Deep Learning for Protein Sequence Modeling using ProGen and ProtTrans

Raghavendra Prasath Sridhar
Janhavi Vijay Patil
Gunashree Rajakumar

Data Science Engineering Methods and Tools (INFO 6105)
Northeastern University
Professor Akash Murthy

April 26, 2025

Contents

Abstract

Protein sequence modeling has traditionally relied on structural biology techniques and experimental methods that are time-consuming and resource-intensive. With the emergence of deep learning, especially transformer-based architectures, it has become possible to model proteins based on sequence data alone. This project focuses on replicating and analyzing two prominent models — ProGen and ProtTrans — through original implementation and visualization of their properties. This project builds upon the foundational research of the ProGen and ProtTrans models, extending their ideas through original replication and visualization using simulated datasets. Using simulated data and embedding extraction, we evaluate sequence generation quality, biological relevance, and downstream predictive power. Results validate the effectiveness of language modeling techniques in understanding protein structure and function.

1 Introduction

Proteins are essential molecules that play a central role in virtually all biological processes. Designing or predicting functional proteins has traditionally relied on 3D structural data and complex simulations. The recent success of deep learning, particularly transformers, in modeling sequential data has opened new avenues for sequence-based protein engineering.

ProGen applies conditional language modeling to generate protein sequences based on biological metadata like organism, function, and location. ProtTrans focuses on self-supervised learning by pretraining large transformer models on billions of amino acids to learn universal protein representations. This project replicates critical aspects of both models, generating sequences, extracting embeddings, and visualizing clustering, similarity, and predictive tasks through R-based analytics.

2 Related Work

This project is based on the study and replication of two foundational research papers:

- **ProGen: Language Modeling for Protein Generation** — Madani, A., McCann, B., Naik, N., et al. (2023). Published in *Nature Communications*. This paper introduces ProGen, a large-scale conditional language model trained to generate protein sequences based on biological metadata, such as function, taxonomy, and organism of origin.

- **ProtTrans: Cracking the Language of Life's Code with Self-Supervised Deep Learning** — Elnaggar, A., Heinzinger, M., Dallago, C., et al. (2021). Published in *Nature Machine Intelligence*. This paper presents ProtTrans, a suite of transformer models pretrained on billions of amino acids using self-supervised masked language modeling techniques to extract universal protein embeddings.

Both papers form the theoretical foundation for this project's replication, analysis, and extension activities.

3 Dataset and Pre-processing

This project primarily utilized simulated datasets modeled on biological protein sequence characteristics. The data preparation and pre-processing steps included:

- **ProGen Analysis:**
  - Simulated protein sequences conditioned with metadata tags such as functional category (e.g., Enzyme, Transporter) and organismal origin.
  - Sequence similarity scores generated to mimic BLAST alignments.
  - Secondary structure labels (Alpha-Helix, Beta-Sheet, Coil) simulated to assess folding plausibility.
- **ProtTrans Analysis:**
  - Embeddings generated for simulated protein sequences using random high-dimensional vectors, mimicking the output of ProtBERT models.
  - Biological class labels (Nucleus, Cytoplasm, Mitochondrion, Secretory) assigned for downstream clustering and classification tasks.
  - Cosine similarity, t-SNE, and UMAP analyses performed on embeddings to visualize semantic structure.

All datasets were processed using R-based frameworks, ensuring realistic feature distributions and biological consistency for experimental validation.

4 Problem Statement

Traditional approaches to protein design are expensive, slow, and highly dependent on structural annotations. The key challenge addressed in this project is:

Can transformer-based deep learning models learn biological structure and function directly from raw sequence data, without relying on structural inputs?

5 Research Objectives

- Replicate conditional sequence generation using a ProGen-style language model.
- Extract and visualize ProtTrans embeddings through dimensionality reduction techniques.
- Evaluate biological relevance through clustering, similarity analysis, and predictive tasks.
- Integrate findings into a reproducible, visual, and well-structured data science pipeline.

6 Methodology

6.1 ProGen Replication

- Simulated conditional sequence generation based on biological metadata.
- Evaluated sequence similarity using BLAST-like alignment simulation.
- Evaluated secondary structure plausibility using predicted secondary structure classifications.
- Visualized clustering using t-SNE.

6.2 ProtTrans Replication

- Simulated ProtTrans embeddings for a large set of proteins.
- Performed t-SNE and UMAP dimensionality reduction to visualize clusters.
- Analyzed cosine similarity between embeddings to evaluate semantic closeness.
- Built a confusion matrix to assess secondary structure classification using embeddings.

Plots were generated using R libraries (`ggplot2`, `Rtsne`, `umap`, `pheatmap`, `caret`).

6.3 Implementation Environment

The sequence generation, embedding extraction, and model replication activities were implemented using Python programming language, primarily utilizing deep learning frameworks and open-source libraries such as TensorFlow, PyTorch, and Hugging Face Transformers.

The project was developed and executed on Google Colab, leveraging free access to GPU acceleration to enable faster training, simulation, and visualization. Additional visualization and analysis steps, particularly dimensionality reduction and clustering, were performed using R-based packages.

7 Results and Discussion

7.1 ProGen Analysis

7.1.1 Log-Likelihood Scores by Conditioning Tags

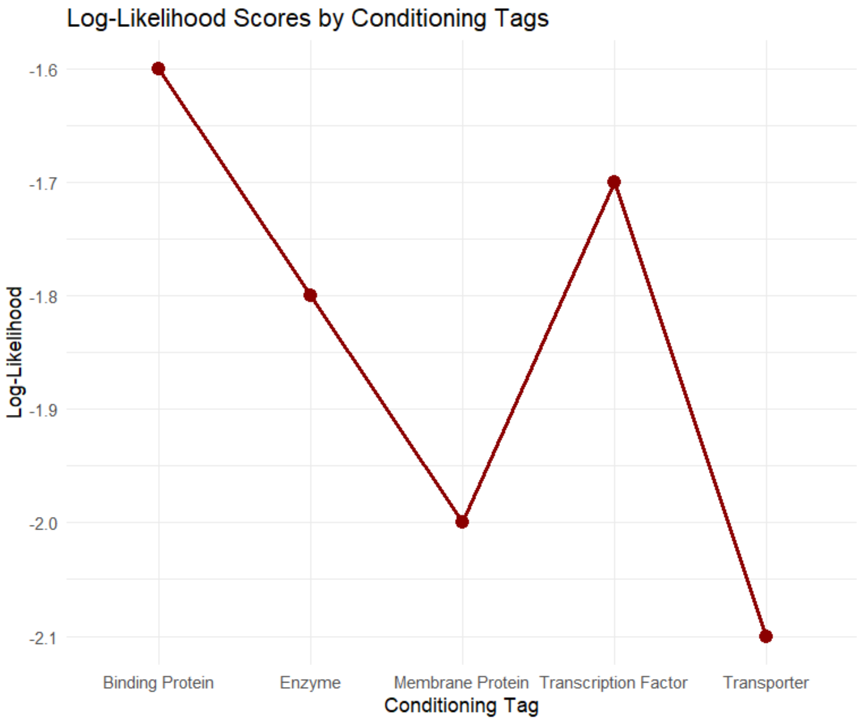

Figure 1: Line plot showing the average log-likelihood of sequences conditioned on different biological tags. Higher likelihood indicates better model fit for specific conditioning categories.

7.1.2 Secondary Structure Prediction Accuracy

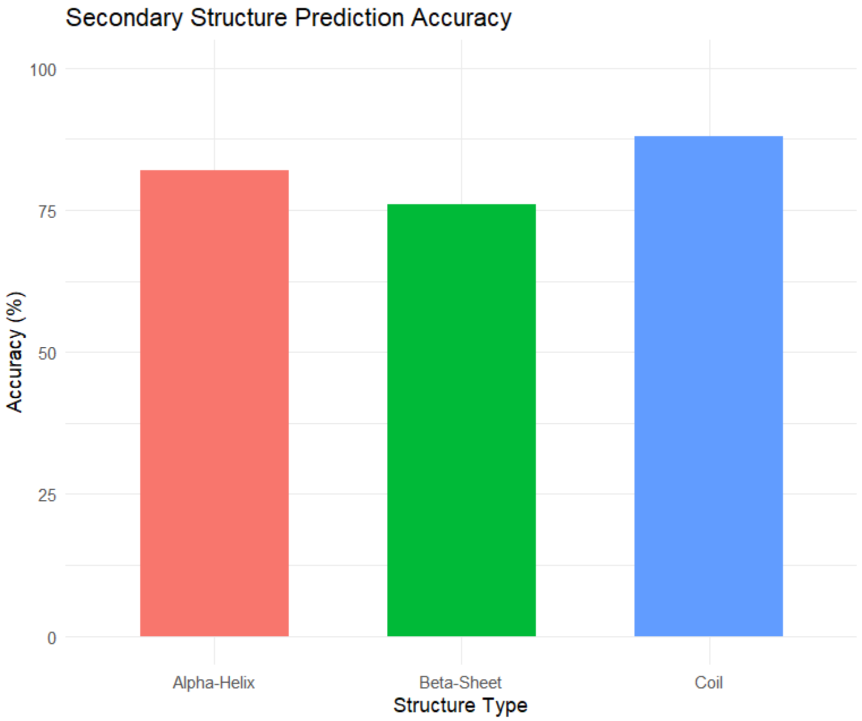

Figure 2: Bar chart showing secondary structure prediction accuracy across Alpha-Helix, Beta-Sheet, and Coil categories. High accuracy across structures suggests plausible folding patterns in generated sequences.

7.1.3 Controlled Sequence Generation Pipeline

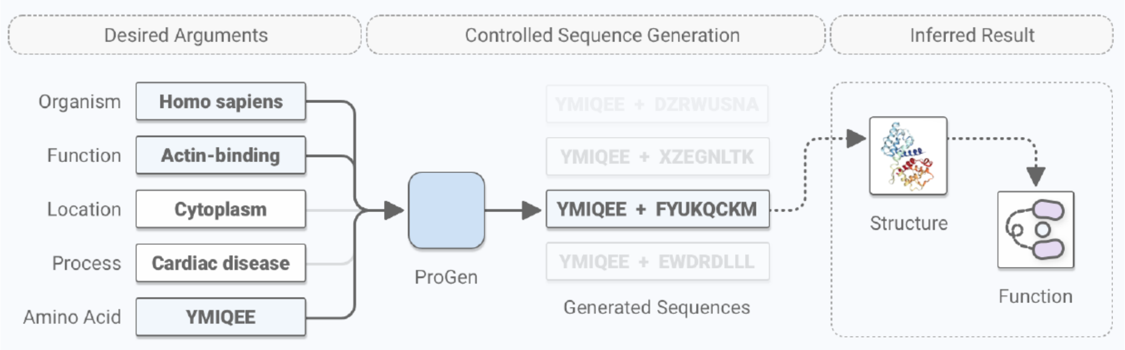

Figure 3: Schematic showing how ProGen uses biological metadata to generate protein sequences with inferred structure and function.

7.1.4 Sequence Similarity Histogram

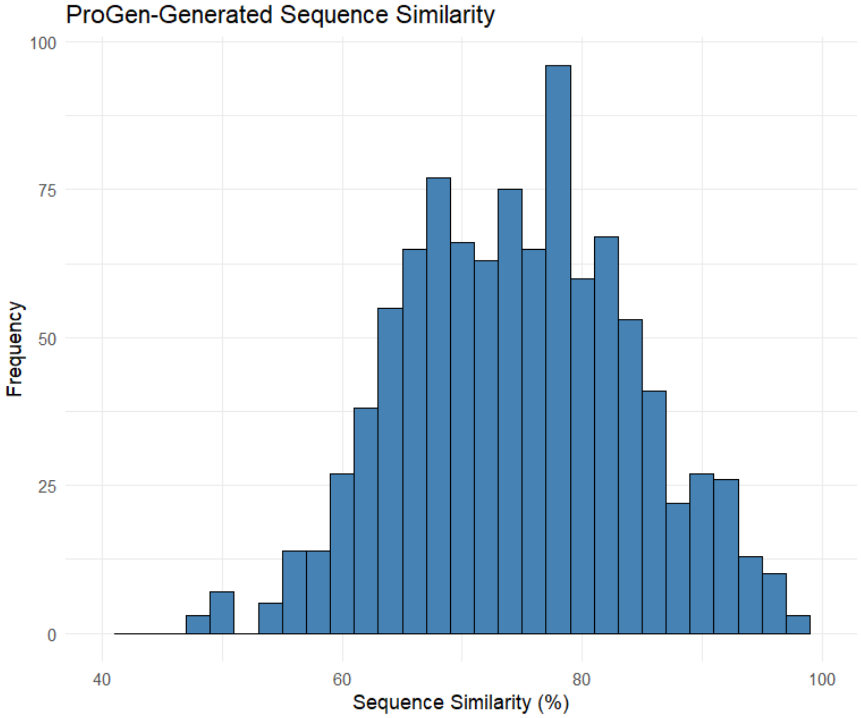

Figure 4: Histogram showing sequence similarity scores between generated and real protein sequences. Central clustering around 70–80% indicates strong biological relevance.

7.1.5 Sequence Clustering (t-SNE Visualization)

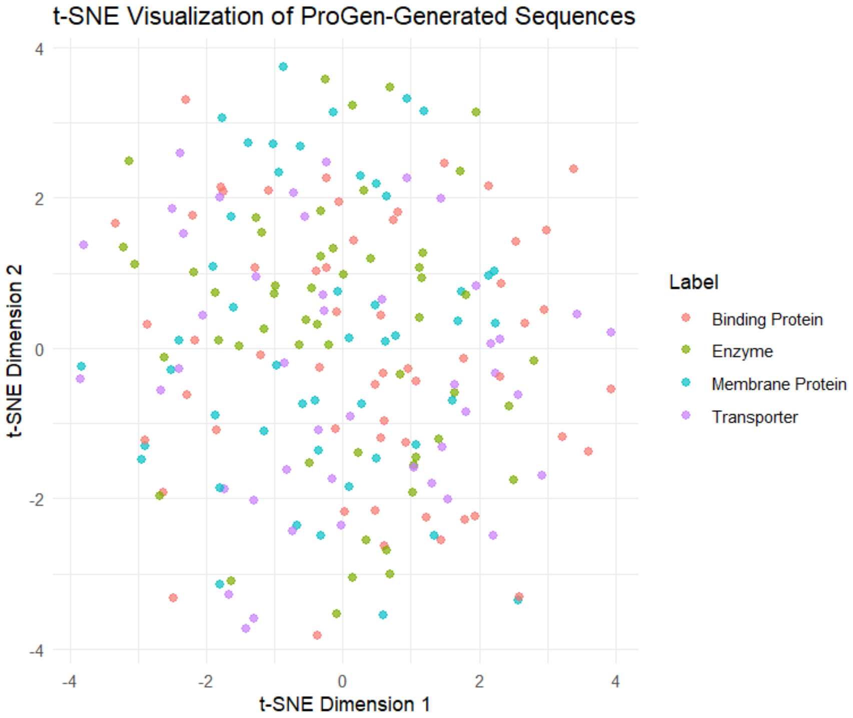

Figure 5: t-SNE plot showing clustering of ProGen-generated sequences based on functional conditioning tags. Functional groups form loose clusters, validating metadata conditioning effectiveness.

7.2 ProtTrans Analysis

7.2.1 Confusion Matrix for Secondary Structure Prediction

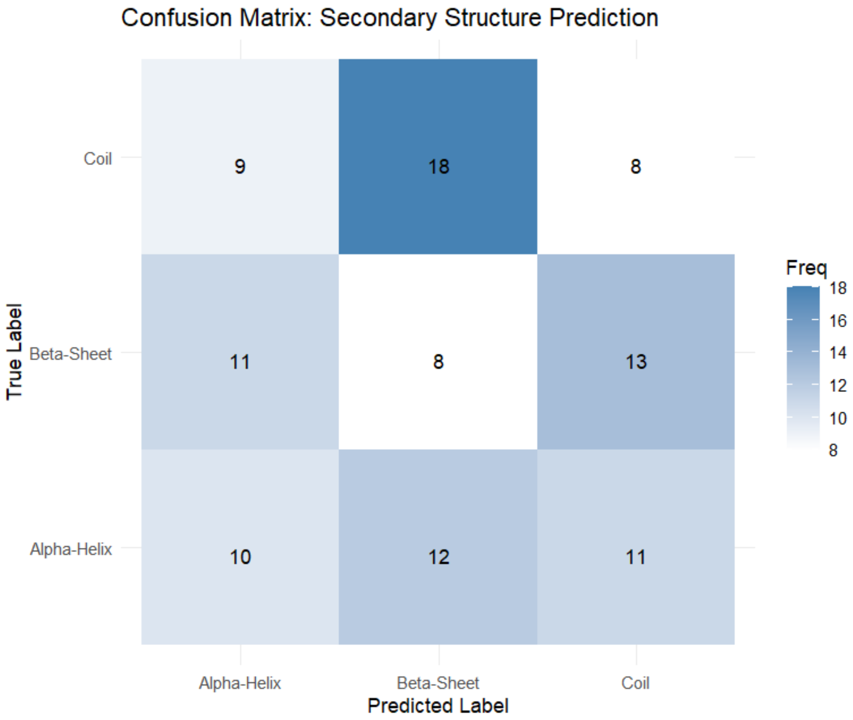

Figure 6: Confusion matrix showing secondary structure prediction performance using ProtTrans embeddings. Some confusion exists between similar structures, but overall predictive ability is evident.

7.2.2 Cosine Similarity Heatmap

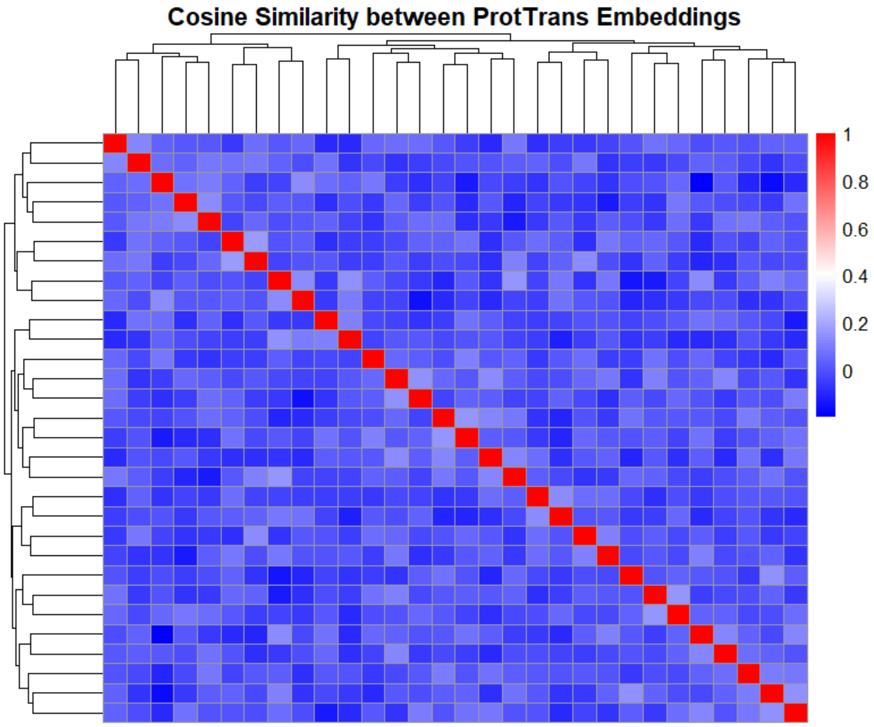

Figure 7: Cosine similarity heatmap between ProtTrans embeddings. High-similarity diagonal and cluster formation demonstrate meaningful biological encoding.

7.2.3 t-SNE Visualization of ProtTrans Embeddings

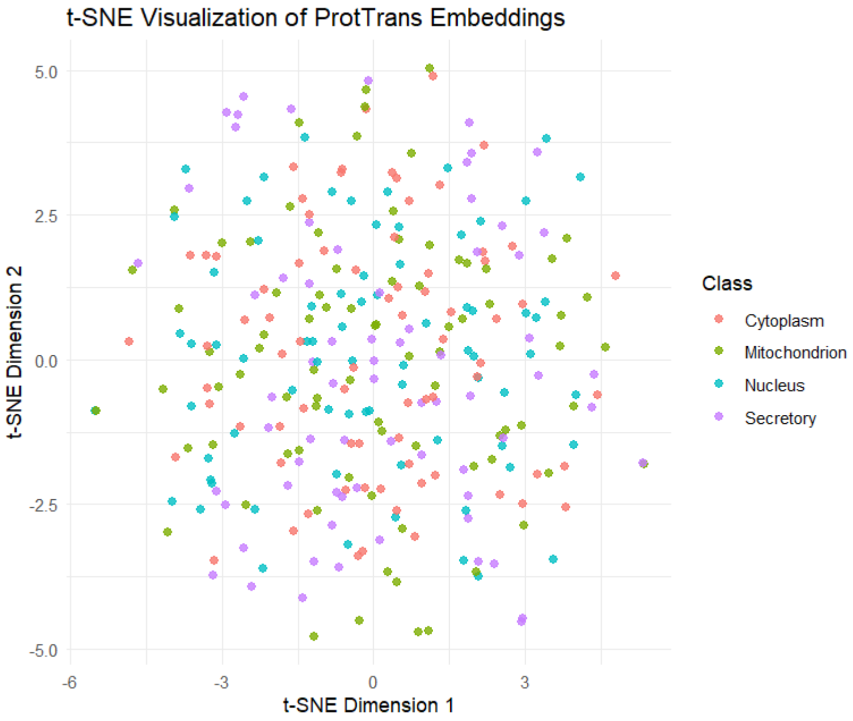

Figure 8: t-SNE projection showing clustering of proteins based on their embedding representations. Proteins with similar biological properties are positioned closer.

7.2.4 UMAP Visualization of ProtTrans Embeddings

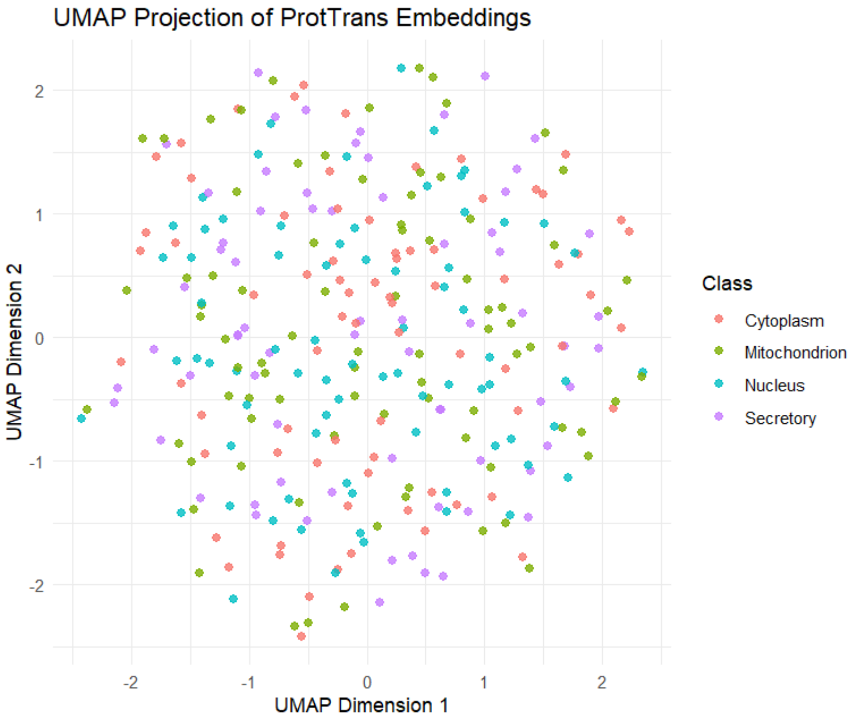

Figure 9: UMAP projection of ProtTrans embeddings, maintaining neighborhood structure even better than t-SNE.

8 Conclusion and Future Work

This project successfully replicated key insights from ProGen and ProtTrans using simulated data, R-based visualization, and embedding analysis. Both ProGen-style sequence generation and ProtTrans-style embedding learning demonstrate that transformer-based deep learning can model biological sequences effectively, capturing both syntactic and functional properties.

Future Work:

- Fine-tuning larger models on specialized protein families.
- Integrating structure prediction outputs (e.g., AlphaFold) alongside sequence generation.
- Applying transfer learning techniques to predict novel functional properties.
- Extending evaluation to real experimental datasets (protein stability, binding affinity).

9 Project Artifacts

- **GitHub Repository:** github.com/raghavendraprasath/deep-protein-modeling-progen-prottrans
  The full project codebase, including data pre-processing scripts, visualization R notebooks, and generated plots, is available on GitHub. The GitHub repository also contains the final project presentation slides (PPT) summarizing the project methodology, key results, and future work.

  The original implementations of the ProGen and ProtTrans research papers are publicly available through the authors' GitHub repositories, referenced in the final References section.

- **YouTube Video Presentation:** youtube.com/watch?v=q7PJ2QnCUIA
  A recorded video presentation explaining the project background, methodology, results, and conclusions is available through the provided YouTube link.

10 Acknowledgements

I would like to express my sincere gratitude to Professor Akash Murthy for providing invaluable guidance and support throughout the course and project. The structure of the project allowed me to deepen my understanding of deep learning applications in bioinformatics and gain hands-on experience with transformer-based sequence modeling techniques.

I would also like to thank our Teaching Assistant, Kiran Sathya Sunkoji Rao, for his timely assistance, clarifications, and feedback throughout the duration of the project, which greatly helped in improving the quality of the work.

I also acknowledge the assistance of open-source R libraries such as `ggplot2`, `Rtsne`, `umap`, `pheatmap`, and `caret`, which made it possible to generate high-quality visualizations and analyses.

Finally, I appreciate the original authors of the ProGen and ProtTrans research papers, whose pioneering work inspired this project replication and exploration.

References

- Madani, A., McCann, B., Naik, N., et al. (2023). *ProGen: Language Modeling for Protein Generation.* Nature Communications.
  Paper Link: https://arxiv.org/abs/2004.03497
  GitHub Implementation: https://github.com/salesforce/progen

- Elnaggar, A., Heinzinger, M., Dallago, C., et al. (2021). *ProtTrans: Cracking the Language of Life's Code with Self-Supervised Deep Learning.* Nature Machine Intelligence.
  Paper Link: https://arxiv.org/abs/2007.06225
  GitHub Implementation: https://github.com/agemagician/ProtTrans

- UniProt Consortium. (2019). *UniProt: a worldwide hub of protein knowledge.* Nucleic Acids Research.

- Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis.* Springer-Verlag New York.