Distinguishing highly and lowly active CRISPR/Cas9 sgRNA via Inception-based Deep Convolutional Neural Network

Bogdan Kirillov

National Research University Higher School of Economics, Moscow, Russia, bakirillov@edu.hse.ru

Clustered Regularly Interspaced Short Palindromic Repeats (CRISPR) are special components of bacterial immune system that have recently drawn attention of scientists from wide spectrum of fields ranging from evolutionary biology to bioengineering and bioinformatics. CRISPR/Cas systems became crucial part of modern approaches for genome editing. Why? It allows bacteria to identify, destroy and remember various bacteriophages and can be used as a mean for very precise *in vitro* and *in vivo* DNA manipulation [1, 2]. There are a lot of Cas systems (read [3] for a good classification of known types) but the most widely used in applications is CRISPR/Cas9.

Cas9 is the main agent that does all the job in editing. It performs a double stranded break in a region of DNA that is specified by its single guide RNA (sgRNA) – a synthetic RNA 20 nucleotides long that should be complementary to the region of interest. One of the CRISPR/Cas9 drawbacks is unclear dependence of Cas9 activity on sgRNA nucleotide sequence.

Highly active Cas9 is more likely to perform a cleavage in the right place (it is on-target cleavage, Cas9 can break DNA in the wrong place, which is off-target cleavage and is not discussed in this paper). The discrimination of high active and low active sgRNA is a challenging Machine Learning task.

A lot of work is done to understand and predict sgRNA activity, but existing tools still are pretty limited in their performance. For example, recent (2017) sgRNA Scorer 2.0 [4] yields 73.7% accuracy which is worse than geCRISPR [5] (pipeline geCRISPRc, 85.54%) that came out in 2016.

sgRNA Scorer 2.0 and geCRISPRc are based on SVM. Current work uses Deep Neural Network-based approach and outperforms both sgRNA Scorer 2.0 and geCRISPRc.

The problem of activity classification is stated as follows:

1. The sgRNA is represented as string twenty nucleotides long;
2. It can be of high activity (class label 1) or low activity (class label 0);
3. The goal is to predict the class label given the string;
4. The output is a pair of probabilities – probability of the sgRNA being active and inactive. This pair can later be converted into class labels.

For the current study the strings of {A, T, G, C}'s are encoded via one-hot encoding as shown on figure 1:

Figure 1: One hot encoding. White is 1, black is 0. C = [1,0,0,0], G = [0,1,0,0], T = [0,0,1,0], A=[0,0,0,1].

One-hot encoding basically means that the sgRNA string is a black and white picture of size 4x20. There is a class of Deep Learning models that is particularly useful for working with pictures – Convolutional Neural Networks.

In this study the classifier used is a Deep Convolutional Neural Network (DCNN) based on so-called Inception module [6]. The main function of the module is simultaneous application of different convolution kernels and aggregation of the explored features to use in the latter layers of the network. This allows the model to mine from picture more features that can be of predictive value. The architecture of DCNN is shown on figure 2 below:

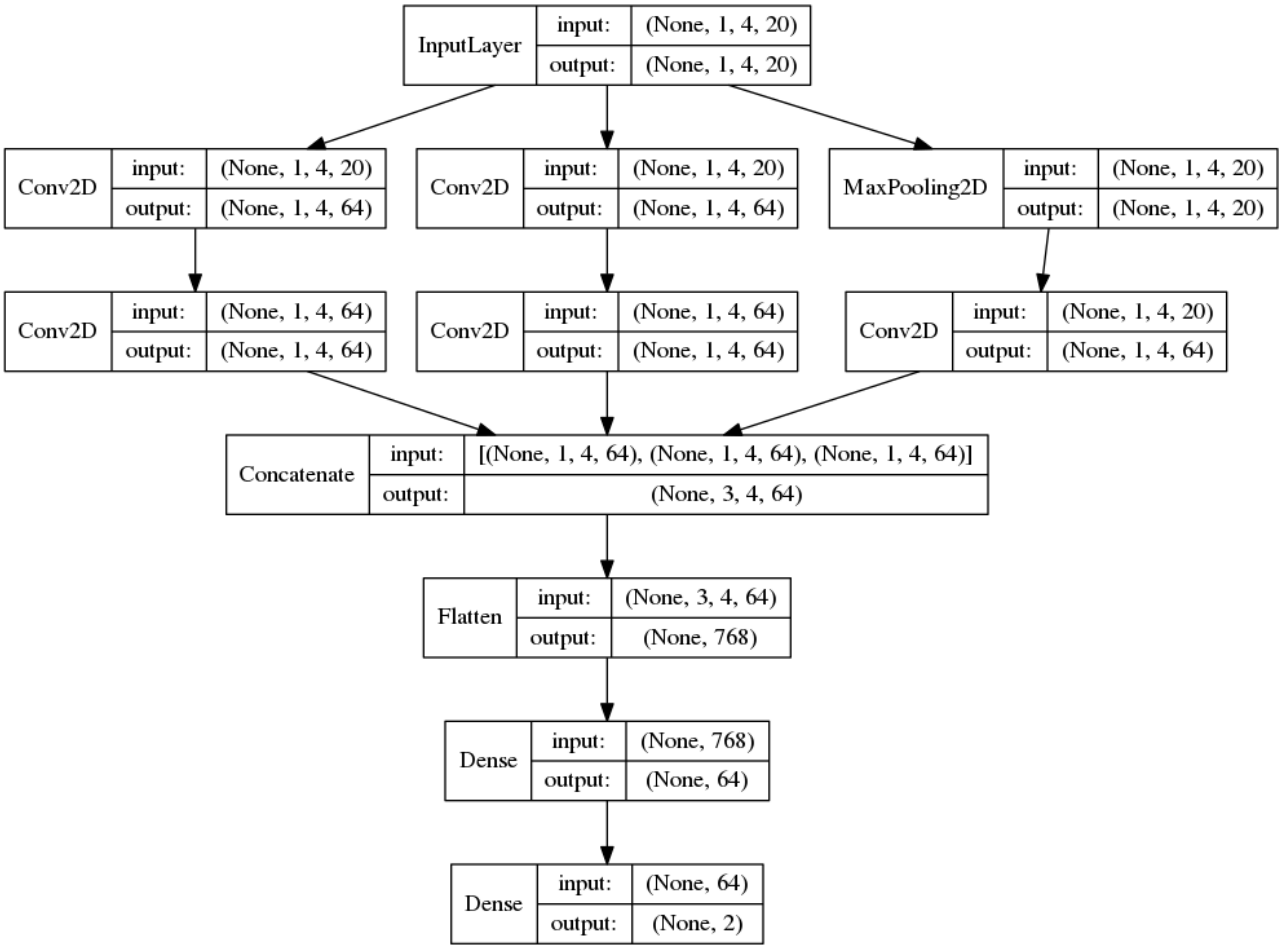

Figure 2: The DCNN's architecture. Activation functions are ReLU everywhere except output layer which is Softmax.

This Inception module consists of three parallel parts – a combination of 1x1 and 3x3 convolutions, a combination of 1x1 and 5x5 convolutions, and a combination of 1x1 convolution and 3x3 max pooling layer. This allows the consequent single hidden-layer feed forward neural network with 64 neurons in hidden layer to get wide range of possible features that may be present in the sgRNA picture. The best network was online (batch size is 1) trained for 32 epochs via AdaGrad method and shows the following behavior in terms of accuracy and loss function (categorical crossentropy):

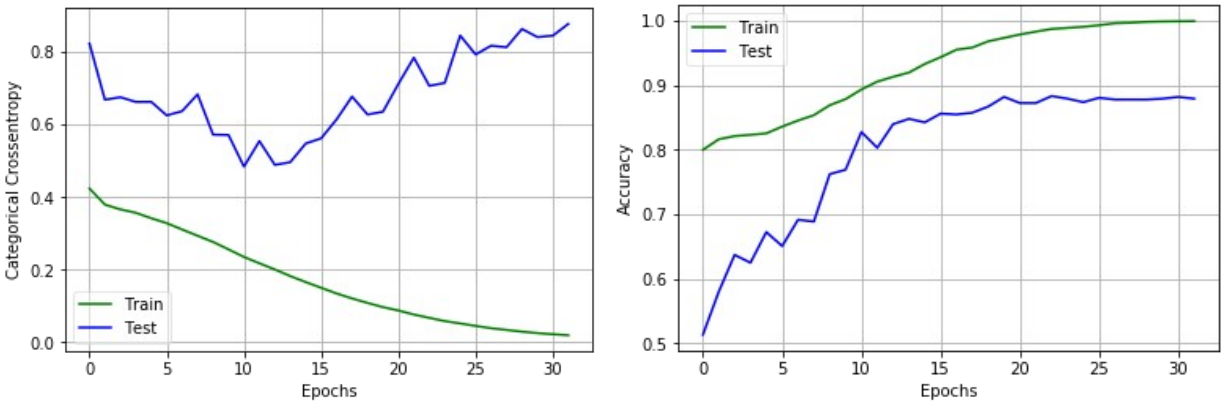

Figure 3: Network behavior during training. Slight overfitting due to small (for Deep Learning) sample size can be noticed but it did no harm for network's performance.

The network was built using Keras [7] with Tensorflow [8] backend. The data about its behavior and example of one hot sgRNA encoding were visualized via Matplotlib [9].

The data used to train the network are the same that were used to train geCRISPR - 2090 and 4139 experimentally verified sgRNAs from *Homo sapiens*, *Mus musculus*, *Danio rerio* and *Xenopus tropicalis*. The data can be accessed at [10].

The network was trained on regression part of geCRISPR dataset (T3619 – 3619 sgRNAs) – raw activity values (ranging from 0 to 1) were converted to class labels the following way: if sgRNA activity is smaller than 0.5, it is considered low and this string gets a label of 0, otherwise – label of 1.

geCRISPRc was trained on T1840 dataset – classification part of 1840 labelled sgRNA. The DCNN has never seen any part of this dataset during training and still outperforms (in terms of accuracy) both best geCRISPRc (based on dinucleotide one hot encodings) and mononucleotide one hot-based geCRISPR. Matthews Correlation (MCC) and Area-under-Curve (AUC) were evaluated on V250 – validation dataset (250 labelled sgRNA previously not seen by the models). The results are shown in Table 1.

Accuracy of sgRNA Scorer 2.0 (73.7%) is not shown, because DCNN and geCRISPR were trained on the same dataset (different parts of geCRISPR data) therefore are more directly comparable.

Table 1: Performance of inception-based model compared to not-DNN models, evaluated on the same datasets.

Model	Accuracy on T3619, %	Accuracy on T1840, %	Accuracy on V250, %	MCC on V250	AUC on V250
geCRISPRc mono-binary	NA	85.54	90.00	0.80	0.95
Best geCRISPRc (dinucleotide binary)	NA	87.15	88,80	0.78	0.94
Inception-based DCNN	**99.99**	**88.53**	**88.40**	**0.784**	**0.90**

Those results are state-of-the-art for sgRNA activity classification and yet it is simple deep model with only one Inception module. Currently much deeper models are used in various applications of Computer Vision. There is a lot of to improve upon. For example, the more interesting task is to predict not a class of activity, but actual activity (probability of successful cleavage), as geCRISPRr does. This might involve different (perhaps deeper) network configuration, because regression is harder to perform for neural networks than classification.

References:

1. Sternberg S. H., Doudna J. A. (2015) Expanding the biologist's toolkit with CRISPR-Cas9, *Molecular cell*, 58, 4: 568–574.
2. Dow L. E. et al. (2015) Inducible in vivo genome editing with CRISPR-Cas9, *Nature biotechnology*. 33, 4: 390-394.
3. Makarova, K. S., et al. (2015) An updated evolutionary classification of CRISPR-Cas systems, *Nature Reviews Microbiology*.
4. Chari, Raj, et al. (2017) sgRNA Scorer 2.0: A Species-Independent Model To Predict CRISPR/Cas9 Activity. *ACS Synthetic Biology*.
5. Kaur K. et al. (2016) ge-CRISPR-An integrated pipeline for the prediction and analysis of sgRNAs genome editing efficiency for CRISPR/Cas system, *Scientific reports.* 6.
6. Szegedy, C, et al. (2015) Going deeper with convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.
7. https://keras.io/
8. Abadi M. et al. (2015) TensorFlow: Large-scale machine learning on heterogeneous systems, Software available from tensorflow.org.
9. Hunter, J. D. (2007) Matplotlib: A 2D graphics environment, *Computing In Science & Engineering*, 9.3: 90-95.
10. http://bioinfo.imtech.res.in/manojk/gecrispr/dataset.php