preliminary result 1: crispAI on PRIDICT 90k dataset

To start testing the viability of crispAI model on the prime editing dataset, the moderately sized PRIDICT 90k dataset was modified to fit the input format of the crispAI model. At the first stage, the model itself underwent very little modification, only taking the sgRNA portion of the prime editors as input.

The crispAI model requires the 73bp long flank sequence of the target site, which would be used to calculate the moving window GC content, NuPoP affinity and occupancy, as well as BDM score[1]. However, the wide target sequence in the PRIDICT 90k dataset was only 99bp long. As a result, a pipeline was developed to map each target site to the human genome reference sequence coordinates(hg38), and extract the 73bp flank sequence from the reference genome using the toolkit provided by crispAI authors.

PRIDICT provided the HGVS(Human Genome Variation Society) notation of the intended edits, as well as the direction of the target strand(forward or reverse, in relation to the reference genome). Using the given information, GeneBe was able to map the target site to the hg38 coordinates, and given the relative location of the edit to the SpCas9 target site, the precise 23 bp target site could be calculated[2]. The flank sequence is then generated using genomepy, a Python package that acquires the reference genome sequence given the coordinates.

However, around 1,200 out of 92,000 target sites were unable to be mapped to the reference genome, mostly due to GeneBe unable to recognize the HGVS notations. These target sites were removed from the datasets for the time being.

At the same time, a large proportion of the coordinates entered were around 1-3bp off from the actual target site returned by genomepy. To adjust for this, the wide target sequence returned by genomepy was in turn used to pinpoint the exact site. With a 73bp flank sequence around the target site, the relative location of the target site should be at the 74rd bp of the wide target sequence. Using this fact, the coordinates are adjusted to ensure the perfect match between the hg38 coordinates and the SpCas9 target site.

The coordinates are then parsed using a helper function provided by crispAI authors, producing the physical features matrices required by the model. Along with the sgRNA sequences, the input data is complete and ready for training.

Unfortunately, the same process could not be repeated for the DeepPrime dataset,

since the wide target is also not long enough, and no additional information was available to map the target site to the reference genome.

The Skorch library was used to wrap the crispAI model, and the training was conducted on a single Nvidia RTX 3050/3080 GPU. The custom criterion function provided by crispAI authors was used to calculate the loss, which evaluates the log likelihood of the observed efficiency given the ZINB distribution set by the three parameters produced by the model.

1 crispAI with no modification

The starting point of the experiment is using the base crispAI model, considering only the SpCas9 nicking stage of the prime editing process. The result doesn't seem compelling when compared to conventional ML models and the pridict model, with significantly lower pearson and spearman correlation(). However, when comparing with DeepSpCas9, a model that also only focuses on the SpCas9 nicking stage, the result is far superior(pearson's r of 0.2 vs 0.45).

2 crispAI with extended input sequence

To accommodate the additional PBS and RTT section of the prime editors, the input sequence was extended from the original 23 bp long spanning across the sgRNA and the PAM sequence, to 60bp, covering the entire target site. This was accomplished by updating the data annotation function to move the end coordinate 59bp downstream from the start coordinate, instead of 22bp.

2.1 crispAI with extended input sequence and functional annotations

Since we are working with on-target datasets, the last two columns of the input data used by crispAI for indicating mismatch direction is unusedI(because there is no mismatch). To utilize the extra space, the PBS and RTT annotation sequences were added to the input data. The functional annotations were binary sequences of 1s and 0s, indicating the presence of the corresponding nucleotide at the given position.

References

[1] Florian Störtz, Jeffrey K. Mak, and Peter Minary. "piCRISPR: Physically Informed Deep Learning Models for CRISPR/Cas9 off-Target Cleavage Prediction". In: *Artificial Intelligence in the Life Sciences* 3 (Dec. 2023), p. 100075. ISSN: 26673185. DOI: 10.1016/j.ailsci.2023.100075. URL: https://linkinghub.elsevier.com/retrieve/pii/S2667318523000193 (visited on 10/14/2024).

[2] Piotr Stawiński and Rafał Płoski. "Genebe.Net: Implementation and Validation of an Automatic ACMG Variant Pathogenicity Criteria Assignment". In: *Clinical Genetics* 106.2 (2024), pp. 119–126. ISSN: 1399-0004. DOI: 10.1111/cge.14516. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/cge.14516 (visited on 11/14/2024).