# PROTEIN SECONDARY STRUCTURE PREDICTION

## IP1301- INTERNSHIP-II

**Report Submitted**

By

**JEROME B (CS22B1019)**
**N KATHIRAVAN (CS22B1036)**
**SAIRAM S (CS22B1048)**

To

Dr. Sanjay S. Bankapur
Assistant Professor
Department of Computer Science and Engineering
National Institute of Technology Puducherry
Karaikal - 609609


**DEPARTMENT OF**
**COMPUTER SCIENCE AND ENGINEERING**
**NATIONAL INSTITUTE OF TECHNOLOGY PUDUCHERRY**
**KARAIKAL – 609 609**

**MAY 2025**

# BONAFIDE CERTIFICATE

This is to certify that the project work phase-II entitled "PROTEIN SECONDARY STRUCTURE PREDICTION" is the bonafide record of the work done by "JEROME B (CS22B1019), N KATHIRAVAN (CS22B1036) and SAIRAM S (CS22B1048)" who carried out the Internship in the Department of Computer Science and Engineering at National Institute of Technology Puducherry, Karaikal during the period from Jan-2025 to May-2025.

**Project viva-voce held on: 14.05.2025**

**Dr. Sanjay S. Bankapur**
Supervisor
Assistant Professor
Department of CSE
NIT Puducherry, Karaikal

**Dr. Venkatesan M**
Head of the Department
Associate Professor
Department of CSE
NIT Puducherry, Karaikal

**Dr. Vani V**
Internship Coordinator
Assistant Professor
Department of CSE
NIT Puducherry, Karaikal

**External Examiner**

# ACKNOWLEDGEMENT

We express our sincere thanks and gratitude to our Institute, National Institute of Technology Puducherry (NITPY) for providing us with the opportunity, support and facilities to successfully complete this project. The conducive academic environment and infrastructure greatly contributed to our research endeavours.

We extend our sincere thanks and a token of appreciation to all the teaching faculty and non-teaching staff of Department of Computer Science and Engineering for their kind cooperation and for fostering a culture of learning, innovation and research for providing access to the AI server, which played a vital role in training and evaluating our models during this project.

We are deeply grateful to our supervisor **Dr. Sanjay S. Bankapur, Assistant Professor**, who took keen interest in our work and guided us all along in bringing out this project in complete shape by providing all the necessary information for developing a good system.

Finally, our sincere thanks to our friends for their continuous encouragement, insightful discussions and unwavering support, which helped us stay motivated and focused. We express our deepest gratitude to our family members for their endless support, understanding and motivation throughout this journey.

# TABLE OF CONTENTS

## LIST OF TABLES



## LIST OF FIGURES



## LIST OF ABBREVIATIONS

1. PSSP - Protein Secondary Structure Prediction
2. DNN - Deep Neural Network
3. BERT - Bidirectional Encoder Representations from Transformer
4. ProtBERT - Protein Bidirectional Encoder Representations from Transformer
5. ESM - Evolutionary Scale Modeling
6. MLP - Multi-Layer Perceptron
7. FFN – Feed Forward Network
8. ReLU - Rectified Linear Unit

# 1.0 INTRODUCTION

Proteins are essential biological macromolecules responsible for a wide range of cellular functions, and their behaviour is largely governed by their three-dimensional structures. Among the structural levels, secondary structure—comprising elements like alpha-helices, beta-sheets, and coils—serves as a foundational layer in understanding protein folding, stability, and function. Accurate prediction of protein secondary structure (PSS) is thus a critical problem in computational biology, with implications in drug design, protein engineering, and disease understanding. While classical machine learning approaches have made progress, they often struggle with capturing long-range dependencies and contextual information inherent in amino acid sequences.

Recent advancements in protein language models, such as ProtBERT and ESM, have introduced powerful contextual embeddings derived from large-scale protein sequence data. This project explores the use of these embeddings as inputs to deep neural networks for 8-class PSSP prediction. In addition to evaluating simple feedforward architectures, we developed a specialized Transformer model tailored to biological sequences. To push the boundaries further, we experimented with hybrid quantum-classical Transformers, integrating quantum layers to assess their potential in enhancing representational capacity and computational efficiency. This layered approach aims to provide insights into both the effectiveness of modern sequence embeddings and the emerging role of quantum computing in structural bioinformatics.

## 1.1 Motivation

Protein secondary structure prediction (PSSP) is essential for understanding protein functions and aiding drug discovery. While recent protein language models like ProtBERT and ESM offer rich sequence embeddings, leveraging them effectively for structural prediction remains an open challenge. This project aims to explore and compare deep learning approaches—ranging from simple MLPs to custom Transformer architectures—for PSSP. Furthermore, we investigate the potential of hybrid quantum-classical models, a frontier area that could bring computational advantages in modelling complex biological patterns. Our goal is to assess both performance and future applicability of these techniques in computational biology.

## 1.2 Applications:

- Drug Discovery and Design: Accurate PSS prediction helps identify functional regions in proteins, enabling the design of more effective and targeted drugs.
- Protein Engineering: Understanding secondary structures aids in modifying proteins for improved stability, solubility, or activity in industrial and therapeutic applications.
- Disease Research: Misfolded proteins are linked to diseases like Alzheimer's and Parkinson's. PSSP helps in identifying structural irregularities linked to such conditions.
- Structural Bioinformatics Pipelines: This model can serve as a module in full protein structure prediction pipelines (e.g., AlphaFold), especially when 3D structure prediction is computationally expensive.

- Fast Annotation of Uncharacterized Proteins: With increasing genomic data, your models can rapidly annotate secondary structures of proteins with unknown functions, aiding in functional genomics.
- Quantum Computing in Bioinformatics: Demonstrates early-stage use of hybrid quantum models for real-world biological data, opening pathways for quantum-accelerated structural predictions in the future.

# 2.0 LITERATURE SURVEY

Table 2.1: Literature Survey

| S. No. | Title | Author(s) | Methodology | Outcome |
|---|---|---|---|---|
| 1 | *Attention Is All You Need* | Vaswani et al., 2017 | Introduced the original Transformer architecture with self-attention, feed-forward layers, and positional encoding. | Enabled parallel sequence processing and accurate modeling of long-range dependencies. |
| 2 | *TransConv: Convolution-Infused Transformer for PSSP* | Das et al., 2025 | Combined Transformer encoders with 1D convolutions using ProtT5-XL-U50 embeddings and PCA for Q8 prediction. | Achieved improved accuracy by modeling both global and local protein structure features. |
| 3 | *Porter 6: PSSP Using PLMs* | Alanazi et al., 2024 | Used ESM-2 embeddings with a Bi-CRNN architecture for predicting secondary structure labels in Q8 format. | Demonstrated high Q8 accuracy through integration of pretrained language model embeddings. |
| 4 | *ESM-1b: Evolutionary Scale Modeling* | Rives et al., 2021 | Trained a 33-layer Transformer on 250M protein sequences to learn unsupervised sequence representations. | Showed strong performance on various protein prediction tasks, including PSSP. |
| 5 | *ProteinBERT: Deep Learning for Protein Sequence and Function* | Brandes et al., 2022 | Employed a BERT-style Transformer pre-trained on protein sequences and function annotations using masked language modeling. | Achieved generalizability across diverse tasks; learned biologically meaningful embeddings. |
| 6 | *TransformerCPI: Attention for Protein–Compound Interaction* | Chen et al., 2020 | Applied Transformers to model compound–protein interactions at the sequence level using token-wise self-attention. | Proved self-attention mechanisms effective in biological interaction modeling. |

## 2.1 Gaps:

Despite significant improvements, several limitations persist in the current landscape of protein secondary structure prediction:

- Lack of Quantum Integration: Most state-of-the-art models rely solely on classical computing paradigms. The potential of quantum computing to enhance representation and computational efficiency remains largely untapped in this domain.
- Overfitting to Specific Datasets: Many models, though high-performing on benchmark datasets, fail to generalize well to unseen proteins due to overfitting and insufficient regularization strategies.
- Computational Inefficiency: Models like TransConv and Porter 6 require significant computational resources due to their complex architectures and high-dimensional embeddings, limiting their real-world applicability.

- Underutilization of Token-Level Prediction Capabilities: While sequence classification is widely explored, token-level prediction (predicting a structure for each amino acid) using quantum-aware architectures is still in its infancy.
- Limited Benchmarking Across Diverse Datasets: Several models are evaluated on only one or two benchmark datasets, making it difficult to compare generalization across protein families.
- Quantum Hardware Limitations: Though theoretical quantum models exist, practical applications are hindered by hardware limitations, such as qubit decoherence and noise, affecting the stability of quantum-enhanced predictions.

## 2.2 Problem Statement:

Protein secondary structure prediction remains a fundamental and challenging problem in computational biology. The intricacies of protein folding involve complex interactions and dependencies across the amino acid sequence, which are difficult to capture using conventional methods. Classical machine learning and deep learning approaches have made strides in improving accuracy but are limited by computational overhead and modelling capacity. Moreover, the use of embeddings from protein-specific pre-trained models (e.g., ProtBERT, ESM-2) has shown promise but is not yet optimized for integration with custom architectures.

This project addresses the need for an efficient and accurate secondary structure prediction system by:

- Developing a custom Transformer model from scratch tailored for biological sequences.

- Leveraging rich embeddings generated by state-of-the-art pre-trained models.
- Integrating quantum computing components into the learning pipeline to potentially enhance pattern recognition, representation learning, and computational efficiency.

By bridging classical deep learning with quantum-inspired techniques, this project aims to contribute a novel hybrid model for protein secondary structure prediction, particularly at the token level, and validate it on benchmark datasets.

## 2.3 Objectives:

The core objectives of this research are:

- To explore and utilize various protein embedding models: Utilize state-of-the-art pre-trained models such as ProtBERT and ESM-2 to extract meaningful contextual embeddings of protein sequences.
- To design and implement a Transformer architecture from scratch: Develop a custom Transformer model specifically optimized for the prediction of protein secondary structure. The architecture will include encoder-decoder components, positional encoding, and multi-head attention mechanisms tailored to protein sequence data.
- To integrate quantum layers into the Transformer model: Introduce quantum-inspired or hybrid quantum-classical layers into the Transformer architecture to study their effect on learning capability, representation power, and prediction performance.
- To conduct a comparative study across models: Benchmark the performance of traditional models, pre-trained

embedding-based models (ProtBERT, ESM-2), the custom Transformer, and the quantum hybrid model on datasets such as CB513 and TS115.

- To evaluate the models using standard performance metrics: Analyze results based on Accuracy, Precision, Recall, F1-score, Q3, and Q8 metrics to provide a comprehensive performance assessment.
- To contribute toward the integration of quantum computing in bioinformatics:

  Investigate the feasibility and benefits of incorporating quantum computation in biological sequence modeling, laying the foundation for future advancements in this interdisciplinary domain.

# 3.0 METHODOLOGY

## 3.1 Dataset Characteristics

For comparison of pretrained embedding models, we used the NetSurfP-2.0 dataset for training, which contains over 10,000 non-redundant protein sequences, each annotated with 8-class (Q8) secondary structure labels derived from DSSP. Each sequence consists of amino acids represented by single-letter codes, and each residue is labelled with a structural class such as helix, strand, or loop. For evaluation, the models were tested on two benchmark datasets: CB513, comprising 513 non-redundant protein sequences, and TS115, which includes 115 non-redundant sequences. Both test datasets provide high-quality, residue-level Q8 annotations, enabling robust evaluation of the models' generalization to unseen protein data.

This project used the TS115 dataset, consisting of 115 non-redundant protein sequences annotated with 8-class (Q8) secondary structure labels derived from DSSP. Each sequence comprises amino acids represented by single-letter codes, and each token in the sequence is labelled with a structural class (e.g., helix, strand, loop).

## 3.2 Pre-Processing

- Tokenization: Protein sequences were tokenized character-wise using a custom CharTokenizer, mapping each amino acid to a unique integer.
- Label Encoding: Secondary structure labels (Q8 format) were mapped to integers using a label_map dictionary.
- Sequence Truncation & Padding: All sequences were fixed to a maximum length of 350 tokens. Shorter sequences were padded, and longer ones truncated. Corresponding attention masks were created.

- Train-Validation Split: The dataset was split 80:20 into training and validation sets using PyTorch's random_split utility.

## 3.3 Proposed Architecture

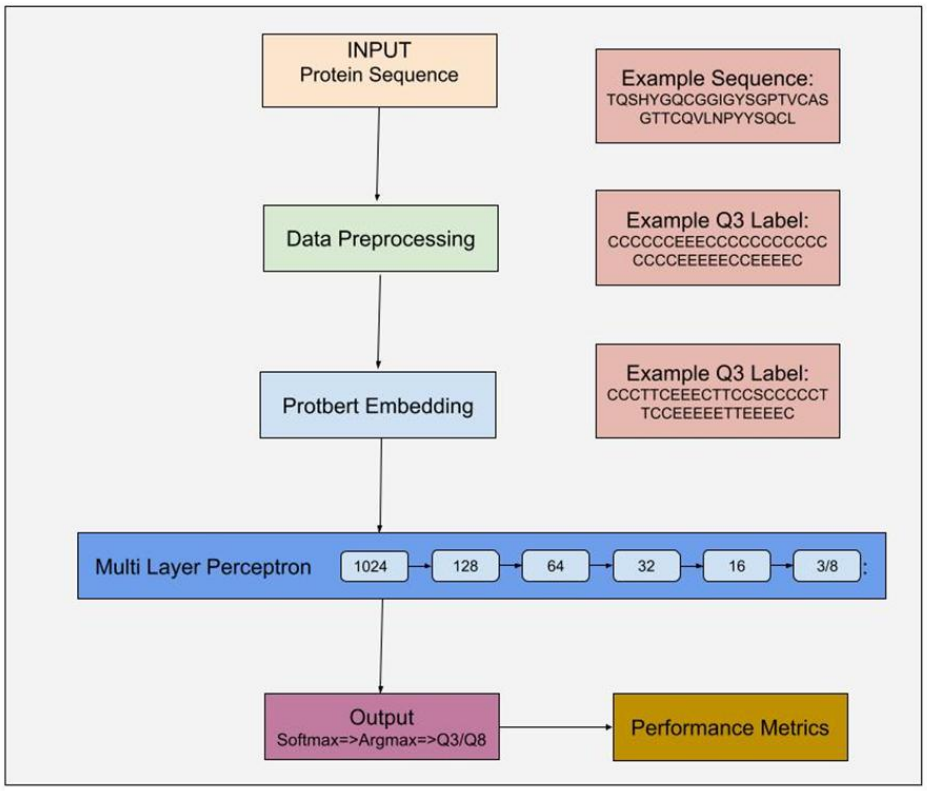


Fig. 3.3.1 Architecture of Protein Secondary Structure Prediction Model (Objective 1)

Protein sequences are first pre-processed and passed through a pre-trained ProtBERT model to obtain contextual embeddings for each amino acid. These embeddings are then fed into a Multi-Layer Perceptron (MLP) with progressively decreasing hidden layer sizes. The network outputs Q3 or Q8 class predictions using a softmax layer, and final predictions are evaluated using performance metrics. Example input sequences and corresponding labels are also shown for reference.

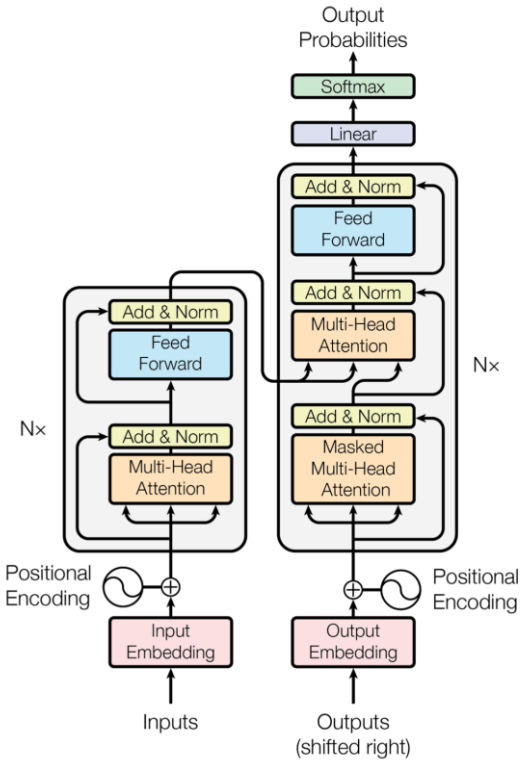


Fig. 3.3.2 Architecture of the Transformer

A Transformer model was developed from scratch in PyTorch, adopting the encoder-only variant of the original Transformer for token-level classification.

Model Components

- Input Embedding Layer: Maps each amino acid token into a 128-dimensional vector space (d_model = 128).
- Positional Encoding: Fixed sinusoidal encodings were added to the token embeddings to retain order information.
- Encoder Block: The architecture uses a single encoder layer comprising:
    - Multi-Head Attention: With 8 attention heads (n_heads = 8), enabling the model to attend to different positions in the sequence simultaneously.
    - Feed-Forward Network: Two linear layers with ReLU activation.

    - Layer Normalization & Dropout: Applied after attention and feed-forward sublayers to enhance generalization and convergence.

- Output Layer: A linear layer maps the output of each token to 8 class logits. A softmax layer is applied during inference.

To achieve quantum hybridization in a transformer, four essential components—such as the embedding layer, multi-head attention, feed-forward network, and post-attention normalization—can be replaced with quantum-inspired modules.

Table 3.3.1: Classical component vs Quantum Equivalent

| Classical Component | Quantum Equivalent |
|---|---|
| **Input Embedding** (Word → Vector) | **Quantum Encoding** (Angle/Amplitude Encoding) |
| **Self-Attention Mechanism** | **Quantum Self-Attention** |
| **Feedforward Network (FFN)** | **Quantum Feedforward Layer** |
| **Post-Attention Normalization** | **Quantum Normalization or Re-Encoding** |

## 3.4 Feature Modelling

The model learns protein sequence features from scratch, without relying on any pre-trained embeddings. Contextual relationships between amino acids are captured through self-attention, enabling the model to identify structural motifs directly from raw sequences.

## 3.5 Classification

The final model predicts the Q8 class for each token in the sequence. The following configurations were used:

- Loss Function: Cross-entropy loss calculated per token, ignoring padded positions using attention masks.
- Optimizer: Adam optimizer with a learning rate of 1e-4.
- Training Epochs: 1 (prototype phase).

- Batch Size: 32
- Epochs: 25
- Evaluation Metrics: Accuracy, Precision, Recall, and F1-score are computed on the validation set to monitor generalization.

# 4.0 PERFORMANCE ANALYSIS

## 4.1 Experimental Setup

- Environment: NVIDIA A100 AI Server
- Programming Language: Python
- Libraries: PyTorch, NumPy, Pandas, Scikit-learn, HuggingFace Transformers, Pennylane-Qiskit
- Models: MLP/DNN with ProtBERT and ESM embeddings, Custom Transformer for PSSP, Quantum-Classical Hybrid Transformer

## 4.2 Evaluation Metrics

In this study, the models were assessed using the following key metrics:

1. Accuracy: This metric measures the overall correctness of the model, representing the proportion of true results (both true positives and true negatives) among the total number of cases evaluated.

Accuracy = (True Positives + True Negatives) / Total Instances

2. Precision: Precision quantifies the number of correct positive predictions made out of all positive predictions. A higher precision indicates fewer false positives.

Precision = True Positives / (True Positives + False Positives)

3. Recall (Sensitivity): This metric evaluates the model's ability to correctly identify positive instances. It is the ratio of true positives to the sum of true positives and false negatives.

Recall = True Positives / (True Positives + False Negatives)

4. F1-Score: The F1-score is the harmonic mean of precision and recall. It provides a single metric that balances both concerns, particularly useful when dealing with imbalanced datasets.

F1-Score = 2*Precision*Recall / (Precision + Recall)

## 4.3 Result & Analysis

Model-1 => 1280 -> 256 -> 64 -> 16 -> 8

Model-2 => 1280 -> 512 -> 128 -> 32 -> 8

Model-3 => 1280 -> 512(0.2d) -> 128 -> 32 -> 8

Model-5 => 1280 -> 512(0.2d) -> 128(0.2d) -> 32 -> 8

Model-6 => 1280 -> 512(0.2d) -> 128(0.2d) -> 32 -> 8 + L2 Reg. Also

Model-7 => 1280 -> 512(0.2d) -> 128(0.2d) -> 32(0.2d) -> 8

### esm2_t33_650M_UR50D

Table 4.3.1: Results of ESM2 Embeddings fed to DNN (in %)

| MODELS | CB513 ACCURACY | CB513 PRECISION | CB513 RECALL | CB513 F1-SCORE | TS115 ACCURACY | TS115 PRECISION | TS115 RECALL | TS115 F1-SCORE |
|---|---|---|---|---|---|---|---|---|
| Model-1 | 74.16 | 61.30 | 53.65 | 55.72 | 76.86 | 63.94 | 54.12 | 56.86 |
| Model-2 | 73.50 | 59.27 | 54.20 | 55.63 | 76.08 | 61.66 | 54.46 | 56.55 |
| Model-3 | 74.74 | 64.65 | 53.65 | 55.99 | 77.45% | 66.12 | 54.90 | 57.81 |
| L2 Regularization | 70.02 | 40.16 | 39.77 | 38.13 | 73.07 | 41.50 | 39.65 | 38.31 |
| **Model-5** | **74.88** | **66.25** | **54.24** | **57.00** | **77.45** | **66.35** | **54.52** | **57.58** |
| Model-6 | 69.67 | 40.16 | 39.64 | 37.47 | 72.82 | 41.31% | 39.72 | 37.95 |
| Model-7 | 74.86 | 65.34 | 53.68 | 56.59 | 77.12 | 67.99 | 53.84 | 57.23 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **TransConV** | **76.10** | **65.00** | **57.00** | **60.00** | **79.10** | **67.00** | **59.00** | **61.00** |

**prot_bert**

Table 4.3.2: Results of ProtBERT Embeddings fed to DNN (in %)

| MODEL | CB513 ACCURACY | CB513 PRECISION | CB513 RECALL | CB513 F1 SCORE | TS115 ACCURACY | TS115 PRECISION | TS115 RECALL | TS115 F1 SCORE |
|---|---|---|---|---|---|---|---|---|
| Model-1 | 62.70 | 48.82 | 36.38 | 36.57 | 69.07 | 53.54 | 38.77 | 39.57 |
| Model-2 | 61.84 | 41.99 | 36.94 | 37.74 | 68.51 | 47.27 | 39.50 | 40.62 |
| Model-3 | 63.48 | 49.97 | 36.96 | 37.16 | **69.79** | **52.38** | **39.33** | **40.03** |
| L2 Regulariza-tion | 59.64 | 28.91 | 31.66 | 29.54 | 65.89 | 31.38 | 33.23 | 31.51 |
| Model-5 | 63.36 | 51.36 | 36.73 | 36.74 | 69.52 | 58.74 | 39.10 | 39.89 |
| **Model-7** | **63.50** | **55.92** | **36.61** | **36.42** | 69.60 | 65.92 | 38.83 | 39.07 |
| **TransConV** | **76.10** | **65.00** | **57.00** | **60.00** | **79.10** | **67.00** | **59.00** | **61.00** |

ESM and ProtBERT models, evaluated using esm2_t33_650M_UR50D and prot_bert respectively, show varying performance compared to TransConV, which consistently outperforms them with 76.10% accuracy and 60.00% F1-score for CB513, and 79.10% accuracy and 61.00% F1-score for TS115; among ESM models, the 1280 → 512(0.2d) → 128(0.2d) → 32 → 8 configuration performs best for CB513 at 74.88% accuracy, while the same configuration excels for TS115 at 77.45% accuracy; for ProtBERT, the 1280 → 512(0.2d) → 128(0.2d) → 32(0.2d) → 8 setup achieves the highest CB513 accuracy at 63.50%, and the 1280 → 512(0.2d) → 128 → 32 → 8 configuration leads for TS115 at 69.79% accuracy, though L2 regularization significantly reduces performance

(e.g., 70.02% accuracy for CB513 in ESM), while dropout layers improve generalization.

Table 4.3.3: PSSP Transformer Results

| Exp. No. | Batch Size | Encoders | Decoders | Attention Heads | Accuracy (%) | Precision(%) | Recall (%) | F1 Score (%) |
|---|---|---|---|---|---|---|---|---|
| 1 | 16 | 1 | 1 | 1 | 31.23 | 33.63 | 30.58 | 31.70 |
| 2 | 16 | 2 | 2 | 2 | 37.52 | 39.23 | 35.85 | 37.49 |
| 3 | 32 | 2 | 2 | 4 | 41.86 | 43.06 | 40.15 | 41.51 |
| 4 | 64 | 3 | 2 | 4 | 45.69 | 47.34 | 44.03 | 45.64 |
| 5 | 32 | 4 | 3 | 6 | 48.90 | 46.26 | 50.17 | 48.12 |
| 6 | 64 | 4 | 4 | 8 | 59.38 | 53.63 | 51.74 | 57.32 |

The experimental results demonstrate a clear trend of performance improvement as the model complexity increases. Starting from a simple configuration with one encoder, one decoder, and one attention head (Exp. 1), which yielded the lowest accuracy (31.23%) and F1-score (31.7%), each subsequent increase in the number of encoders, decoders, attention heads, and batch size led to consistent gains in all metrics. Notably, **Experiment 6**, with the highest configuration (batch size 64, 4 encoders, 4 decoders, and 8 attention heads), achieved the best results—**59.38% accuracy** and **57.32% F1-score**—indicating that deeper and wider transformer architectures significantly enhance model performance. This progression highlights the importance of scaling transformer components to achieve better learning and generalization.

Table 4.3.4: Hybrid Quantum Transformer Results

| Modification | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| Angular Encoding | 52.87 | 38.19 | 35.21 | 30.59 |
| Quantum Attention | 49.83 | 41.90 | 38.39 | 40.06 |
| Quantum Feed Forward Layer | 62.83 | 57.89 | 50.60 | 51.15 |
| Re-Encoding | 49.73 | 36.29 | 34.28 | 35.26 |

Among the various quantum hybrid modifications tested, the **Quantum Feed Forward Layer** demonstrated the most significant improvement, achieving the highest accuracy (62.83%), precision (57.89%), recall (50.6%), and F1-score (51.15%), indicating its strong impact on overall model performance. In contrast, **Angular Encoding** and **Re-Encoding** resulted in the lowest metrics, showing minimal benefit from quantum integration in those components. **Quantum Attention** showed moderate performance, with balanced precision and recall but lower accuracy. These results suggest that while some quantum replacements may offer limited gains, integrating quantum elements into the feed-forward layer holds the most promise for enhancing transformer-based models.

# 5.0 CONCLUSION & FUTURE WORK

## 5.1 Conclusion

This project addressed the task of protein secondary structure prediction (PSSP) using a blend of modern machine learning and quantum computing techniques. We began by utilizing pre-trained protein language models, ProtBERT and ESM, to generate contextual embeddings from raw amino acid sequences. These embeddings were then passed through deep learning models, including MLPs, to perform 8-class secondary structure classification. To further capture sequential dependencies, we implemented a custom Transformer model tailored to protein sequences. Finally, we extended our work by integrating hybrid quantum-classical layers into the Transformer architecture, demonstrating the viability of applying quantum computing techniques in bioinformatics. Overall, our approach showed promising results and highlighted the potential of combining classical and quantum methods for complex biological sequence modeling.

## 5.2 Future Work

Moving forward, several extensions can enhance this research. First, the quantum components of the hybrid model can be optimized using more advanced variational quantum circuits, quantum feature maps, and efficient encoding strategies. Exploring full quantum attention mechanisms and testing on actual quantum hardware would provide deeper insights into real-world feasibility. Second, training on larger and more diverse datasets

can improve the robustness and generalizability of the models. Integration into full protein structure prediction pipelines or comparison with end-to-end models like AlphaFold could help evaluate real-world performance. Finally, incorporating domain adaptation or fine-tuning techniques for specific protein families, as well as exploring interpretability tools, would make the models more applicable to biological and pharmaceutical research contexts.

# REFERENCES


[1] Alanazi, W., Meng, D., & Pollastri, G. “Porter 6: Protein secondary structure prediction by leveraging pre-trained language models (PLMs),” *Int. J. Mol. Sci.*, vol. 26, no. 1, p. 130, 2024. doi: https://doi.org/10.3390/ijms26010130

[2] Cordoves Delgado, Greneter & García-Jacas, César. (2024). “Predicting Antimicrobial Peptides Using ESMFold-Predicted Structures and ESM-2-Based Amino Acid Features with Graph Deep Learning.” *Journal of Chemical Information and Computer Sciences*.doi: 10.1021/acs.jcim.3c02061

[3] Das, S., Ghosh, S., & Jana, N. D., “TransConv: Convolution-Infused Transformer for Protein Secondary Structure Prediction,” *J. Mol. Model.*, vol. 31, no. 37, 2025. doi: 10.1007/s00894-024-06259-7

[4] Kollias, G., Kalantzis, V., Salonidis, T., & Ubaru, S., "Quantum Graph Transformers," *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Rhodes Island, Greece, 2023, pp. 1-5. doi: 10.1109/ICASSP49357.2023.10096345

[5] Manfredi, Matteo & Savojardo, Castrense. (2024). “E-pRSA: Embeddings Improve the Prediction of Residue Relative Solvent Accessibility in Protein Sequence.” *Journal of Molecular Biology*. 436. 168494. doi: 10.1016/j.jmb.2024.168494

[6] Meier, Joshua, Rao, Roshan, Verkuil, Robert, Liu, Jason, Sercu, Tom & Rives, Alexander. (2021). “Language models enable zero-shot prediction of the effects of mutations on protein function.” doi: 10.1101/2021.07.09.450648

[7] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I., “Attention Is All You Need,” in *Proc. 31st Conf. Neural Inf. Process. Syst. (NIPS)*, Long Beach, CA, USA, 2017. doi: 10.48550/arXiv.1706.03762

[8] Wang, YiMing & Fang, Chun. (2024). “Cycle-ESM: Generation-assisted classification of antifungal peptides using ESM protein language model.” *Computational Biology and Chemistry*. 113. 108240. doi: 10.1016/j.compbiolchem.2024.108240