1 Objective

The goal of Project 2 is to train a classifier capable of predicting enhancer sequences based on DNA sequence data using the frequency of k-mers.

2 Biological Context

Enhancers are non-coding DNA sequences that can influence gene expression when in close 3D proximity to promoters, even if located up to 1Mbp away. They play a key role in gene regulation, contributing to genetic differences between species and individuals by affecting non-coding DNA regions rather than gene coding sequences.

Studying enhancers is vital for understanding gene regulation, disease physiology, and genetic diversity. Enhancers contain Transcription Factor Binding Sites (TFBS), where transcription factors bind to recruit RNA Polymerase II and mediate enhancer-promoter interactions essential for transcriptional activation.

3 Methods

For best performance, we used the following methods in our project.

3.1 Data Preprocessing

The original data contained long sequences of ACTG characters. We split the sequences into k-mers of length 4,5,6, and then counted the frequency of each k-mer in each sequence. Lastly we divided the frequency of each k-mer by its length. This approach allows for analysis of data of different lengths and it condenses the data into a more manageable form.

3.2 Classifier

For classification, we used the support vector machine classifier (SVC from sklearn). The SVC was picked due to its simplicity and efficiency.

4 Results

To measure the quality of classifiers we used the 10-fold corss-validation method. We used two datasets. The first one being a set of labeled sequences labeled positive or negative taken from the VISTA database. The second dataset used the same positive data as the first one but used random sequences from the human genome as negative data. The results are detailed in this section. The confusion matrices are show in the appendix.

4.1 4-mers

Using the 4-mers as features, we obtained the following results:

Metric	Value for first dataset	Value for the second dataset
AUC-ROC	0.6459	0.9918
Accuracy	0.6083	0.9637
Precision	0.6062	0.9799
Recall	0.7630	0.9517
F1-Score	0.6756	0.9656

Table 1: Performance Metrics for 4-mers.

4.2 5-mers

Using the 5-mers as features, we obtained the following results:

Metric	Value for first dataset	Value for the second dataset
AUC-ROC	0.6662	0.9915
Accuracy	0.6218	0.9603
Precision	0.6243	0.9729
Recall	0.7348	0.9522
F1-Score	0.6751	0.9624

Table 2: Performance Metrics for 5-mers.

4.3 6-mers

Using the 6-mers as features, we obtained the following results:

Metric	Value for first dataset	Value for the second dataset
AUC-ROC	0.6752	0.9896
Accuracy	0.6288	0.9519
Precision	0.6351	0.9584
Recall	0.7183	0.9513
F1-Score	0.6741	0.9548

Table 3: Performance Metrics for 6-mers.

4.4 Comparison

The results clearly show two things. First the larger the k-mer the better the results. Although it gives better results it also increases the processing time as the number of features grows exponentially with the k growing. The calculation for 6-mers took 10 times longer than for 4-mers.

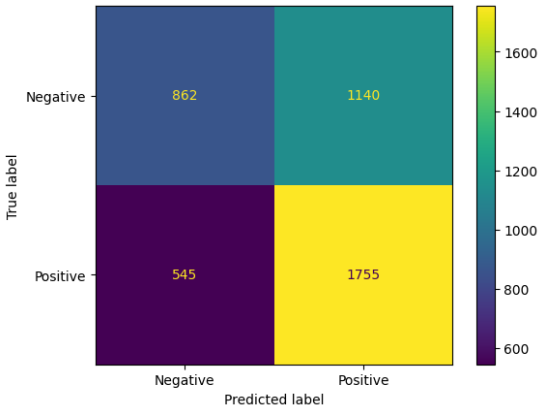

Figure 1: Results for 4-mers for first dataset.

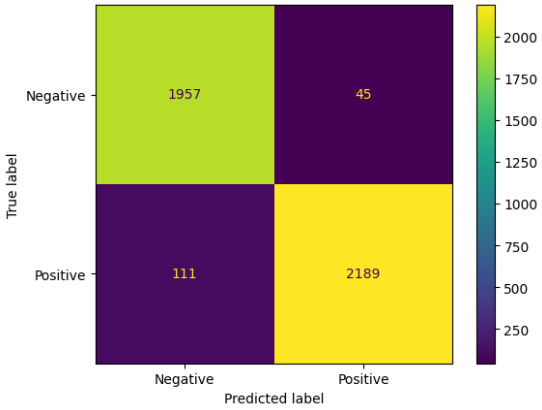

Figure 2: Results for 4-mers for second dataset.

The second thing the results show it that the second dataset gives much better results than the first one. The accuracy for the second dataset is around 30% points higher. This could not be explained due to the quality of the data. The most likely explanation is that the model realized that the negative data from the second dataset came from a different source (human DNA).

Appendix

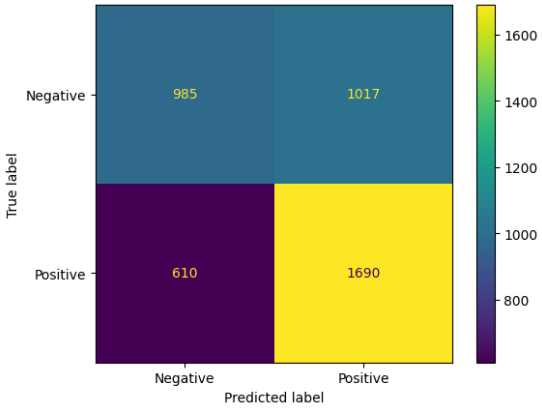

Figure 3: Results for 5-mers for first dataset.

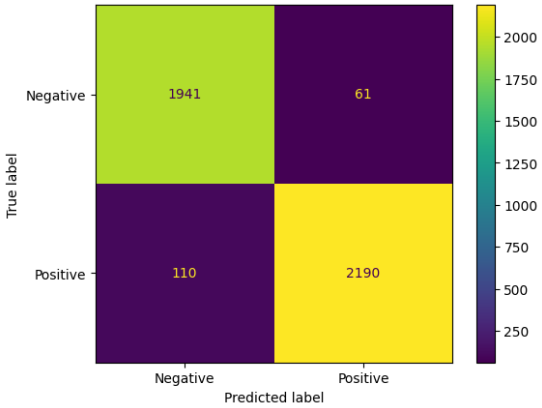

Figure 4: Results for 5-mers for second dataset.

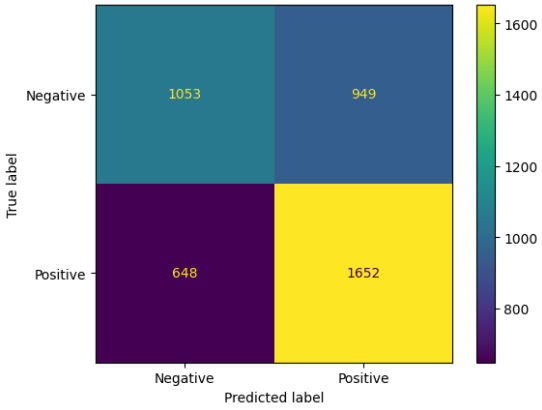

Figure 5: Results for 6-mers for first dataset.

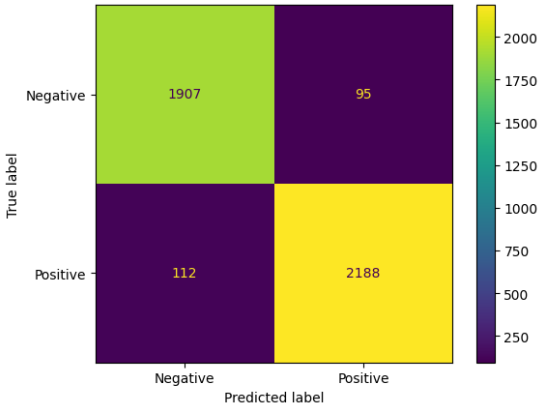

Figure 6: Results for 6-mers for second dataset.