Slide1:

Genome dataset is large and complicated, so understanding all functions and interactions of genome is very challenging. In this project, we use deep learning approaches (specifically a bi-directional LSTM) to model genome data and extract highly complex patterns inside that. To do so, we introduce a novel learning algorithm using the bidirectional LSTM model to generate fixed-length vector representations of the reference sequence; then we conduct a case study on real DNA sequencing dataset, and compute the perplexity, accuracy and sensitivity of the LSTM-based sequence modeling. Then we perform different clustering algorithms on the obtained models to cluster genome dataset. We use the human reference genome as our training dataset, in addition to a set of short reads generated using Illumina sequencing technology. We first decompose the reference genome into multiple sequences. These sequences are then fed into the bidirectional LSTM model and then mapped into fixed-length vectors. The obtained vectors are then used to cluster and group similar sequences into same clusters. We then evaluate the accuracy of our clustering algorithms.

Slide 2:

Modeling genome dataset using deep learning has different applications, one of them is genome clustering, which is used for this project. The main goal of genome clustering is to detect group of sequences in the genome that are close together (in the sense of pairwise distance and far from other sequences. The main reason of clustering genome using deep learning model is that traditional genome clustering methods are extremely computationally- intensive which make them impractical for real applications. The picture shows one of the common traditional genome clustering methods which is called MicroScope. As the picture shows, at the first step the genome is divided to sketches which are set of sequences of the same length, then pairwise comparisons are done on each pair of sequences (which is a bottleneck step and very time-consuming). Once the pairwise distances are computed, some filtering methods are used to filter any redundant distances. This process has to be done for each set of query sequences, however in this project we are modeling the genome dataset using BI-LSTM once, and use the model multiple times for any list of query sequences.

Slide 3:
More specifically, in this project we model the genome dataset using Bi-LSTM. We use the following approach to train the model:

Step 1 (Constructing Words): Each word is made of w characters, so the reference genome that includes a set of characters is split into unique words of size w characters (where w represents the word size). Hence, for the reference genome of size |G|, the number of unique words is at most |G| w , and for the alphabet size of Σ, the max number of unique words is bounded by |Σ| w.

Step 2 (Constructing Dictionary): Dictionary is defined as a collection of unique words. To construct a dictionary, each unique word, generated at the previous step, is added to the dictionary. The dictionary size is bounded by |Σ| w.

Step 3 (Constructing Batch): Each batch is a collection of fixed-length words (words of length w). Indeed, the reference genome is split into multiple batches where each batch consists of multiple words.

Step 4 (Constructing Epoch (Iteration)): Epoch is a collection of all batches. Each epoch runs over the whole reference genome. Hence, epoch size equals to |G|. Indeed, epoch consists of |G| B batches. LSTM network runs batch by batch sequentially. The final updated weights are what is called the training model.

The model (vector representations) is then used for different clustering algorithms (i.e., DBSCAN, Kmeans, GMM).

Slide 4:
Other applications of genome modeling are as follows (which are not used for this project):

1. Connecting genotype to phenotype
2. Predicting regulatory function
3. Classifying mutation types