Neural Network Taxon Prediction

Background

- Task: Predict microbial lineage using DNA sequences and k-mer features. K-mer-based feature extraction is a widely used technique for microbial taxonomy classification
- Our Approach
    - Uses a Sequential Neural Network Model with k-mer values (3 to 5).
- Other Approach
    - Deep Neural Networks (DNNs) and Convolutional Neural Networks (CNNs) offer high accuracy but face computational and scalability challenges. (Park H, Lim SJ, Cosme J, et al. Investigation of machine learning algorithms for taxonomic classification of marine metagenomes. *Microbiol Spectr*. 2023;11(5):e0523722. doi:10.1128/spectrum.05237-22)

Data

- Filtered data to only data containing a full taxonomic labels down to species level = 2.21GB. Original(5.2GB)
- Randomly sampled 25% of the data set for the neural network to be trained on.
- Different ASV Length:
  - Min: 90 bases
  - Max: 564 bases

Why sample 25% data

- Issue is the nature of the data. Where 0 = A; 1= T; 2 =C; 3 =G
- As the code convert ACTG into kmers it takes exponentially more data each time. Due to the different possible combination of ACTG
- K = 3
  - 4(different types of nucleotides) ^3 =64 features
- K = 4
  - 4(different types of nucleotides) ^4 =256 features
- K = 5
  - 4(different types of nucleotides) ^5 = 1024 features

Hardware limitation

- Each time K increases the amount of ram required to store the feature increase by factor of 4
- K = 3 ; 1GB RAM
- K = 4 ; 4 GB RAM
- K = 5; 16 GB RAM
- ….
- K = 11; 65,536 GB RAM = 64 TiB RAM

Method

- Using a sequential neural network to predict taxonomic levels down to species level using different values of Kmers to see which value gives the best results.
- Neural Net:
  - Input layer: shape of the data(90-564)
  - First Hidden layer: K ^4 - Relu activation function
  - Dropout layer
  - Second hidden layer:(K^4)// 2 - Relu activation function
  - Dropout layer
  - Third hidden layer: (K^4)// 4 - Relu activation function
  - Dropout layer
  - Output layer: number of distinct classes

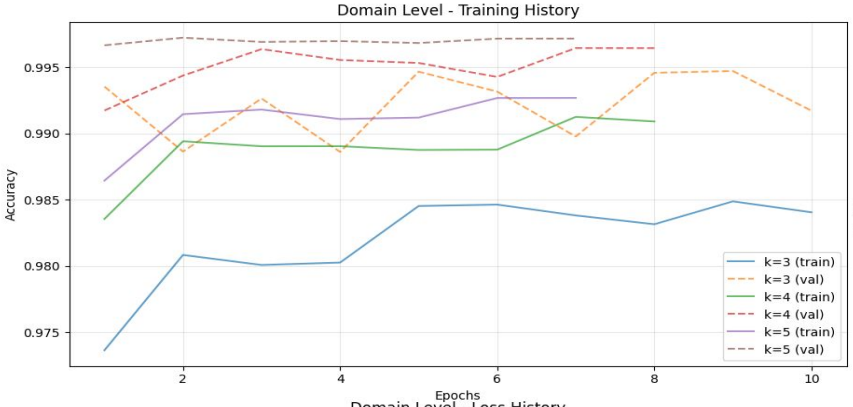
Neural Network

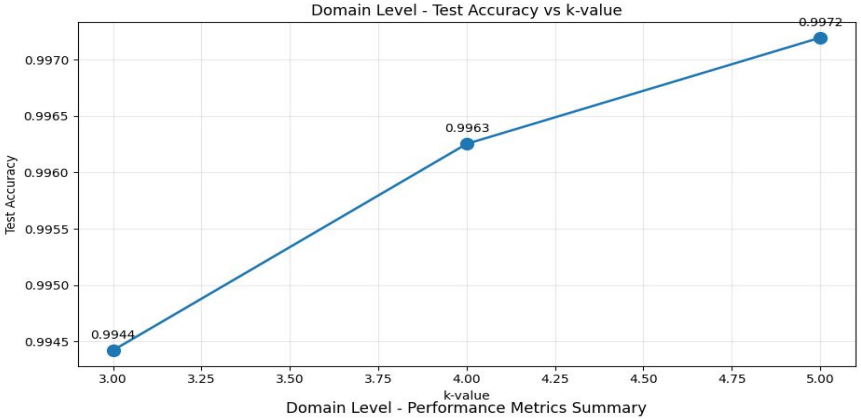
- Stop the model if improvements plateaus after 10 epoch
- Stop model from adjusting the weights to much for more stable training
- Use sparse categorical cross entropy loss metrics for taxonomic label classification
  - Reason: multiple taxonomic labels to predict right
  - Metric ranges: 0 (very good) -> infinity (very bad)
  - Goal: sub 1.0 loss value
- Backtracking:
  - Model use previous best weights when the model is overfitting the data
- Due to the constraints mentioned before the model only tested k values from 3 - 5

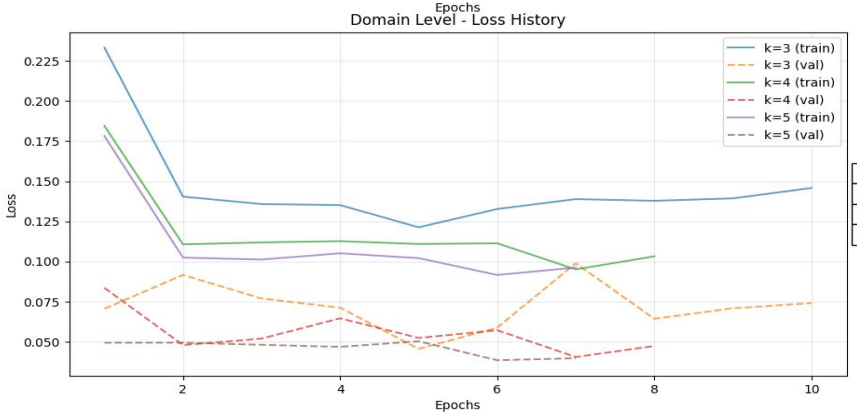
Neural Network

- Number of Models trained: 21
- 3 different K value
- 7 different taxonomic layers
- Use 80:20 split to receive the following results

Domain Level Results



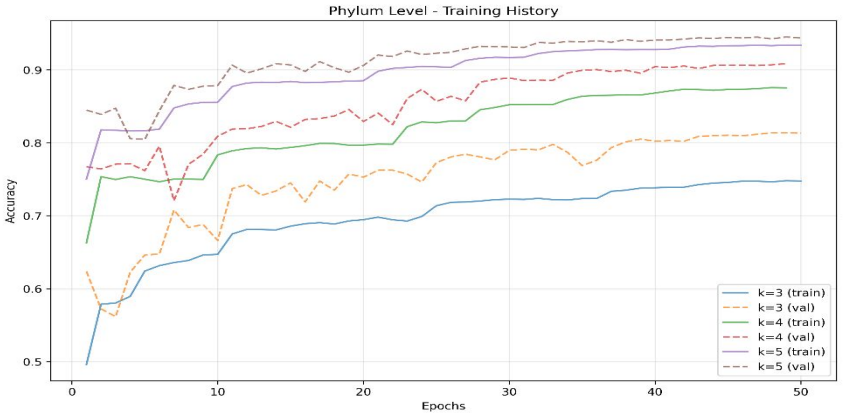


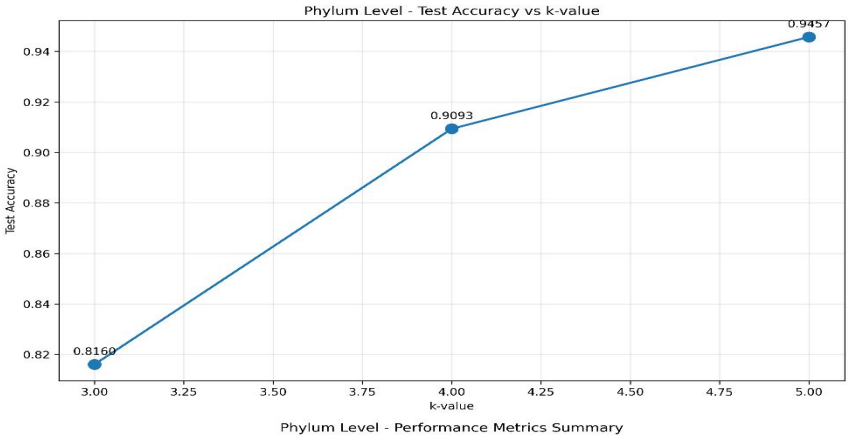


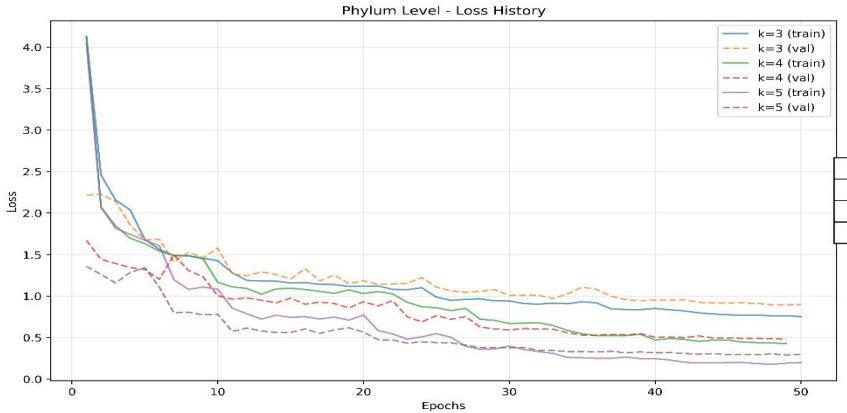
Domain Level - Performance Metrics Summary

k-value	Test Acc	Test Loss	Classes	Features
3	0.9944	0.0471	2	64
4	0.9963	0.0533	2	256
5	0.9972	0.0505	2	1024

Phylum Level Results



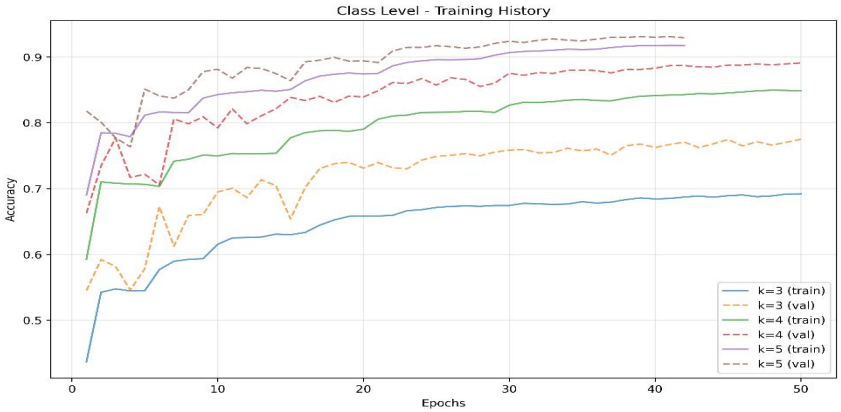


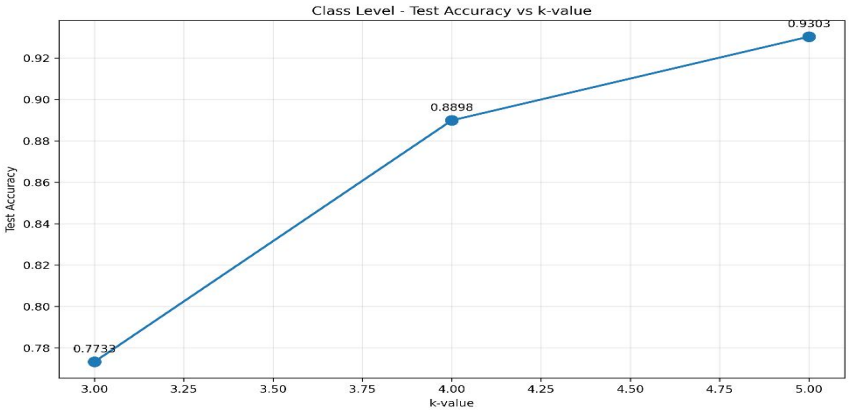


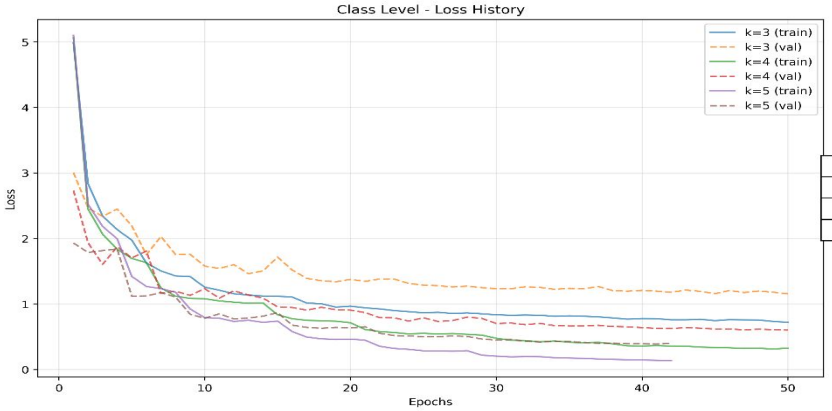
Phylum Level - Performance Metrics Summary

k-value	Test Acc	Test Loss	Classes	Features
3	0.8160	0.8856	117	64
4	0.9093	0.4726	117	256
5	0.9457	0.2832	117	1024

Class Level Results



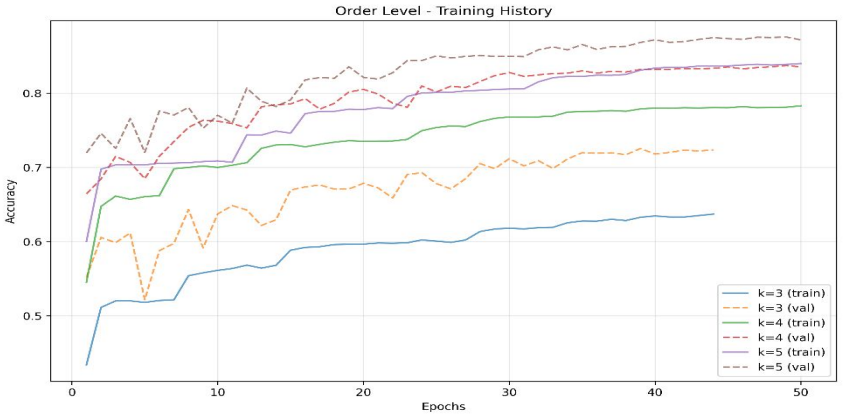


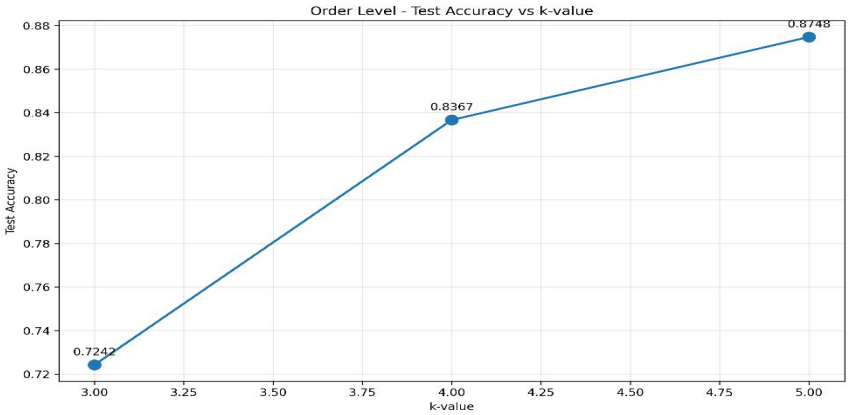


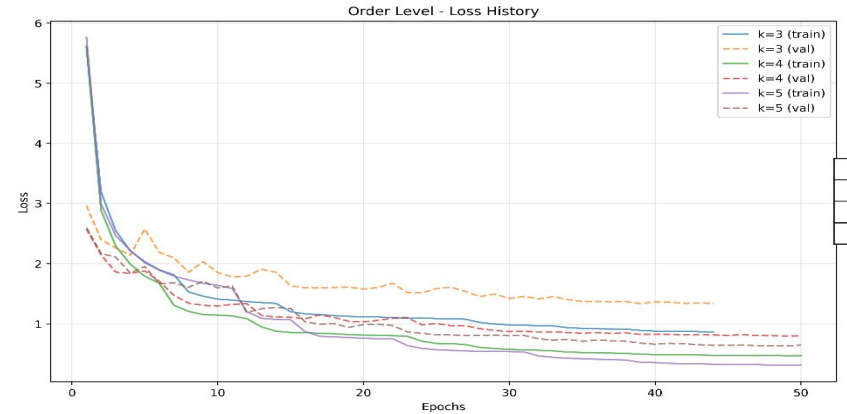
Class Level - Performance Metrics Summary

k-value	Test Acc	Test Loss	Classes	Features
3	0.7733	1.1579	284	64
4	0.8898	0.6013	284	256
5	0.9303	0.3930	284	1024

Order Level Results



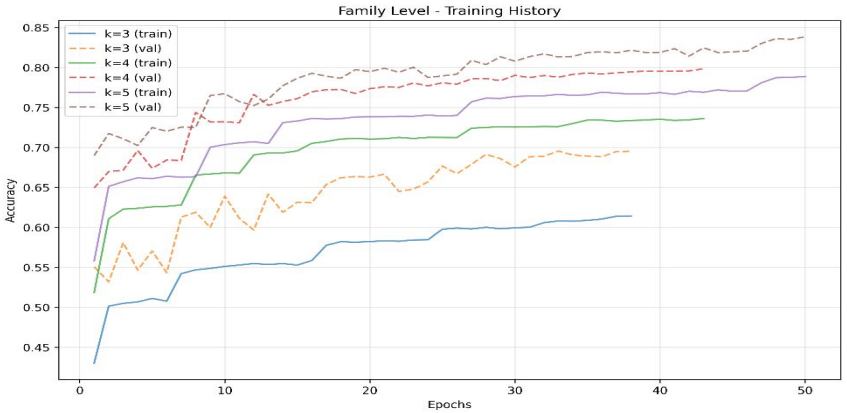


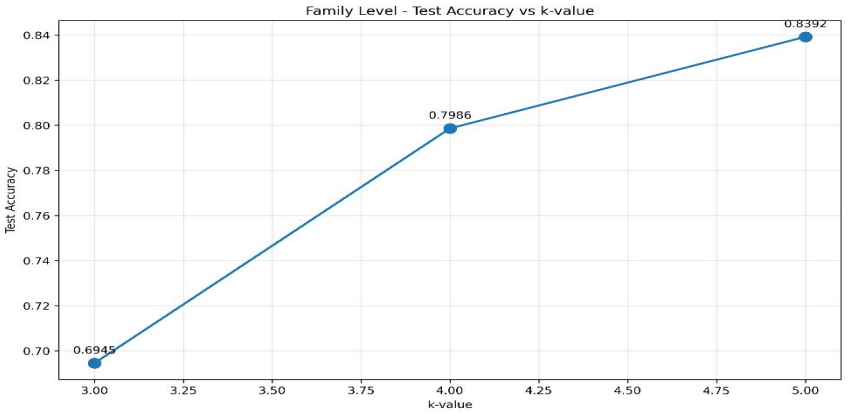


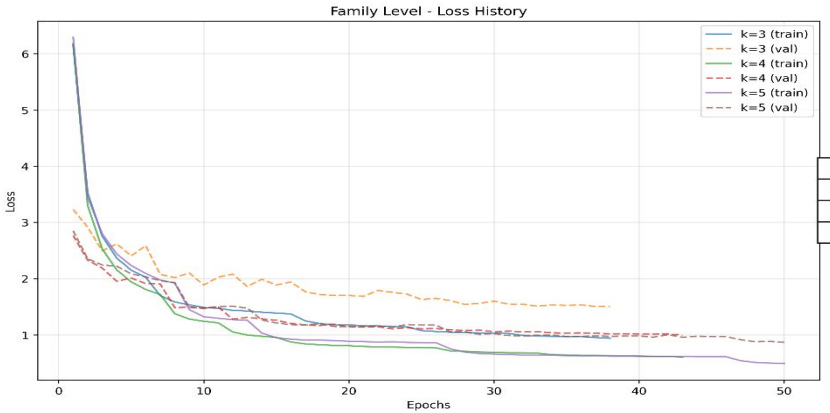
Order Level - Performance Metrics Summary

k-value	Test Acc	Test Loss	Classes	Features
3	0.7242	1.3430	791	64
4	0.8367	0.8007	791	256
5	0.8748	0.6398	791	1024

Family Level Results



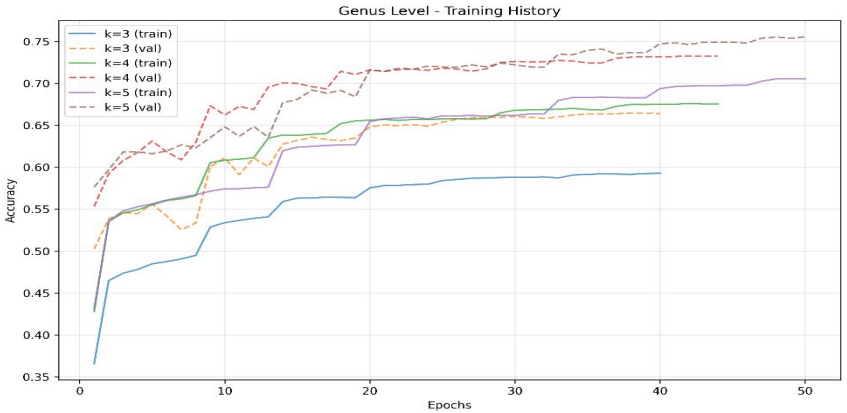


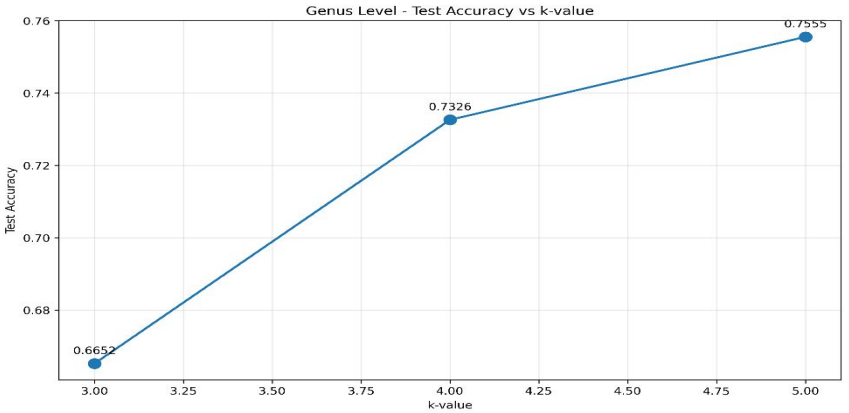


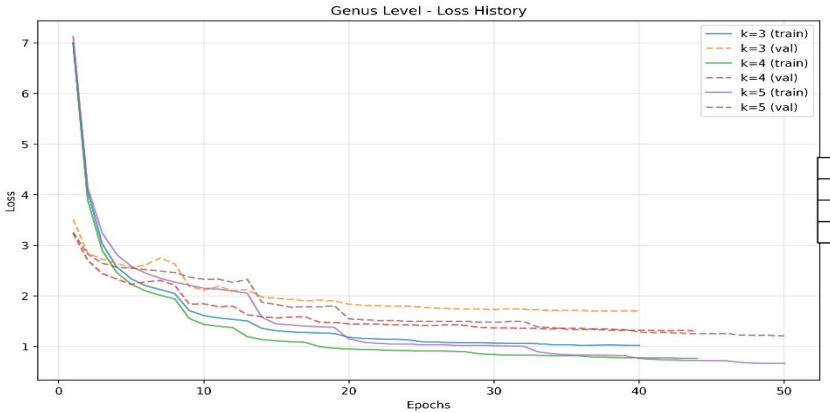
Family Level - Performance Metrics Summary

k-value	Test Acc	Test Loss	Classes	Features
3	0.6945	1.5119	1674	64
4	0.7986	0.9965	1674	256
5	0.8392	0.8604	1674	1024

Genus Level Results



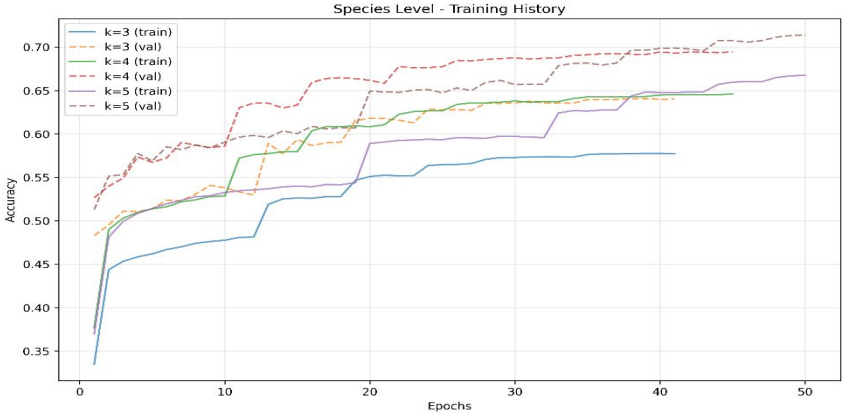


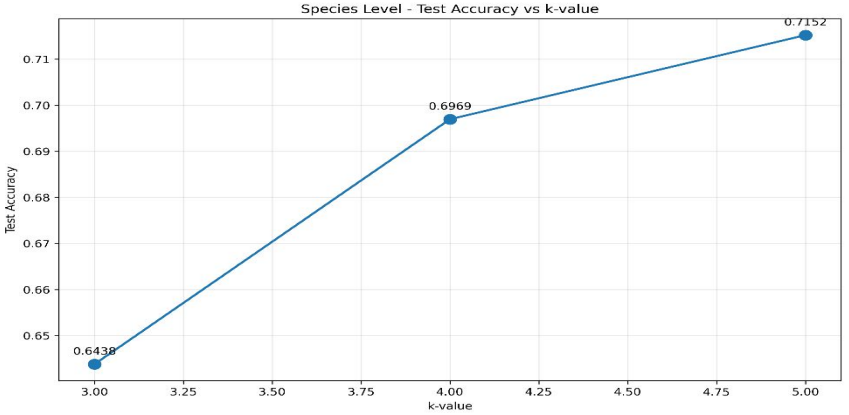


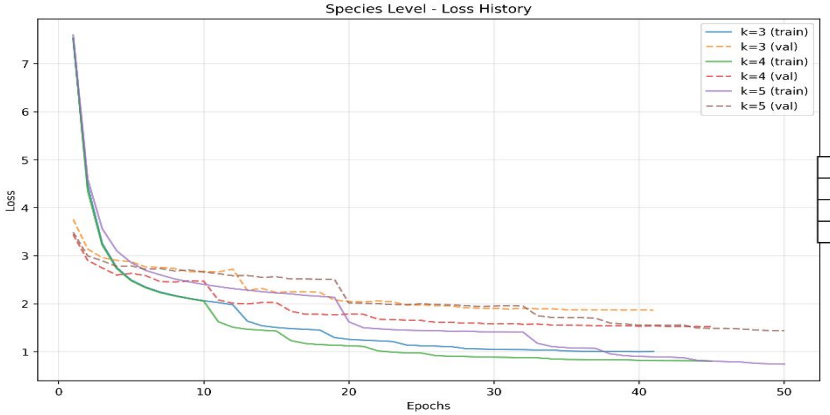
Genus Level - Performance Metrics Summary

k-value	Test Acc	Test Loss	Classes	Features
3	0.6652	1.6954	5178	64
4	0.7326	1.3054	5178	256
5	0.7555	1.2097	5178	1024

Species Level Results







Species Level - Performance Metrics Summary

k-value	Test Acc	Test Loss	Classes	Features
3	0.6438	1.8451	8864	64
4	0.6969	1.5128	8864	256
5	0.7152	1.4235	8864	1024

Challenges of Using Needleman-Wunsch Alignment

- Needleman-Wunsch
  - a baseline method for finding the closest matches to query sequences.
  - Comparing ~8,000,000 references against just 5% of data meant ~700 billion pairwise comparisons
    - Iterative approach:
      - Sequential computation of alignments for each query against all references.
      - **Issue:** Even with a single GPU, processing would take **years** to complete.
    - Multiprocessing on CPU:
      - **Issue:** CPU threads were slower compared to GPU, and resource utilization remained suboptimal for this dataset size.
    - Parallelization with GPU:
      - **Issue:** Memory constraints limited batch sizes, and the total time was still estimated at **months**.

Future Work

- Improve the neural network model:
  - Change value representation of ACTG into smaller data sizes such as using binary bits. Since we only need to represent values from (0-3)
  - Different method to store the values of Kmers
  - Other types of neural network model
- Use 100% of the data

References

- Hongyuan Zhao, Suyi Zhang, Hui Qin, Xiaogang Liu, Dongna Ma, Xiao Han, Jian Mao, Shuangping Liu, DSNetax: a deep learning species annotation method based on a deep-shallow parallel framework, *Briefings in Bioinformatics*, Volume 25, Issue 3, May 2024, bbae157, https://doi.org/10.1093/bib/bbae157

- Asgari E, Garakani K, McHardy AC, Mofrad MRK. MicroPheno: predicting environments and host phenotypes from 16S rRNA gene sequencing using a k-mer based representation of shallow sub-samples [published correction appears in Bioinformatics. 2019 Mar 15;35(6):1082. doi: 10.1093/bioinformatics/bty652]. *Bioinformatics*. 2018;34(13):i32-i42. doi:10.1093/bioinformatics/bty296

- McDonald, D., Jiang, Y., Balaban, M. et al. Greengenes2 unifies microbial data in a single reference tree. Nat Biotechnol 42, 715–718 (2024). https://doi.org/10.1038/s41587-023-01845-1

- GreenGenes database release (2024.09). Retrieved from: https://ftp.microbio.me/greengenes_release/2024.09/00README