(2):187-197.
23. Kleinstiver BP, Prew MS, Tsai SQ, Nguyen NT, Topkar VV, Zheng Z, Joung JK: **Broadening the targeting range of Staphylococcus aureus CRISPR-Cas9 by modifying PAM recognition**. *Nature Biotechnology* 2015, **33**(12):1293-1298.
24. Kleinstiver BP, Prew MS, Tsai SQ, Topkar VV, Nguyen NT, Zheng Z, Gonzales APW, Li Z, Peterson RT, Yeh JRJ: **Engineered CRISPR-Cas9 nucleases with altered PAM specificities**. *Nature* 2015, **523**(7561):481-485.
25. Chiarle R, Zhang Y, Frock R, Lewis S, Molinie B, Ho YJ, Myers D, Choi V, Compagno M, Malkin D: **Genome-wide Translocation Sequencing Reveals Mechanisms of Chromosome Breaks and Rearrangements in B Cells**. *Cell* 2011, **147**(1):107-119.
26. Frock RL, Jiazhi H, Meyers RM, Yu-Jui H, Erina K, Alt FW: **Genome-wide detection of DNA double-stranded breaks induced by engineered nucleases**. *Nature Biotechnology* 2015, **33**(2):179-186.
27. Crosetto N, Mitra A, Silva MJ, Bienko M, Dojer N, Wang Q, Karaca E, Chiarle R, Skrzypczak M, Ginalski K: **Nucleotide-resolution DNA double-strand break mapping by next-generation sequencing**. *Nature Methods* 2013, **10**(4):361-365.
28. Xiaoling W, Yebo W, Xiwei W, Jinhui W, Yingjia W, Zhaojun Q, Tammy C, He H, Ren-Jang L, Jiing-Kuan Y: **Unbiased detection of off-target cleavage by CRISPR-Cas9 and TALENs using integrase-defective lentiviral vectors**. *Nature Biotechnology* 2015, **33**(2):175-178.

29. Osborn MJ, Webber BR, Knipping F, Lonetree CL, Tennis N, Defeo AP, Mcelroy AN, Starker CG, Lee C, Merkel S: **Evaluation of TCR Gene Editing Achieved by TALENs, CRISPR/Cas9, and megaTAL Nucleases**. *Molecular Therapy* 2016, **24**(3):570-581.
30. Listgarten J, Weinstein M, Kleinstiver BP, Sousa AA, Joung JK, Crawford J, Gao K, Hoang L, Elibol M, Doench JG: **Prediction of off-target activities for the end-to-end design of CRISPR guide RNAs**. *Nature Biomedical Engineering* 2018, **2**(1):38-47.
31. Hui KK, Min S, Song M, Jung S, Choi JW, Kim Y, Lee S, Yoon S, Kim H: **Deep learning improves prediction of CRISPR–Cpf1 guide RNA activity**. *Nature Biotechnology* 2018, **36**(3):239-241.
32. Yanni L, Cradick TJ, Brown MT, Harshavardhan D, Piyush R, Neha S, Wile BM, Vertino PM, Stewart FJ, Gang B: **CRISPR/Cas9 systems have off-target activity with insertions or deletions between target DNA and guide RNA sequences**. *Nucleic Acids Research* 2014, **42**(11):7473-7485.
33. Doench JG, Fusi N, Sullender M, Hegde M, Vaimberg EW, Donovan KF, Smith I, Tothova Z, Wilen C, Orchard R: **Optimized sgRNA design to maximize activity and minimize off-target effects of CRISPR-Cas9**. *Nature Biotechnology* 2016, **34**(2):184-191.
34. Pei FK, Powers S, He S, Li K, Zhao X, Bo H: **A systematic evaluation of nucleotide properties for CRISPR sgRNA design**. *Bmc Bioinformatics* 2017, **18**(1):297.
35. Abadi S, Yan WX, Amar D, Mayrose I: **A machine learning approach for predicting CRISPR-Cas9 cleavage efficiencies and patterns underlying its mechanism of action**. *Plos Computational Biology* 2017, **13**(10):e1005807.
36. Rahman MK, Rahman MS: **CRISPRpred: A flexible and efficient tool for sgRNAs on-target activity prediction in CRISPR/Cas9 systems**. *Plos One* 2017, **12**(8):e0181943.
37. Chuai G, Ma H, Yan J, Chen M, Hong N, Xue D, Zhou C, Zhu C, Chen K, Duan B: **DeepCRISPR : optimized CRISPR guide RNA design by deep learning**. *Genome Biology* 2018, **19**(1):80.
38. Jiecong L, Ka-Chun W: **Off-target predictions in CRISPR-Cas9 gene editing using deep learning**. *Bioinformatics* 2018, **34**(17):i656-i663.
39. Henriette OG, Henry IM, Bhakta MS, Meckler JF, Segal DJ: **A genome-wide analysis of Cas9 binding specificity using ChIP-seq and targeted sequence capture**. *Nucleic Acids Research* 2015, **43**(6):3389-3404.
40. Xuebing W, Scott DA, Kriz AJ, Chiu AC, Hsu PD, Dadon DB, Cheng AW, Trevino AE, Silvana K, Sidi C: **Genome-wide binding of the CRISPR endonuclease Cas9 in mammalian cells**. *Nature Biotechnology* 2014, **32**(7):670-676.
41. Doench JG, Ella H, Graham DB, Zuzana T, Mudra H, Ian S, Meagan S, Ebert BL, Xavier RJ, Root DE: **Rational design of highly active sgRNAs for CRISPR-Cas9-mediated gene inactivation**. *Nature Biotechnology* 2014, **32**(12):1262-1267.
42. Tim W, Wei JJ, Sabatini DM, Lander ES: **Genetic screens in human cells using the CRISPR-Cas9 system**. *Science* 2014, **343**(6166):80-84.
43. Nathan W, Weijun L, Xiaowei W: **WU-CRISPR: characteristics of functional guide RNAs for the CRISPR/Cas9 system**. *Genome Biology* 2015, **16**(1):218.
44. Alkhnbashi OS, Fabrizio C, Shah SA, Garrett RA, Saunders SJ, Rolf B: **CRISPRstrand: predicting repeat orientations to determine the crRNA-encoding strand at CRISPR loci**. *Bioinformatics* 2014, **30**(17):489-496.

45. Haeussler M, Kai S, Eckert H, Eschstruth A, Mianné J, Renaud JB, Schneider-Maunoury S, Shkumatava A, Teboul L, Kent J: **Evaluation of off-target and on-target scoring algorithms and integration into the guide RNA selection tool CRISPOR**. *Genome Biology* 2016, **17**(1):148.
46. Pennington J, Socher R, Manning CD: **GloVe: Global Vectors for Word Representation**. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing(EMNLP)*. 2014: 1532-1543.
47. Hochreiter S, Schmidhuber J: **Long Short-Term Memory**. *Neural Computation* 1997, **9**(8):1735-1780.
48. David F, Benjamin R: **Estimation of the area under the ROC curve**. *Statistics in Medicine* 2002, **21**(20):3093-3106.

Tables:

Table1. Brief network framework description of several pre-selected models.

Model_Name	Description
CnnCrispr	The final model including all units we mentioned
CnnCrispr_NoLSTM	Without biLSTM layer
CnnCrispr_Conv_LSTM	Reverse the order of the convolution layer and the recurrent layer
CnnCrispr_NoBatchNor	Without Batch Normalization layer
CnnCrispr_NoDropout	Without Dropout layer

CnnCrispr is the benchmark model. On this basis, some parts were removed to obtain the comparison models. Except parts mentioned in the description, the structure of the contrast model was completely consistent with the benchmark model. The CnnCrispr_Conv_LSTM model replaced the sequence of convolutional layers and recurrent layer to compare the influence of network sequence on the prediction results.

Table2. The prediction results in Model Selection.

Test Set	Model	Recall	ROC_AUC	PRC_AUC
Total test set	CnnCrispr	**0.857**	0.975	**0.679**
	CnnCrispr_NoLSTM	0.611	**0.987**	0.651
	CnnCrispr_Conv_LSTM	0.643	0.986	0.67
	CnnCrispr_NoBatchNor	-	0.5	0.504
	CnnCrispr_NoDropout	0.810	0.985	0.625
Hek293t test set	CnnCrispr	**0.864**	0.971	0.686
	CnnCrispr_NoLSTM	0.631	**0.988**	0.658
	CnnCrispr_Conv_LSTM	0.660	0.988	**0.694**

	CnnCrispr_NoBatchNor	-	0.5	0.504
	CnnCrispr_NoDropout	0.816	0.988	0.636
K562 test set	CnnCrispr	**0.826**	**0.995**	**0.688**
	CnnCrispr_NoLSTM	0.522	0.985	0.589
	CnnCrispr_Conv_LSTM	0.565	0.981	0.57
	CnnCrispr_NoBatchNor	-	0.5	0.503
	CnnCrispr_NoDropout	0.783	0.973	0.597

CnnCrispr had the best comprehensive performance in the three test sets, and the calculated recall value was higher than other pre-selected models.

Table3. Performance comparison with states-of-the-art models under test pattern 1.

Test Set	Model	auROC	auPRC	Pearson value	Spearman value
Total test set	CnnCrispr	0.975	**0.679**	**0.682**	**0.154**
	CFD	0.942	0.316	0.343	0.140
	MIT	0.77	0.044	0.150	0.085
	CNN_std	0.947	0.208	0.321	0.141
	DeepCrispr	**0.981**	0.497	-	0.133
Hek293t test set	CnnCrispr	0.971	**0.686**	**0.712**	**0.160**
	CFD	0.936	0.318	0.371	0.143
	MIT	0.756	0.048	0.153	0.084
	CNN_std	0.939	0.204	0.330	0.144
	DeepCrispr	**0.984**	0.521	-	0.136
K562 test set	CnnCrispr	**0.995**	**0.688**	**0.426**	**0.134**
	CFD	0.965	0.322	0.336	0.128
	MIT	0.814	0.033	0.057	0.086
	CNN_std	0.983	0.287	0.319	0.132
	DeepCrispr	0.953	0.41	-	0.126

We downloaded the prediction models of CFD, MIT and CNN_std from relevant websites and obtained the prediction results on the same test set as Cnncrispr. Since the training process of CnnCrispr was consistent with DeepCrispr's, we directly used the test results in **additional file 2** given by DeepCrispr for performance comparison. We bold out the optimal values under the same conditions in the table.

Table4. Different ratios indicate the possible relationship between three words, w_i, w_j, w_k.

$ratio_{i,j,k}$	w_i **is related to** w_k	w_i **is not related to** w_k
w_j **is related to** w_k	Tends to 1	Very small and tends to 0
w_j **is not related to** w_k	Greater than 1	Tends to 1

Figure Legends：

Fig. 1 The structure diagram of CnnCrispr. The other four pre-selected models were obtained by adding some parts on the basis of this frame. **a)** Rules for setting index values for different *r-d* pair, and an example of index representation for a sgRNA-DNA sequence pair. **b)** Based on index representation, the unsupervised GloVe model was used to train the embedded vectors and embed the sequence information into the new input matrix. **c)** biLSTM layer was used to capture context information in input information. **d)** CNN containing 5 layers with different kernels, scanning the above results to obtain different features. The last fully connected layers were used to obtain the final result.

Fig. 2 Visualization of auPRC and Recall value in three test sets. CnnCrispr had higher auPRC and Recall values in the three test sets. Although the performance of CnnCrispr_NoDropout was similar to CnnCrispr, the addition of Dropout layer can reduce the network parameters, shorten the training duration and prevent the over-fitting of the model to some extent, hence we finally chose the CnnCrispr with Dropout layer.

Fig. 3 ROC curves and PRC curves of CnnCrispr and three state-of-the-arts prediction models including CFD, MIT and CNN_std under test pattern 1.

Fig. 4 Performance comparison results of CnnCrispr and states-of-the-arts prediction models under test pattern 1. The two figures above shows the Pearson and Spearman values obtained by regression schema, the two figures below shows the AUC values under ROC and PRC curves by referring to the CRISTA's evaluation method.

Fig. 5 Leave-one-sgRNA-out comparison of sgRNA off-target propensity prediction with auROC, auPRC, Pearson correlation coefficient, and Spearman correlation coefficient. The bar chart shows the mean value obtained by 29-fold cross validation. The two figures above shows the performance comparison results in classification schema and the two figures below shows the performance comparison results in regression schema.

Fig. 6 Leave-one-sgRNA-out comparison of sgRNA off-target efficacy prediction with auROC, auPRC, Pearson correlation coefficient, and Spearman correlation coefficient. The 29 test results were plotted as the violin plots. The two figures above shows the performance comparison results in classification schema and the two figures below shows the performance comparison results in regression schema. It is worth noting that the article of DeepCrispr did not offer the detailed test results of leave-one-sgRNA-out validation, so we did not draw the corresponding violin plot. It is anticipated that this comparison process can further improve upon future works.

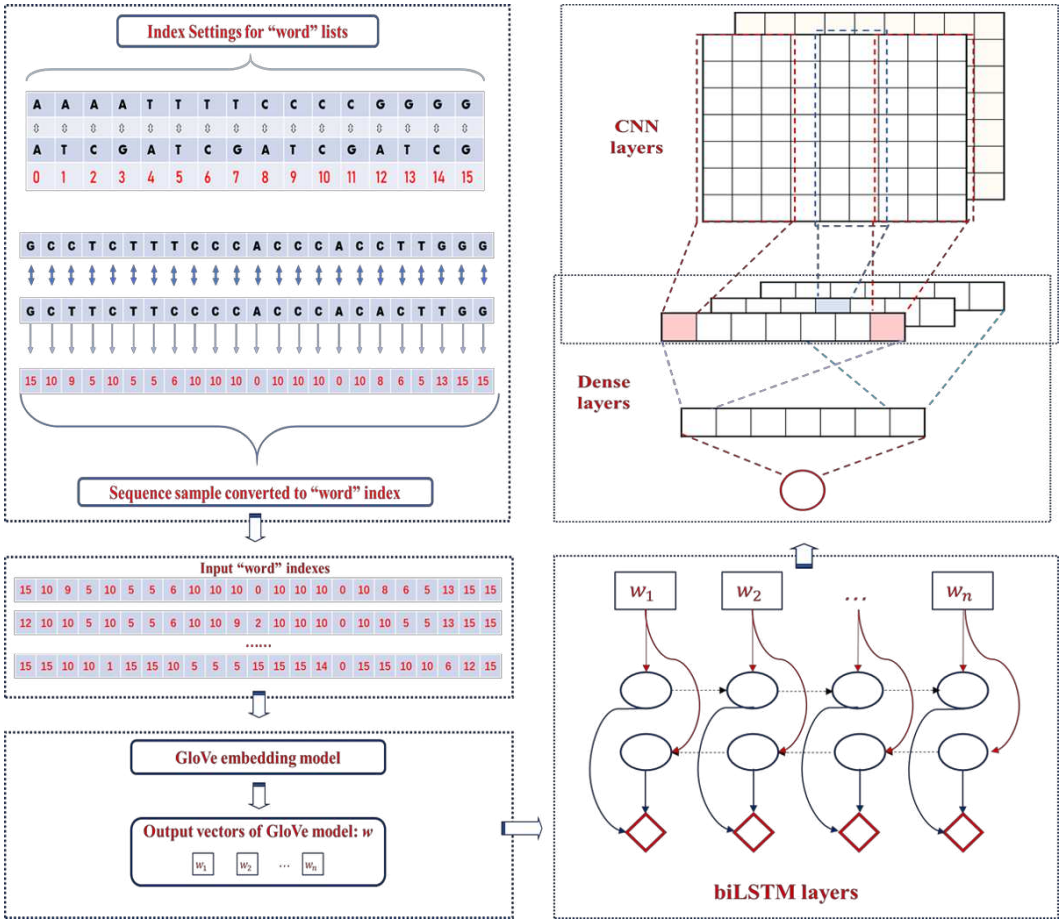


Fig. 1

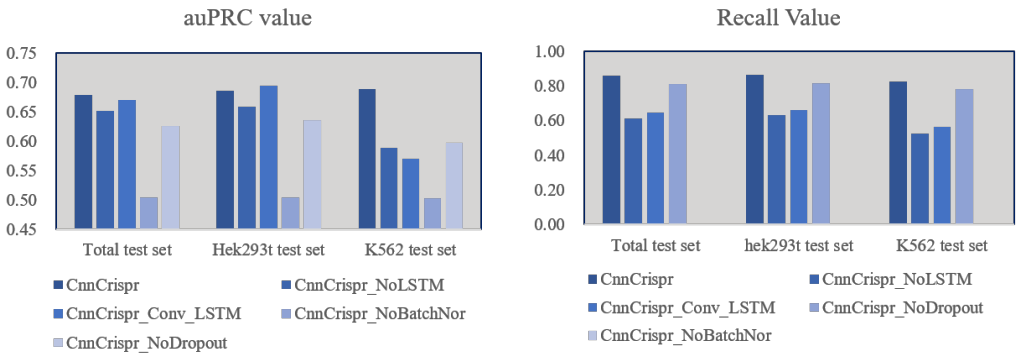
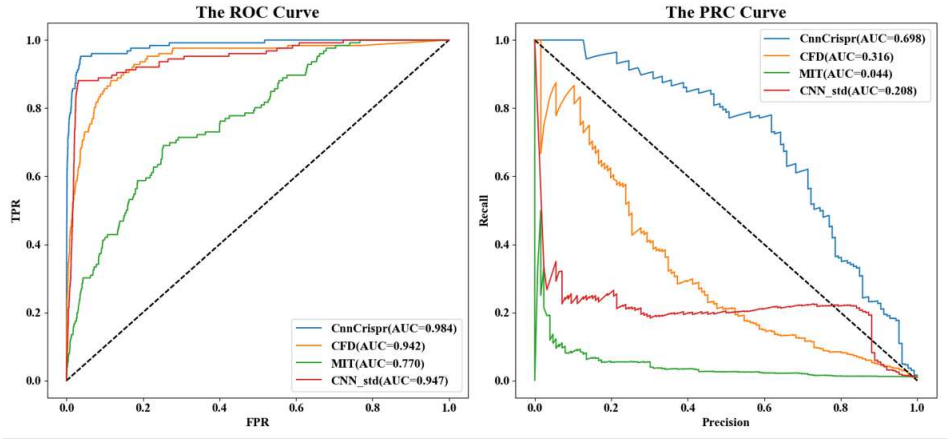
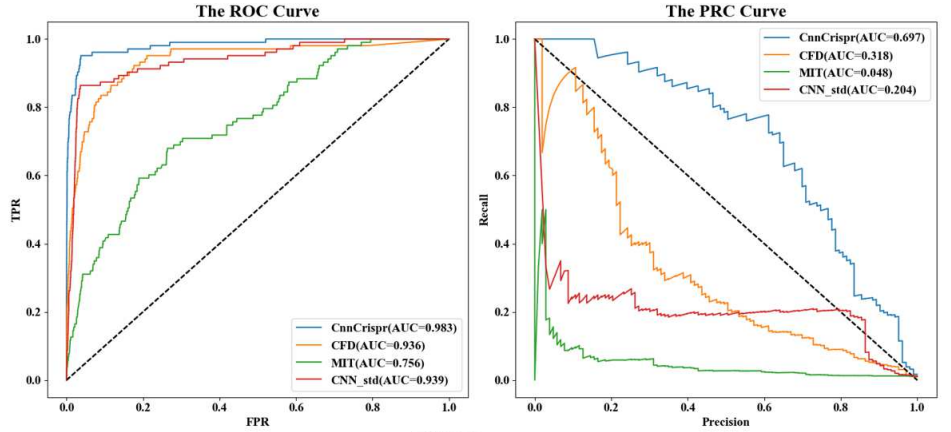


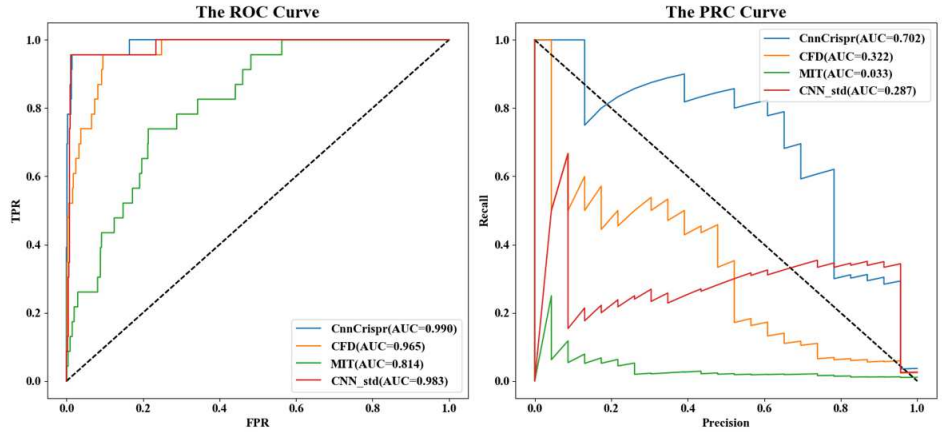
Fig. 2



Total test set



Hek293t test set



K562 test set

Fig. 3

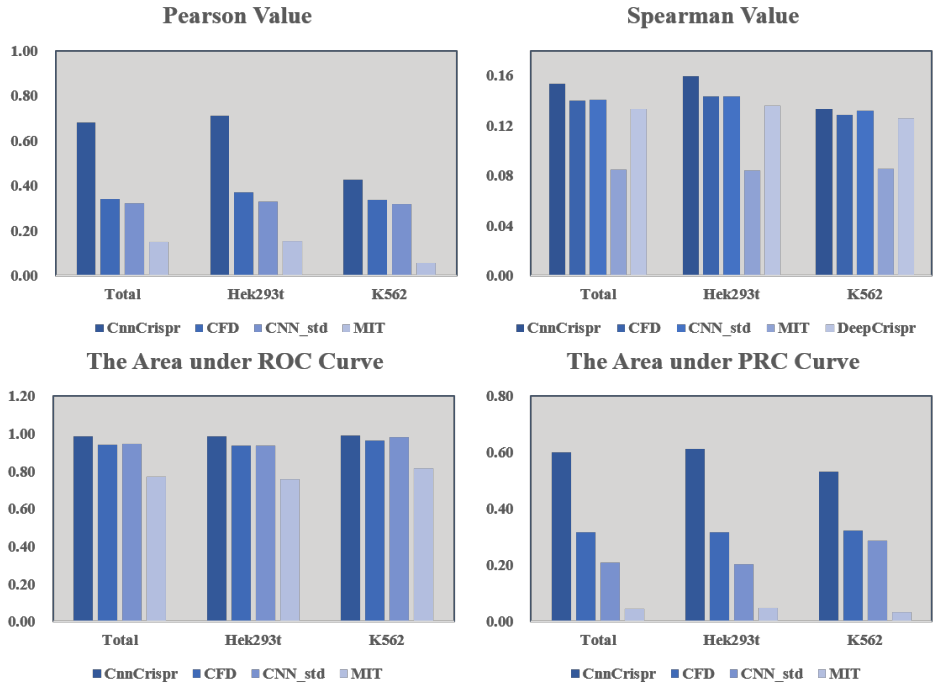

Pearson Value
1.00
0.80
0.60
0.40
0.20
0.00
Total
Hek293t
K562
CnnCrispr
CFD
CNN_std
MIT
Spearman Value
0.16
0.12
0.08
0.04
0.00
Total
Hek293t
K562
CnnCrispr
CFD
CNN_std
MIT
DeepCrispr
The Area under ROC Curve
1.20
1.00
0.80
0.60
0.40
0.20
0.00
Total
Hek293t
K562
CnnCrispr
CFD
CNN_std
MIT
The Area under PRC Curve
0.80
0.60
0.40
0.20
0.00
Total
Hek293t
K562
CnnCrispr
CFD
CNN_std
MIT

Fig. 4

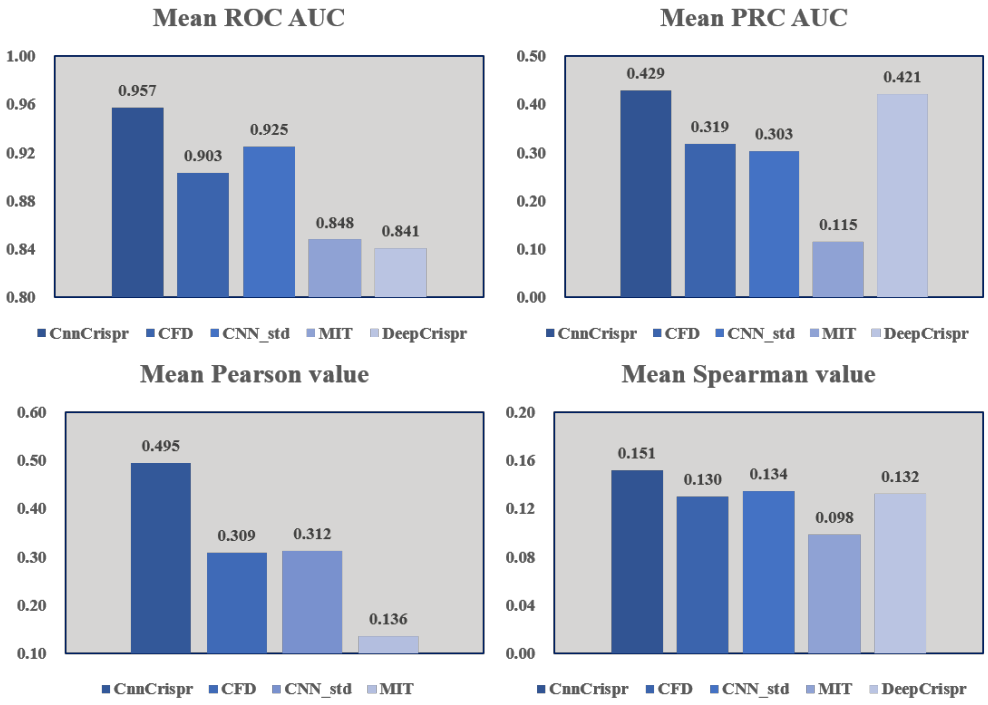

Mean ROC AUC
1.00
0.96
0.92
0.88
0.84
0.80
0.957
0.903
0.925
0.848
0.841
CnnCrispr
CFD
CNN_std
MIT
DeepCrispr
Mean PRC AUC
0.50
0.40
0.30
0.20
0.10
0.00
0.429
0.319
0.303
0.115
0.421
CnnCrispr
CFD
CNN_std
MIT
DeepCrispr
Mean Pearson value
0.60
0.50
0.40
0.30
0.20
0.10
0.495
0.309
0.312
0.136
CnnCrispr
CFD
CNN_std
MIT
Mean Spearman value
0.20
0.16
0.12
0.08
0.04
0.00
0.151
0.130
0.134
0.098
0.132
CnnCrispr
CFD
CNN_std
MIT
DeepCrispr

Fig. 5

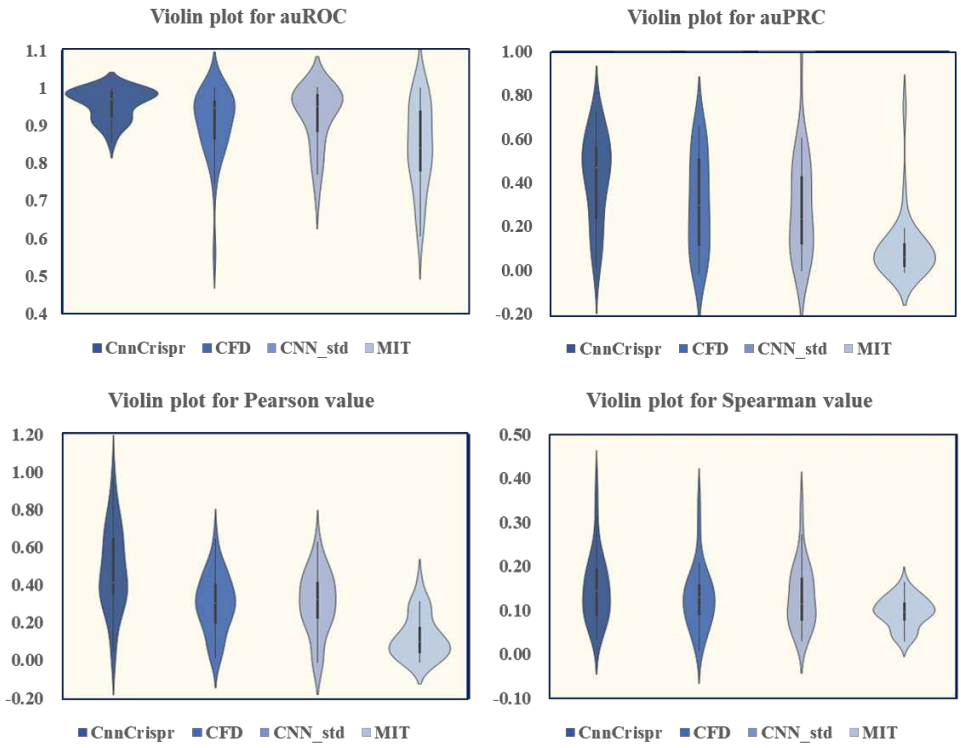

Violin plot for auROC
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
CnnCrispr CFD CNN_std MIT
Violin plot for auPRC
1.00
0.80
0.60
0.40
0.20
0.00
-0.20
CnnCrispr CFD CNN_std MIT
Violin plot for Pearson value
1.20
1.00
0.80
0.60
0.40
0.20
0.00
-0.20
CnnCrispr CFD CNN_std MIT
Violin plot for Spearman value
0.50
0.40
0.30
0.20
0.10
0.00
-0.10
CnnCrispr CFD CNN_std MIT

Fig. 6

Figures

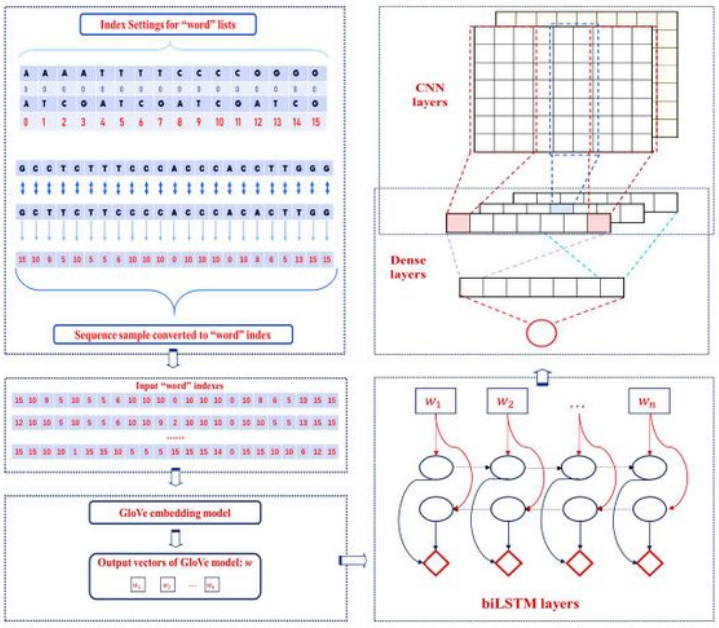


Figure 1

The structure diagram of CnnCrispr. The other four pre-selected models were obtained by adding some parts on the basis of this frame. a) Rules for setting index values for different r-d pair, and an example of index representation for a sgRNA-DNA sequence pair. b) Based on index representation, the unsupervised

GloVe model was used to train the embedded vectors and embed the sequence information into the new input matrix. c) biLSTM layer was used to capture context information in input information. d) CNN containing 5 layers with different kernels, scanning the above results to obtain different features. The last fully connected layers were used to obtain the final result.

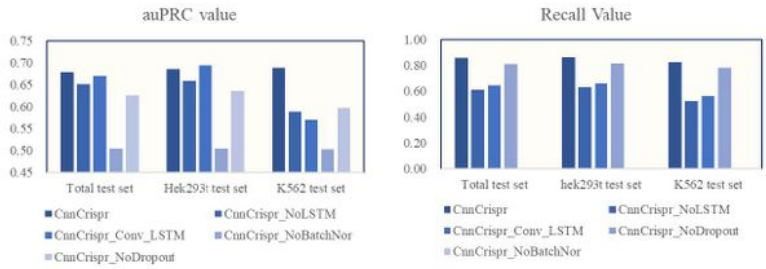


Figure 2

Visualization of auPRC and Recall value in three test sets. CnnCrispr had higher auPRC and Recall values in the three test sets. Although the performance of CnnCrispr_NoDropout was similar to CnnCrispr, the addition of Dropout layer can reduce the network parameters, shorten the training duration and prevent the over-fitting of the model to some extent, hence we finally chose the CnnCrispr with Dropout layer.

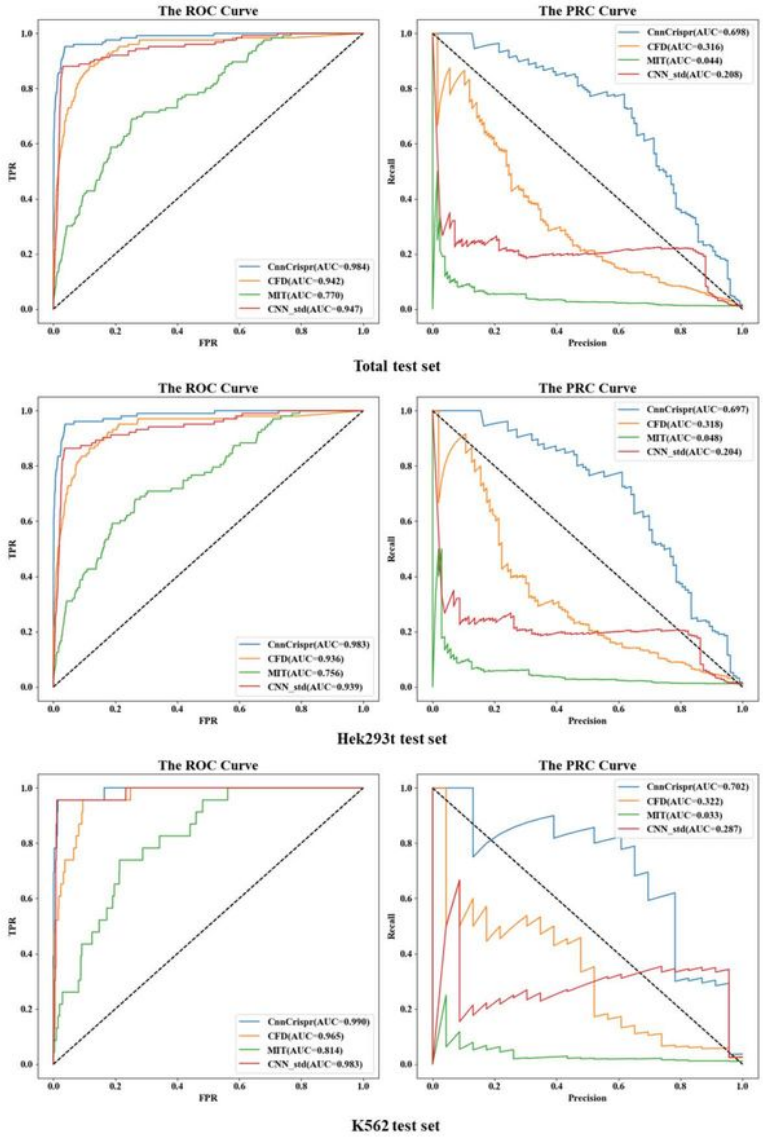


Figure 3

ROC curves and PRC curves of CnnCrispr and three state-of-the-arts prediction models including CFD, MIT and CNN_std under test pattern 1.

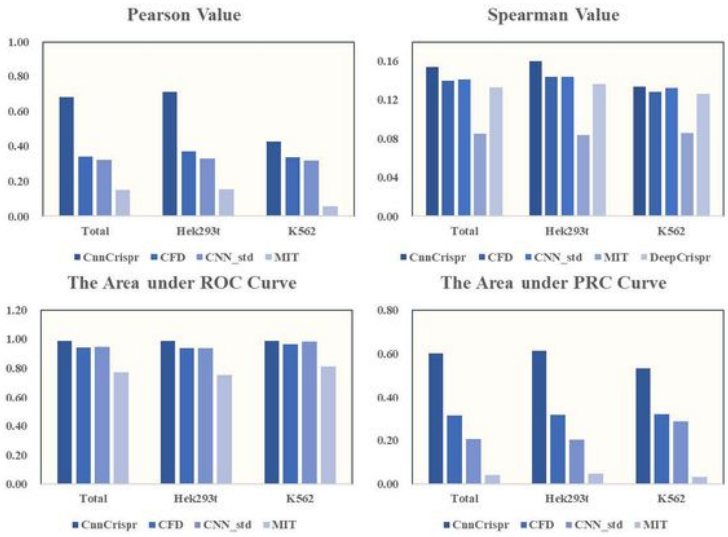


Figure 4

Performance comparison results of CnnCrispr and states-of-the-arts prediction models under test pattern 1. The two figures above shows the Pearson and Spearman values obtained by regression schema, the

two figures below shows the AUC values under ROC and PRC curves by referring to the CRISTA's evaluation method.

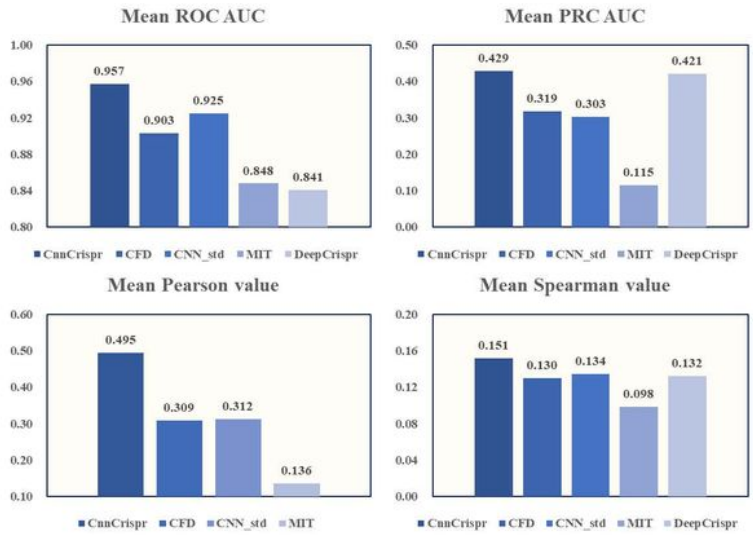


Figure 5

Leave-one-sgRNA-out comparison of sgRNA off-target propensity prediction with auROC, auPRC, Pearson correlation coefficient, and Spearman correlation coefficient. The bar chart shows the mean value obtained by 29-fold cross validation. The two figures above shows the performance comparison results

in classification schema and the two figures below shows the performance comparison results in regression schema.

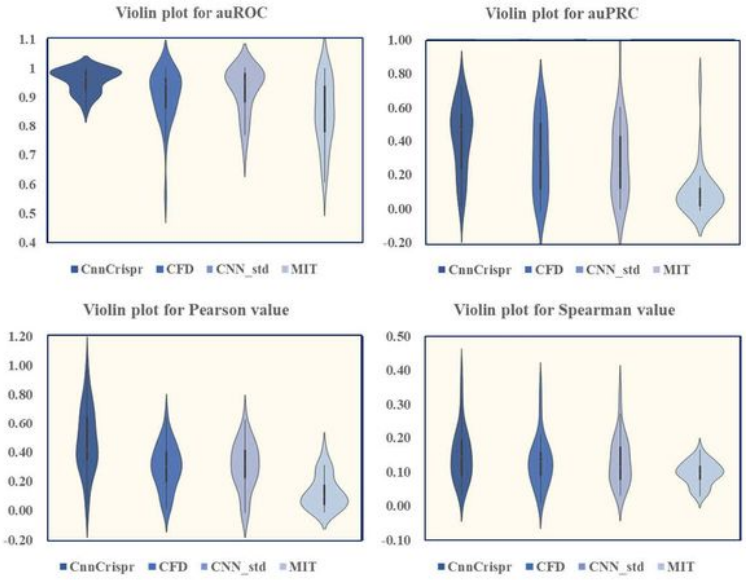


Figure 6

Leave-one-sgRNA-out comparison of sgRNA off-target efficacy prediction with auROC, auPRC, Pearson correlation coefficient, and Spearman correlation coefficient. The 29 test results were plotted as the violin plots. The two figures above shows the performance comparison results in classification schema and the

two figures below shows the performance comparison results in regression schema. It is worth noting that the article of DeepCrispr did not offer the detailed test results of leave-one-sgRNA-out validation, so we did not draw the corresponding violin plot. It is anticipated that this comparison process can further improve upon future works.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- Additionalfile2.pdf
- Additionalfile1.xlsx