Predicting CRISPR Activity by Target Sequence *a priori* Using NLP Models

Ian Hay, Jaeson Pyeon, and Zain Alam

April 19th, 2023

Background

In 2014, chemists Jennifer Doudna and Emmanuelle Charpentier discovered how to utilize CRISPR, or clustered regularly interspaced short palindromic repeats, to efficiently edit any gene in an organism's genome. By fusing the different ribonucleic acid (RNA) components into a single guide RNA (sgRNA), the system could target any conceivable region of deoxyribonucleic acid (DNA) in a gene (Jinek et. al, 2012). Their experiments were so significant they were awarded the Nobel Prize in Chemistry in 2020, only 6 years after their discovery. The ability to engineer the genome opened up a flurry of biotechnology research into studying and curing genetic disorders that impact millions of individuals worldwide.

However, despite this monumental technological advancement, treatments for genetic diseases using CRISPR engineering remain off the market. This in itself is owed to many factors, including necessary regulation and clinical testing to ensure a treatment is safe and effective before being marketed to individuals with genetic diseases. There is another issue at play, and it's due to the expansive nature of CRISPR to target any conceivable region of DNA: there are a lot of potential targets even within a particular gene. The target sgRNA is only 20 nucleotides long, and the gene of interest may be hundreds or thousands of base-pairs. Not all guide RNAs are created equal and some sequences have proven more efficient editors than others within a particular gene. Additionally, there is an essential error to take account when divulging into genetic engineering - off-target edits. When looking at laboratory research and especially therapeutic development, it is essential to minimize any opportunity for off-target mutations that can lead to devastating consequences including cancer. As such, it is essential for genetic therapies to select the optimal guide RNA for their target gene (Mohr et. al, 2016).

To aid in research and development for genetic therapies, computational biologists have begun developing tools for selecting the optimal guide RNA sequence to target a given sequence. Many of these models rely on manually extracted features from the sequence, including one-hot encoding for nucleotides by position, as well as additional thermodynamic features such as the melting point of the target sequence. These features can be time and computationally consuming to generate (Jun Wang et. al, 2020).

Here, we propose to use natural language processing to treat the sgRNA sequence as a sequence of tokens. We will train a Word2Vec embedding model to generate vector representations of each of the four nucleotides adenine (A), cytosine (C), guanine (G), and thymine (T). These embeddings will then be fed into neural networks to learn the experimental sgRNA efficiency scores, and compared with published models trained using the manually extracted features.

Data Sources

To train our models and compare with the state-of-the-art, we sourced data from published papers (Doench et. al, 2014; Doench et. al, 2016; Hart et. al, 2015; Fusi et. al, 2015; Tim Wang et al, 2014; Chari et. at, 2015). These data include CRISPR screens for guide RNAs by measuring their on-target knockout efficacy through laboratory experiments; knockout refers to the ability of genes edited by CRISPR to lose function, indicating successful on-target editing. Additionally, we used one supplemental data source screening guide RNAs for off-target editing (Shalom et. al, 2013). To adequately compare the experimental results from each study, the on- and off-target scores were normalized using Equation 1 to be between 0 and 1, continuously, prior to concatenating the datasets.

$$D = \frac{r - \frac{3}{8}}{n + \frac{1}{4}}$$

Equation 1: The normalization equation used, where *D* is the normalized score, *r* is this score rank among its dataset, and *n* is the number of samples in the dataset.

sgRNA Sequence	sgRNA on/off-target Score
ATGGCTTCCTCGTGAGTTGG	0.410978

Table 1: Representative datapoint sourced from the data sources above after normalization.

In total, we sourced 24,187 on-target and 64,751 off-target sgRNA sequences with associated experimental editing efficiency scores. Each sequence is 20 bases long, ending with the 3-nucleotide NGG motif necessary for CRISPR functionality where N is any nucleotide in {A,C,G,T}. After normalization, all scores fall between [0, 1] with a roughly equal frequency of scores across the distribution, representative of their percentile within their experimental dataset.

Modeling

The goal for the model was to predict both on target and off target editing activity of a particular guide RNA sequence by generating word embeddings for each of the four nucleotides. We encoded the embeddings using a Word2Vec model, which generated 25-length vectors for each nucleotide (“Gensim: Topic Modelling for Humans.”). The sequence embeddings were made by concatenating the encoding for each of the 20 nucleotides in a sequence. Additional embeddings were produced by a pre-trained BERT transformer Dilstilbert, sourced from the HuggingFace library of sentence transformers (“Distilbert.”) . Each set of embeddings were then fed into a feedforward neural network programmed using the TensorFlow library’s dense layers (“Tf.keras.layers.dense : Tensorflow V2.12.0.”). For predictions of unseen sequences, the trained Word2Vec embeddings would be used to generate their corresponding sequence embeddings and be fed into the trained feed forward neural network. With only four nucleotides, the model architecture and embeddings do not have functionality for unseen nucleotides as those are not found in nature.

We trained the neural network to predict sgRNA activity based on embedded sequence data using the experimental results sourced and described above. The input data is split into training and validation sets and the input sequences are converted from strings to arrays and then fed into the neural network model. The model architecture is defined using TensorFlow, which includes a Flatten layer to flatten the input arrays, 3 Dense layers with 128 neurons and ReLU activation function each, and a final Dense layer with a single neuron for regression output. The model is compiled with a mean squared error loss function and Adam optimizer. The fit method is called to train the model on the training set, with 100 epochs and a batch size of 256.

To compare with previously published results, we also extracted features manually using one-hot positional encoding for both the on-target and off-target datasets. First, a version of features were extracted with unigrams and bigrams for each combination of nucleotides in each of the 20 positions of the sgRNA sequence, as well as a binary variable for each that is 1 if the combination occurs at any position in the sequence and 0 otherwise. Additional melting temperature features were extracted using the BioPython package, as well as 3 features for the GC content of the sequence (i.e., the number of nucleotides in the sequence that are G or C) (“Biopython.”) . This gave us 413 features initially. We then extended these features with trigram combinations for each position, as well as a parameter extracted from the OligoArrayAux package to estimate the Gibbs free energy of binding to the DNA sequence (*Oligoarrayaux*). This gave us 1,634 features. The manually encoded features were then fed into a Bayesian ridge linear regression and a gradient boosted regression tree (GBRT), both implemented using the Scikit Learn library. The work here for comparison is based on the GNL Scorer paper (Jun Wang et. al, 2020).

Results & Discussion

Based on the model comparisons, we found that for on-target predictions, the Word2Vec embeddings into the feed forward network and gradient boosted regression tree on manual features performed the best, with Spearman correlation coefficients of ~0.45 achieved via 10-fold cross-validation. When the various models were plotted based on predicted and true sgRNA score, there were flat slopes concentrated around the average score of 0.5 for the manually extracted feature models, meaning that the predictions here are not following the sequence correctly. On the other hand, the neural network models trained on embeddings had more variable predictions that were nevertheless inaccurate. The BERT embeddings performed worst of all.

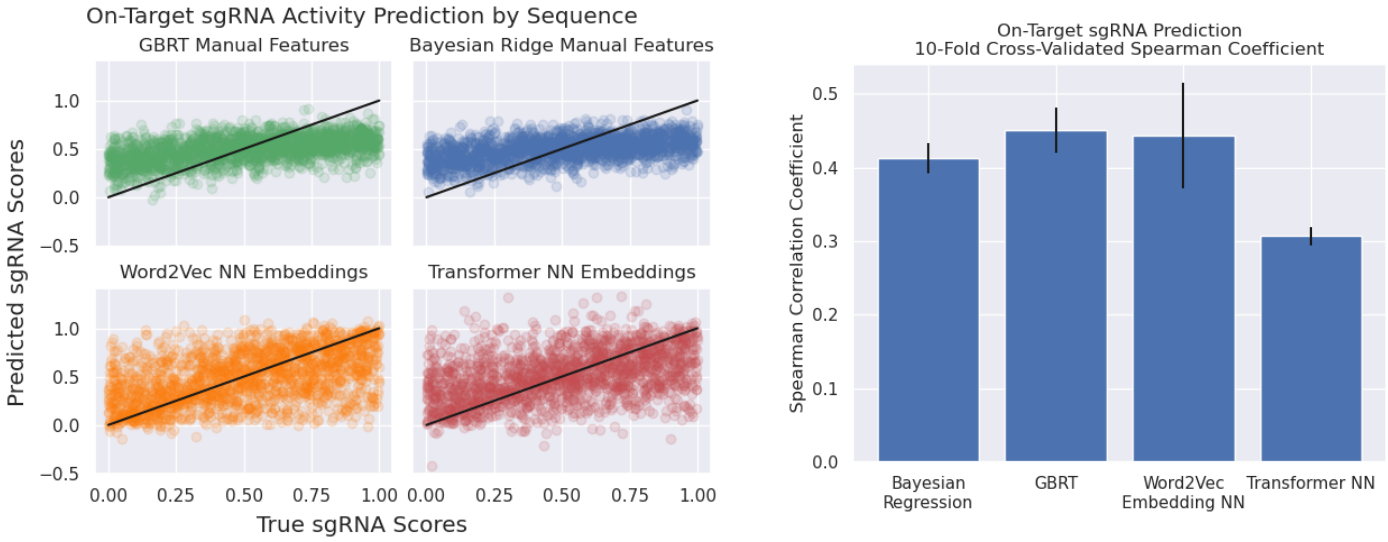

Figure 1: On-target prediction results from the testing data on the left, and the 10-fold cross-validation Spearman correlation coefficient for the test set on the right. The solid black line in the left graphs shows the true testing data results. The data on the right bar chart shows the 10-fold mean, with error bars denoting standard error of the mean.

For the off-target predictions, they showed a much higher learnability. Both the manually extracted features fed into GBRT and Bayesian ridge regression were comparable to the performance of the Word2Vec & BERT embeddings and feed forward neural network. Overall, these models' predictions showed a much higher correlation to the experimental data with a coefficient of less than 0.8 for each on the testing data.

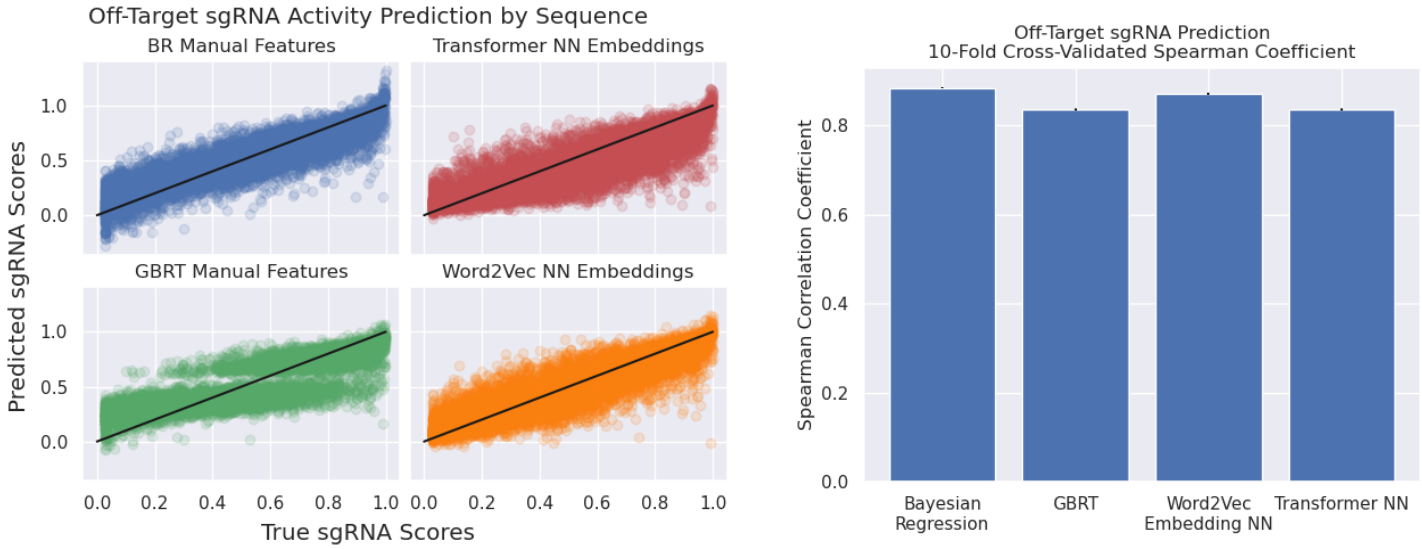

Figure 2: Off-target prediction results from the testing data on the left, and the 10-fold cross-validation Spearman correlation coefficient for the test set on the right. The solid black line in the left graphs shows the true testing data results. The data on the right bar chart shows the 10-fold mean, with error bars denoting standard error of the mean.

Future Directions

There are several potential improvements that could be explored to enhance the performance of feature extraction methods in NLP. One approach is to include more diverse sources of data to improve model generalization, particularly for the off-target data prediction. A potential explanation for the improved learnability of the off-target data when compared to on-target is the single data source for off-target sgRNA activity, while the on-target data was sourced from numerous experimental results. Another potential improvement is to train a transformer encoder/decoder specifically for embeddings, rather than relying on a pre-trained transformer. This could help optimize the embeddings for a specific task or domain, and could be accomplished via transfer learning from human genome sequences (Chuai, 2018). Additionally, adding more parameters to pass into embeddings, such as the sequence surrounding the target sequence or thermodynamic properties, could further improve performance. Finally, hyperparameter tuning and experimenting with different neural network architectures could also help improve the accuracy of feature extraction methods, particularly the comparison of feed forward networks with a recurrent network using long short term memory (LSTM) modules.

Works Cited

"Biopython." *Biopython · Biopython*, https://biopython.org/.

Chuai, Guohui, et al. "DEEPCRISPR: OPTIMIZED CRISPR Guide RNA Design by Deep Learning - Genome Biology." *BioMed Central*, BioMed Central, 26 June 2018, https://genomebiology.biomedcentral.com/articles/10.1186/s13059-018-1459-4.

Chari, Raj et al. "Unraveling CRISPR-Cas9 genome engineering parameters via a library-on-library approach." *Nature methods* vol. 12,9 (2015): 823-6. doi:10.1038/nmeth.3473

"Distilbert." *DistilBERT*, https://huggingface.co/docs/transformers/model_doc/distilbert.

Doench, J., et al. "Rational design of highly active sgRNAs for CRISPR-Cas9–mediated gene inactivation." *Nature Biotechnology*, Nature Publishing Group, 2014, https://doi.org/10.1038/nbt.3026

Doench, John G, et al. "Optimized Sgrna Design to Maximize Activity and Minimize off-Target Effects of CRISPR-Cas9." *Nature News*, Nature Publishing Group, 18 Jan. 2016, https://www.nature.com/articles/nbt.3437.

"Gensim: Topic Modelling for Humans." *Models.word2vec – Word2vec Embeddings - Gensim*, 21 Dec. 2022, https://radimrehurek.com/gensim/models/word2vec.html.

Hart, Traver. *High-Resolution CRISPR Screens Reveal Fitness Genes and Genotype ...* Cell, 25 Nov. 2015, https://www.cell.com/fulltext/S0092-8674(15)01495-6.

Jinek, Martin, et al. "A Programmable Dual-RNA-Guided DNA Endonuclease in Adaptive Bacterial Immunity." *Science (New York, N.Y.)*, U.S. National Library of Medicine, 17 Aug. 2012, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6286148/.

Mohr, Stephanie E, et al. "CRISPR GUIDE RNA Design for Research Applications." *The FEBS Journal*, U.S. National Library of Medicine, Sept. 2016, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5014588/.

Oligoarrayaux, http://www.unafold.org/Dinamelt/software/oligoarrayaux.php.

"Scikit-Learn." *Scikit*, https://scikit-learn.org/stable/index.html.

Shalem O;Sanjana NE;Hartenian E;Shi X;Scott DA;Mikkelson T;Heckl D;Ebert BL;Root DE;Doench JG;Zhang F; "Genome-Scale CRISPR-Cas9 Knockout Screening in Human Cells." *Science (New York, N.Y.)*, U.S. National Library of Medicine, https://pubmed.ncbi.nlm.nih.gov/24336571/.

"Tf.keras.layers.dense : Tensorflow V2.12.0." *TensorFlow*, https://www.tensorflow.org/api_docs/python/tf/keras/layers/Dense.

Wang, Tim et al. "Genetic screens in human cells using the CRISPR-Cas9 system." *Science (New York, N.Y.)* vol. 343,6166 (2014): 80-4. doi:10.1126/science.1246981

Wang, Jun, et al. "GNL-Scorer: A Generalized Model for Predicting CRISPR on-Target Activity by Machine Learning and Featurization." *Journal of Molecular Cell Biology*, U.S. National Library of Medicine, 3 Dec. 2020, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7883820/.