# Predicting Human CRISPR Gene-Editing Efficiency with Deep Learning

Max O'Meara[1], Rehan Samaratunga[2]

*Boston University, Massachusetts, USA*[1]

## Introduction

CRISPR gene editing or CRISPR for short refers to a biotechnological tool used to make precise changes to DNA. By editing a gene we can cut away unhealthy mutations or change it's function so the produced cells are altered with new DNA. CRISPR can be used to change traits in almost any kind of organism making it a potential tool for clinical uses in the biomedical sciences.

CRISPR technology is based on the immune system response that bacteria uses to fight off viruses. It works by using a CRISPR-associated protein called Cas9 to make a precise cut across both strands of the DNA. Once the strand is cut, a new custom sequence can be added when the DNA is repaired.

## Motivation

The purpose of this project is an attempt to save resources in labs. When biotech companies are designing new therapies, they're targeting genes they've never screened before. Instead of testing dozens of DNA tags to find one that works, researchers can start with the most promising options our model suggests. That means fewer trial and error experiments, faster progress on basic biology studies, and a smoother path for teams using this tool for disease research, drug studies, or classroom labs.

This reduces library size, sequencing, and hands-on work, freeing time for confirmatory tests and follow-up biology. It also gives a clear, repeatable ranking method that helps small labs and teaching settings. Because the setup is fully configurable by each parameter, teams can run quick design–test–update cycles to steadily improve guide selection.

## Research Objectives

We use the GenomeCRISPR dropout screen data as our primary corpus, consisting of sequence-level measurements of 23-nt guides with associated log2 fold changes. We treat cell line and phenotype variations as out-of-domain conditions to evaluate generalization.

- $RQ_1$: What is the baseline performance of a sequence-only prediction model?
- $RQ_2$: Does adding experimental context (cell line, phenotype, chromosome, strand) improve prediction accuracy?
- $RQ_3$: Can the model generalize to sgRNAs for genes not present in the training set?
- $RQ_4$: Does performance transfer across cell lines, and what features matter most?

## Dataset Overview

The dataset we used for this project was the publicly available GenomeCRISPR dataset. It is a database for high-throughput CRISPR/Cas9 screening experiments. Currently, it contains data on the performance of approximately $700,000$ DNA sequences used in around $500$ different experiments performed in $421$ different human cell lines.

The dataset contains a multitude of variables from the experiments but we are choosing the inputs listed in Table 1 as we are choosing to exclude gene specific information. The goal of this is to push the model to generalize to genes it hasn't seen before. Doing so, it must rely on sequence based rules that transfer to other genes rather than relying on properties of specific genes.

Table 1. Input features used in our model.

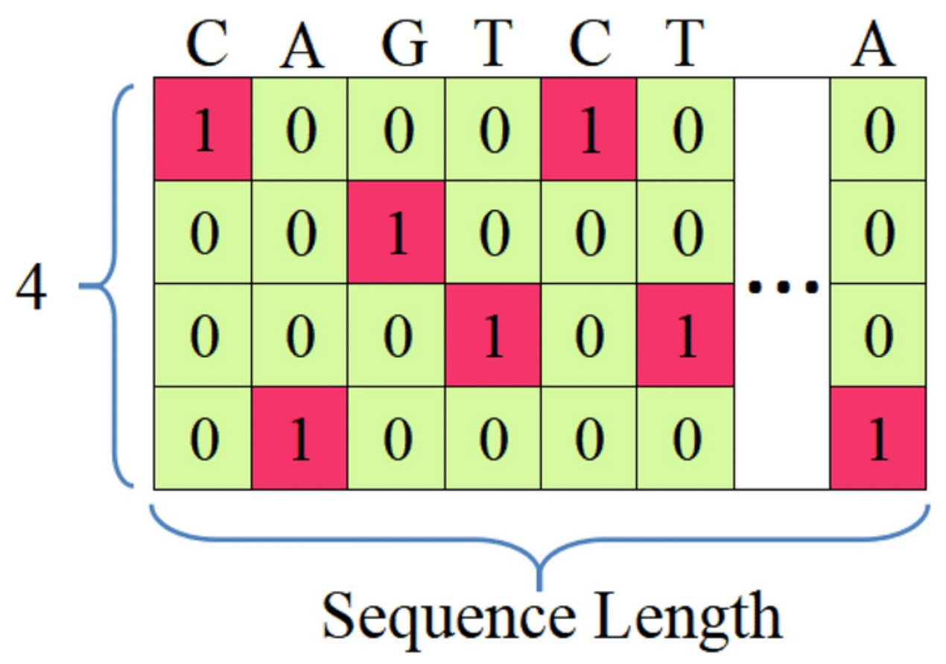
| **Feature** | **Type** | **Description** |
|---|---|---|
| Sequence | 23 chars | Guide RNA target sequence |
| Cell line | categorical | Source cell type |
| Phenotype | categorical | Dropout (–) or enrichment (+) |
| Chromosome | categorical | Genomic location |
| Target | real scalar | $\log_2$ fold change value |

## Sequence and Metadata Encoding

Each sgRNA is a 23-nt string $s_1, \ldots, s_{23} \in \{A, C, G, T\}$. We convert this into a $4 \times 23$ one-hot tensor $X$ with channels (A, C, G, T), where $X_{k,j} = 1$ if base $s_j$ equals base $k$ and $0$ otherwise.

To remove strand bias while preserving biological meaning, we normalized sequence orientation by reverse complementing all "$-$" strand sequences. After this transformation, all guides are represented in the "$+$" direction, ensuring a uniform $5' \rightarrow 3'$ orientation.

For cell line, phenotype, and chromosome we treat each as a categorical variable and map every unique value to an integer ID using `pandas.factorize`. This yields three index features: cell line ID, phenotype ID, and chromosome ID.



$$
\begin{array}{c|cccccccc|}
 & C & A & G & T & C & T & & A \\
\hline
 & 1 & 0 & 0 & 0 & 1 & 0 & & 0 \\
4 & 0 & 0 & 1 & 0 & 0 & 0 & \cdots & 0 \\
 & 0 & 0 & 0 & 1 & 0 & 1 & & 0 \\
 & 0 & 1 & 0 & 0 & 0 & 0 & & 1 \\
\end{array}
$$

Figure 1: Each sequence is encoded as a $4 \times 23$ one-hot tensor.

## Train/Test Split Strategy

To obtain reliable estimates of model generalization, we partitioned our dataset into training, validation and testing using an $80/10/10$ split. Splits were performed using distinct index sets so no sequence appears in more than one split. The validation set is used for early stopping and hyperparameter tuning, while the test set provides an unbiased estimate of model performance.

## Model Details

We present a sequence-only baseline model that predicts sgRNA $\log_2$ fold change from a one-hot encoded 23-nt guide RNA sequence. The model takes as input a $4 \times 23$ tensor $\mathbf{X}$ and applies a single 1D convolution, nonlinearity, spatial pooling, and a linear output layer to produce a scalar prediction.

*Baseline CNN (sequence-only)*:

Input $\mathbf{X} \in \mathbb{R}^{4 \times 23}$ one-hot guide; Conv1D $\rightarrow$ ReLU $\rightarrow$ AdaptiveMaxPool.

Linear head predicts scalar $\log_2$ fold change; trained with MSE loss.

*Model 2 (sequence + metadata embeddings)*:

Same Conv1D encoder for sequence features as the baseline.

Adds learned embeddings for cell line, phenotype/condition, and chromosome; concatenates with sequence features.

Linear head on fused vector (MSE) captures experiment-specific shifts in effect size.

*Model 3 (deep Conv2D + multi-scale pooling + MLP fusion)*:

Reshapes to $(B, 1, 4, 23)$ and uses stacked Conv2D blocks (BatchNorm + Dropout) for richer motif learning.

Multi-scale sequence summary via two branches ($1 \times 1$ and $3 \times 3$) and global average pooling; concatenate.

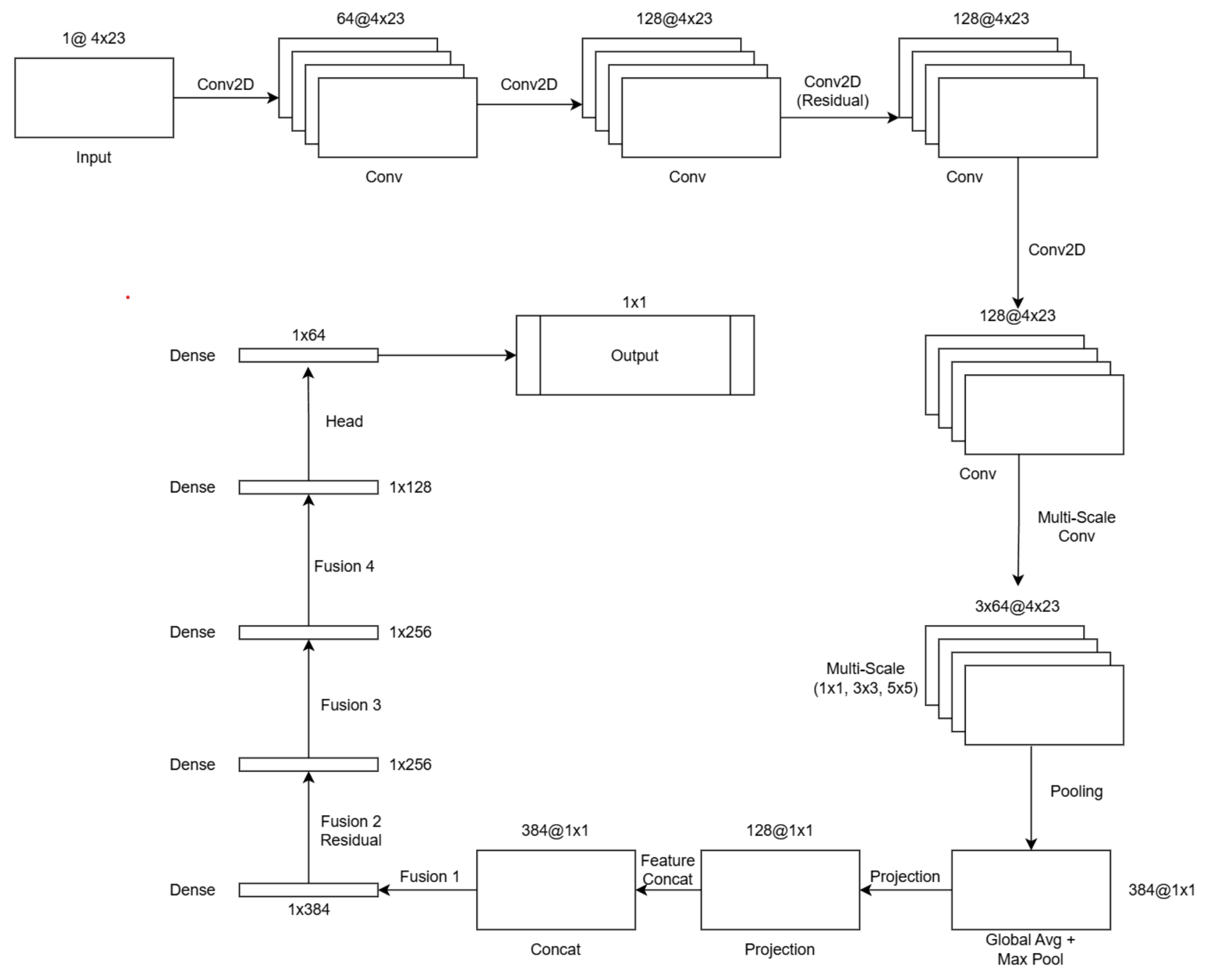
Adds strand embedding and replaces the single linear head with an MLP for nonlinear sequence and metadata patterns.

*Model 4 (residual Conv2D + richer pooling + explicit interactions)*:

Builds on Model 3 with a residual Conv2D block to stabilize deeper feature learning.

Three branches ($1 \times 1$, $3 \times 3$, $5 \times 5$) with both global avg & max pooling for stronger sequence descriptors.

Adds explicit metadata interaction features and a deeper fusion MLP with a skip connection.



## Experiments & Results

Starting from a simple CNN baseline, we designed a controlled set of experiments to isolate the impact of (i) adding experimental context metadata, (ii) increasing sequence modeling capacity, and (iii) explicitly modeling interactions between metadata features.

All models were trained to regress the $\log_2$ fold change using mean-squared error (MSE). Across experiments, performance improved steadily from Model 1 to Model 4. We found that Model 2 outperformed the sequence-only baseline, indicating that cell line and phenotype context explains systematic shifts in editing efficiency measurements. Additionally, moving from Conv1D (Models 1–2) to deep Conv2D with normalization and dropout (Model 3) improved generalization by capturing higher order motif interactions in the $4 \times 23$ representation. Model 4 achieved the best overall results by combining (i) multiple receptive fields ($1 \times 1$, $3 \times 3$, $5 \times 5$), (ii) complementary pooling statistics (avg+max), and (iii) explicit interaction terms between metadata embeddings. Showing that multi-scale and increased interactions helped the model the most.

These results suggest sgRNA efficiency is not determined by sequence motifs alone: experimental context (cell line, phenotype/condition, chromosome, strand) and motif patterns at multiple scales all influence the observed $\log_2$ fold change.

## Conclusion

We found that our model was able to predict much more accurately when the sequence features are modeled at multiple motif scales and combined with experimental context metadata.