成都理工大学中英合作办学
CDUT Sino-British Collaborative Education

OXFORD BROOKES UNIVERSITY

UNDERGRADUATE PROJECT REPORT

Project Title:	**Protein Secondary Structure Prediction Based on Triple Fusion Explainable Model**
Surname:	**Chu**
First Name:	**Xiyan**
Student Number:	**202118010204**
Supervisor Name:	**Dr Grace Ugochi Nneji**
Module Code:	**CHC 6096**
Module Name:	**Project**
Date Submitted:	**May 6, 2025**

Chengdu University of Technology Oxford Brookes College

Chengdu University of Technology

BSc (Single Honours) Degree Project

Programme Name: Computer Science

Module No.: CHC 6096

Surname: Chu

First Name: Xiyan

Project Title: Protein Secondary Structure Prediction Based on Triple Fusion Explainable Model

Student No.: 202118010204

Supervisor: Dr Grace Ugochi Nneji

Date submitted: 6th May 2025

A report submitted as part of the requirements for the degree of BSc (Hons) in Computer Science

At

Chengdu University of Technology Oxford Brookes College

Declaration

Student Conduct Regulations:

Please ensure you are familiar with the regulations in relation to Academic Integrity. The University takes this issue very seriously and students have been expelled or had their degrees withheld for cheating in assessment. It is important that students having difficulties with their work should seek help from their tutors rather than be tempted to use unfair means to gain marks. Students should not risk losing their degree and undermining all the work they have done towards it. You are expected to have familiarised yourself with these regulations.

https://www.brookes.ac.uk/regulations/current/appeals-complaints-and-conduct/c1-1/

Guidance on the correct use of references can be found on www.brookes.ac.uk/services/library, and also in a handout in the Library.

The full regulations may be accessed online at https://www.brookes.ac.uk/students/sirt/student-conduct/

If you do not understand what any of these terms mean, you should ask your Project Supervisor to clarify them for you.

I declare that I have read and understood Regulations C1.1.4 of the Regulations governing Academic Misconduct, and that the work I submit is fully in accordance with them.

Signature Chu Xiyan (Cecilia) Date ……6th May 2025……

REGULATIONS GOVERNING THE DEPOSIT AND USE OF OXFORD BROOKES UNIVERSITY MODULAR PROGRAMME PROJECTS AND DISSERTATIONS

Copies of projects/dissertations, submitted in fulfillment of Modular Programme requirements and achieving marks of 60% or above, shall normally be kept by the Oxford Brookes University Library.

I agree that this dissertation may be available for reading and photocopying in accordance with the Regulations governing the use of the Oxford Brookes University Library.

Signature Chu Xiyan (Cecilia) Date ……6th May 2025……

Acknowledgment

I would like to sincerely thank my supervisor, Dr. Grace Ugochi Nneji, who has given me unwavering guidance and support for my undergraduate project. Dr. Grace's professional knowledge, patience and encouragement played a significant role in the completion of this project.

Additionally, I wish to express gratitude to Joojo Walker, the module leader, and all the other educators who have imparted their knowledge and offered invaluable advice throughout my undergraduate journey.

Furthermore, I am grateful for the resources and facilities made available through the collaborative efforts of Oxford Brookes University and Chengdu University of Technology, which have provided an exceptional environment for academic growth.

Lastly, to my cherished family and friends, your long-term love and support are the driving force for my progress. Thank you for your company and dedication on my path of growth.

Table of Contents

List of Figures

List of Table

Abstract

Proteins are the fundamental biomolecules that perform multiple functions in living organisms. The structure of proteins is divided into four levels: primary, secondary, tertiary and quaternary structures. Among them, the secondary structure is formed by hydrogen bonds, including α -helix, β -fold and random coiling, which is crucial for the stability and function of the protein. Although experimental methods such as X-ray crystallography and nuclear magnetic resonance imaging (NMR) can provide high-resolution structural information, they are costly and time-consuming, and are not suitable for large-scale applications. This project develops a novel deep learning Model called Triple Fusion Explainable Model which integrates with Convolutional Structural Predictor (CSP) model, Memory-based Structure Network model, and Self-Attention Convolutional Structural Predictor model, enabling it to deeply extract local features and long-distance dependencies in protein sequences, in order to improve the accuracy and efficiency of protein secondary structure classification. Finally, training and validation are conducted on the Protein Secondary Structure dataset and the Protein Data Bank (PDB) dataset from Kaggle. The model achieved remarkable predictive performance of 95.64% accuracy, AUC of 98.82% and F1-score of 95.55%. The experimental results of the proposed model indicate its robust capability and generalization in the prediction of protein secondary structure and a valuable AI-based tool necessary in medical settings.

Keywords: *Protein Secondary Structure, Deep Learning, Convolutional Structural Predictor, Memory-based Structure Network, Attention Mechanism*

Abbreviations

DNA: Deoxyribonucleic Acid

NMR: nuclear magnetic resonance

CNN: Convolutional Neural Network

PDB: Protein Data Bank

AUC: Area Under Curve

CSP: Convolutional Structural Predictor

Attention-CSP: Self-Attention Convolutional Structure Predictor

GOR: Garnier-Osguthorpe-Robsons

HMM: Hidden Markov Models

LSTM: Long Short-Term Memory

PSSM: Position-Specific Scoring Matrix

CSV: Comma Separated Value

sst3: three-state labels

sst8: eight-state labels

ReLU: Rectified Linear Unit

Acc: Accuracy

TN: True Negative

TP: True Positive

FP: False Positive

FN: False Negative

GUI: Graphical User Interface

GPU: Graphics Processing Unit

Glossary

Protein Secondary Structure: The secondary structure of a protein refers to the specific conformation formed by the polypeptide backbone atoms spiraling or folding along a certain axis.

Deep Learning: Learning from large amounts of data by using network with multiple layers of processing units, and is widely used for tasks such as image recognition, speech recognition, and semantic segmentation.

Convolutional Structural Predictor: An architecture of network utilized for deep learning mission which are commonly used for computer vision projects

Attention Mechanism: Attention Mechanism allows deep learning models to dynamically focus on important parts of the input data. It plays a key role in improving the explainable and performance of the model.

Memory-based Structure Network: An improved recurrent neural network (RNN) used for processing sequential data. By considering both forward and backward context information simultaneously, it enhances the model's performance in sequence tasks.

One-Hot Encoding: One-Hot Encoding is a technique for converting categorical variables into binary vectors. Its main purpose is to transform categorical features into a format suitable for machine learning algorithms.

Chapter 1 Introduction

1.1 Background

Proteins are essential biomolecules in living organisms, performing diverse functions such as catalyzing metabolic reactions, facilitating DNA replication, forming cellular structures, constructing tissues, and supporting the immune system by generating antibodies [1]. They are the most abundant organic molecules in the body and serve as structural components, energy sources, and facilitators of muscle contraction.

Protein structure is classified into four hierarchical levels: primary, secondary, tertiary, and quaternary. The primary structure refers to the linear sequence of amino acids forming a polypeptide chain. The secondary structure arises from hydrogen bonding within the polypeptide backbone, leading to regular formations such as helices, strands, and coils. Further folding of these secondary structures results in the tertiary structure, which defines the three-dimensional conformation of the protein and is crucial for its function. The quaternary structure consists of multiple polypeptide chains assembled into a functional protein complex [2]. Understanding a protein's structure is crucial for determining its function and exploring ways to regulate, modify, or manipulate it. Additionally, this knowledge plays a vital role in drug development and enzyme design [3]. However, deriving a protein's three-dimensional structure from its primary sequence demands significant computational resources, making it one of the most challenging problems in computational biology [4].

Therefore, protein secondary structure prediction has attracted widespread attention as it serves as an intermediate step in three-dimensional structure prediction [5]. The secondary structure of proteins includes key types such as α-helices, β-sheets, and random coils, which are crucial for the stability and function of proteins [6]. Traditional experimental methods, such as X-ray crystallography and nuclear magnetic resonance imaging (NMR), can offer high-resolution structural information but are costly, time-consuming, and unsuitable for large-scale applications [7]. Traditional machine learning methods, such as support vector machines and hidden Markov models, have achieved some success in secondary structure prediction but struggle to capture complex features and long-range dependencies in sequences [8]. Recently, the introduction of deep learning, particularly convolutional neural networks (CNNs), has significantly improved prediction efficiency and accuracy. This project aims to develop an efficient

protein secondary structure prediction model by optimizing deep learning architectures, providing robust tools for protein structure and related disease research.

1.1.1 Risk and Factor

In biological and medical research, understanding the risks and factors that influence the secondary structure of proteins is crucial. These factors can be divided into extrinsic (environmental) factors and intrinsic (biological) factors, which affect the structure and function of proteins [9]. Additionally, the prediction and analysis of protein secondary structures face specific risks that could impact the accuracy and safety of scientific research. The following is a thorough analysis of these factors and risks:

Extrinsic Factors:

- **Temperature:** High temperatures can disrupt hydrogen bonds and other interactions that stabilize secondary structures.
- **Agitation:** Physical agitation can lead to mechanical stress, potentially disrupting secondary structures.
- **Radiation:** Exposure to radiation can cause damage to the protein structure.
- **Pressure:** High pressure can alter the stability of secondary structures.
- **Buffer Conditions:** The ionic strength and composition of the buffer can influence protein folding and stability.
- **pH:** Changes in pH can affect the ionization states of amino acids, impacting hydrogen bonding and electrostatic interactions.

Intrinsic Factors:

- **Protein Aggregates:** Aggregation can lead to misfolding and disruption of secondary structures.
- **Pressure Inside the Cell:** Internal cellular pressure can affect protein conformation.
- **Oxidative Stress:** Reactive oxygen species can damage amino acids, affecting the protein structure.
- **Ageing:** Over time, proteins can undergo conformational changes that affect their secondary structure.
- **Impaired Autophagy:** Inefficient removal of damaged proteins can lead to accumulation and structural changes.
- **Mutation:** Genetic mutations can alter the amino acid sequence, potentially disrupting secondary structures.

These factors can influence the stability and formation of secondary structures such as alpha-helices and beta-sheets in proteins. Figure 1 shows the summary of factors for protein structure.

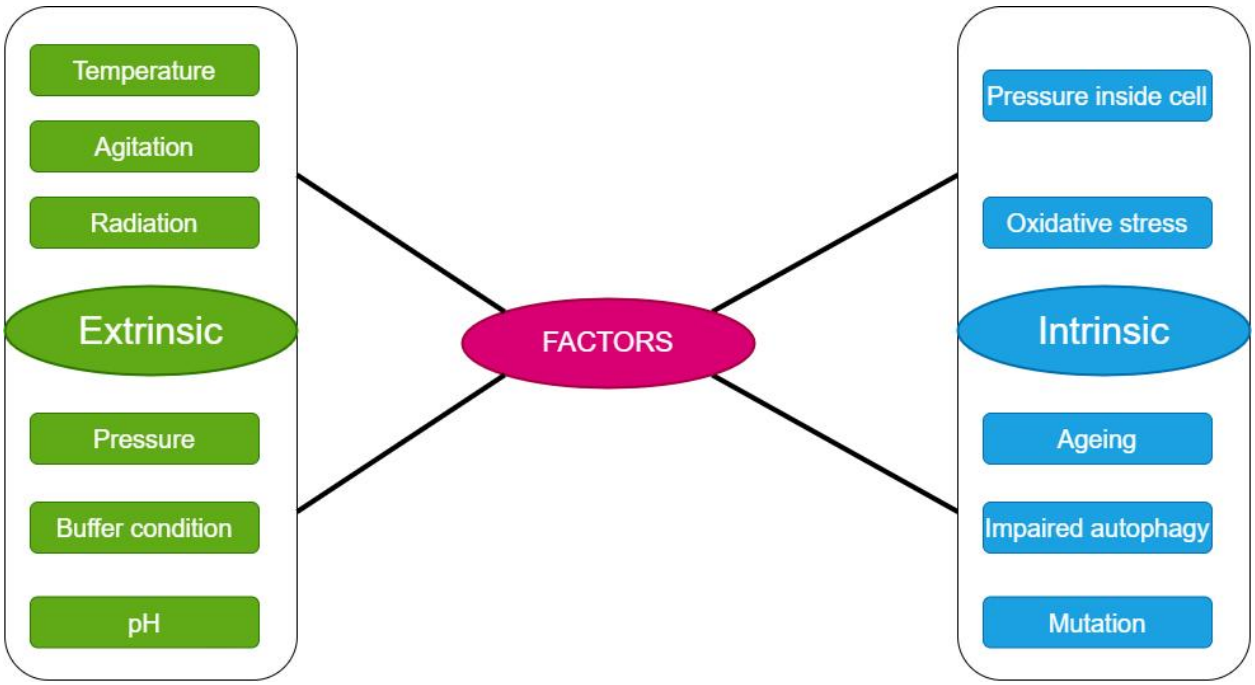


Figure 1 Factors of protein structure

Risks about protein secondary structure:

- **Misfolding and Disease Association** – Incorrect secondary structures can lead to protein misfolding, contributing to diseases like Alzheimer's, Parkinson's, and prion disorders.
- **Protein Aggregation and Toxicity** – Errors in secondary structure formation can cause aggregation, forming amyloid fibrils that disrupt cellular function.
- **Loss of Stability and Function** – Misformed secondary structures can render proteins non-functional or prone to degradation, affecting biological pathways.
- **Disordered Regions and Prediction Challenges** – Some proteins have intrinsically disordered regions that do not adopt stable structures, making them hard to predict.

Figure 2 illustrates the potential outcomes of protein sequence.

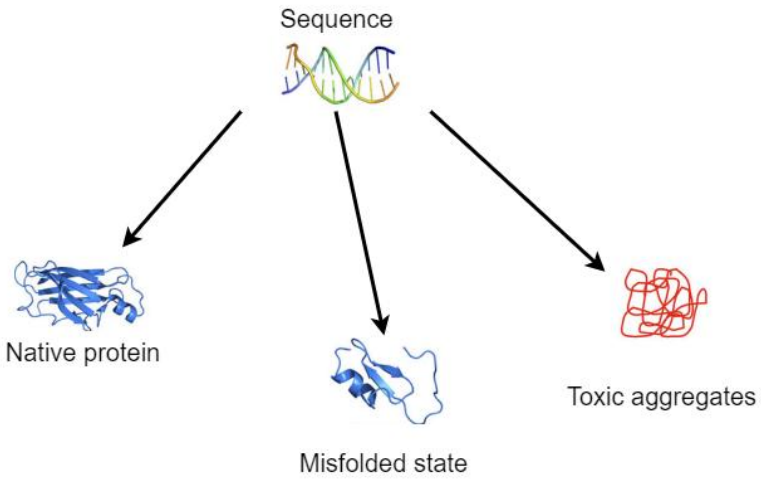


Figure 2 Risks of protein structure

Risk about protein secondary structure prediction:

- **Misclassification in Prediction** – Computational models may misidentify helices, sheets, or loops, leading to incorrect structural and functional interpretations.
- **Impact on Drug Discovery** – Errors in secondary structure prediction can lead to ineffective or harmful drug designs, delaying biomedical advancements.
- **Ethical and Security Concerns** – The ability to predict and manipulate protein structures raises concerns about dual-use risks, such as designing synthetic proteins for harmful purposes.
- **Over-Reliance on AI-Based Predictions** – While AI models improve accuracy, experimental validation is still necessary to ensure reliability and avoid misleading conclusions.

1.1.2 Challenge

- **Limited Experimental Data** – Many proteins lack high-resolution structural data, limiting training sets for AI models.
- **Intrinsic Disorder in Proteins** – Some proteins do not fold into stable secondary structures, making their prediction difficult.
- **Complex Folding Mechanisms** – Interactions between secondary and tertiary structures complicate predictions, requiring more computational power.
- **Prediction Accuracy** – While methods like neural networks improve accuracy, predicting rare or novel folds remains challenging.

- **Computational Cost** – Accurate simulations, such as molecular dynamics, require high-performance computing resources.

1.2 Aim

The primary objective of this project is to develop and optimize a deep learning-based protein secondary structure prediction model by leveraging Convolutional Structural Predictor (CSP), Memory-based Structure Network, and Self-Attention Convolutional Structural Predictor. The model aims to deeply extract both local features and long-range dependencies within protein sequences to improve the accuracy and efficiency of protein secondary structure classification. This will provide a reliable computational tool for protein folding research and the exploration of mechanisms related to associated diseases. Furthermore, the outcomes of this project are expected to provide new research insights for further investigating the relationship between protein structure and function, as well as the mechanisms underlying diseases related to protein misfolding.

1.3 Objectives

The project will collect protein sequence data from Kaggle, utilizing datasets like Protein Data Bank (PDB), and Protein Secondary Structure. The dataset is divided into training, validation, and test sets, and the protein sequences are processed using one-hot encoding.

The model in this project adopts an architecture that combines Convolutional Structural Predictor (CSP), Memory-based Structure Network, and Self-Attention Convolutional Structural Predictor to effectively capture local features and long-range dependencies within protein sequences. During training, hyperparameters are adjusted, and techniques such as dropout and early stopping are employed to enhance the model's generalization capability and prevent overfitting.

Additionally, the model evaluation will include metrics such as Loss and Accuracy. The performance will be assessed using Precision, Recall, Specificity, F1-Score, Sensitivity, Confusion Matrix, and ROC-AUC.

Lastly, the project will be deployed through GUI which allows uploading the sequence of protein, and then give the classification results.

1.4 Project Overview

This chapter will introduce different methods of protein structure prediction. Start with traditional methods and move on to machine learning and deep learning techniques to show how this field has evolved and technologically advanced over time.

1.4.1 Scope

The objective of this project is to leverage deep learning techniques to develop a model that integrates Convolutional Structural Predictor (CSP), Memory-based Structure Network, and Self-Attention Convolutional Structural Predictor to classify the secondary structure of protein sequences. Protein secondary structure prediction is a critical step in understanding protein folding mechanisms and plays an essential role in uncovering the relationships between protein function, structure, and disease. However, due to the complexity of protein sequences and their long-range dependencies, traditional prediction methods often lack sufficient accuracy and long processing times.

This project holds significant value. On the one hand, deep learning techniques enable the effective extraction of local features and long-range dependencies in protein sequences, improving prediction accuracy and providing efficient tools for studying protein folding and misfolding mechanisms. On the other hand, the outcomes of this project can provide data support for research on protein function, disease mechanisms, and drug development, offering broad application potential. Furthermore, this project can advance the application of deep learning techniques in the life sciences and provide new directions for research in computational biology.

1.4.2 Audience

The audience for this project includes the following groups:

- **Bioinformatics Researchers:** This project provides new approaches for protein secondary structure classification, helping biologists efficiently analyze the structure-function relationships of proteins, thus advancing research in fields such as drug design and genetic engineering.
- **Medical Researchers:** Particularly those working on rare genetic diseases related to protein folding, this project offers a new perspective for studying disease mechanisms and helps identify potentially disease-related proteins.

- **Pharmaceutical Companies:** By improving the accuracy of protein secondary structure prediction, this method can accelerate the screening of drug targets and the design of drug molecules, shortening the drug development cycle.
- **Computational Biologists and Data Scientists:** The deep learning models used in this project, combining Convolutional Structural Predictor (CSP), Memory-based Structure Network, and Self-Attention mechanisms, provide a reference for computational biologists working with protein data analysis and offer data scientists an example of applying deep learning techniques in biology.

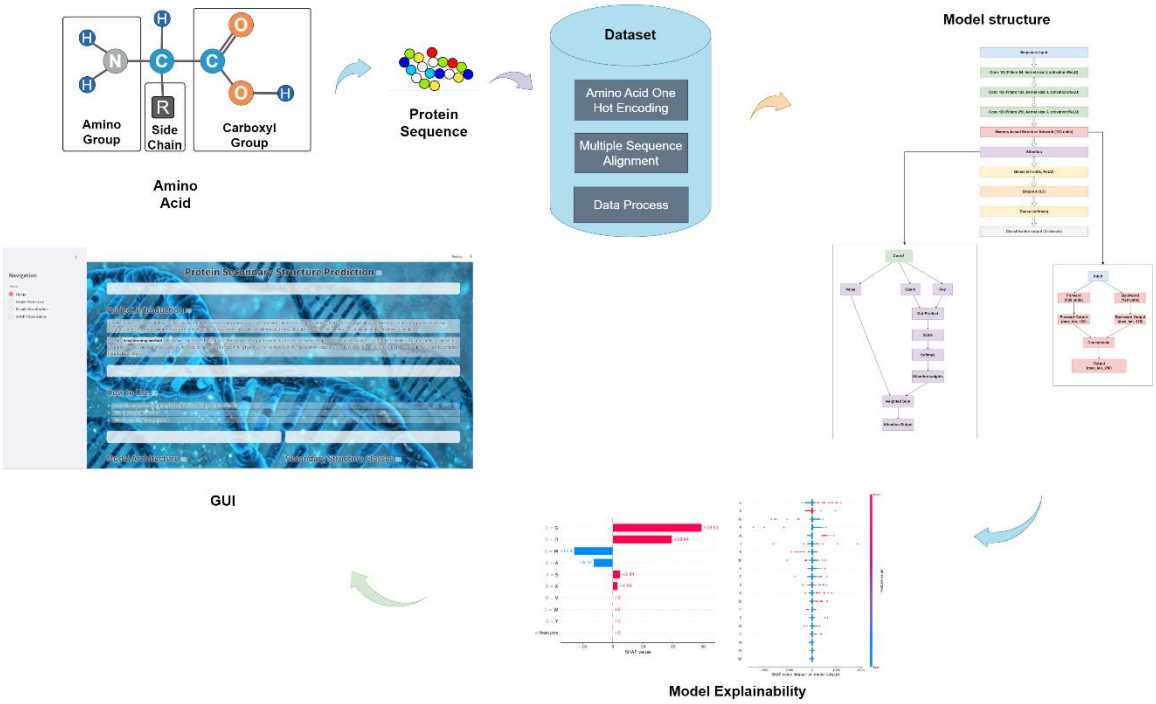


Figure 3 Workflow of the project

Chapter 2 Background Review

This chapter will introduce different methods of protein structure prediction. Start with traditional methods and move on to machine learning and deep learning techniques to show how this field has evolved and technologically advanced over time.

2.1 Protein Structure Prediction Using Traditional Method

Accurate prediction of protein secondary structure not only aids in understanding its function but also supports disease research and drug development. Early prediction methods were primarily based on statistical analysis and empirical rules, among which the Garnier-Osguthorpe-Robson (GOR) method is of milestone significance. Figure 4 shows the architecture of the GOR method.

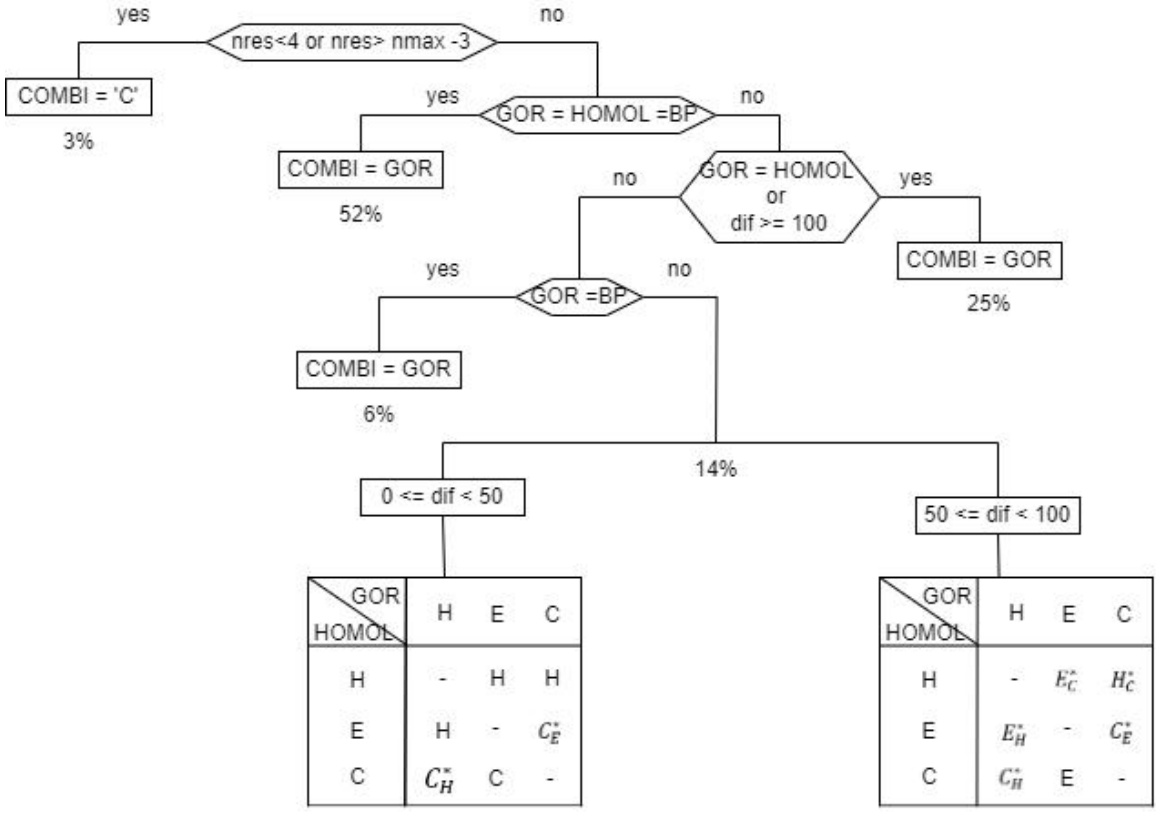


Figure 4 Garnier and Robson's architecture of the GOR method [10]

The GOR method [10], introduced by Garnier and Robson, is a statistical approach to predicting secondary structures in proteins. By analyzing known protein X-ray crystallographic structures available at the time, this method calculates conformational parameters for each amino acid, reflecting its tendency to form alpha-helices (H), beta-sheets (E), turns (T), or random coils (C). The essence of the method is to use these

parameters to identify nucleation sites within a sequence and extend the predicted structure based on empirical rules. The GOR method laid the groundwork for structure prediction with its simplicity and achieved prediction accuracy ranging from 60% to 70%. However, it primarily focuses on local interactions among amino acids, overlooks long-range interactions, and relies on fixed empirical rules, making it difficult to deal with complex sequences.

2.2 Protein Structure Prediction Using Machine Learning

With the development of computing technology, Hidden Markov Models (HMM) have been introduced in protein secondary structure prediction, providing a probabilistic modeling method capable of capturing dependencies between sequences, thus achieving significant progress in prediction accuracy and applicability [11]. Figure 5 shows the structure of Hidden Markov Model.

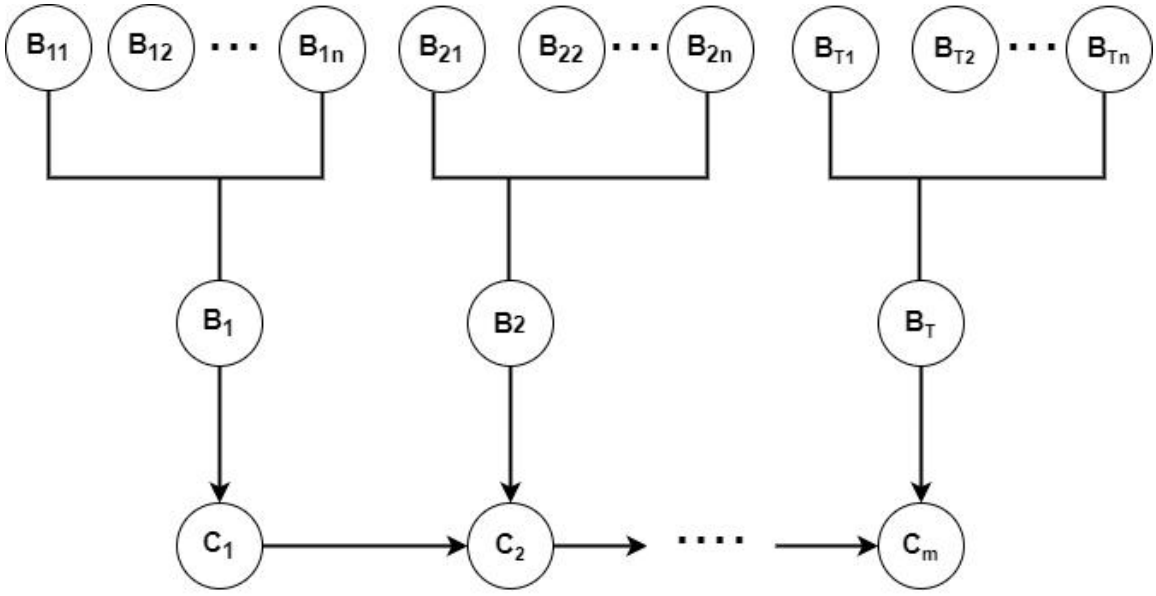


Figure 5 Chelvi and Rangarajan's Hidden Markov Model [11]

The Hidden Markov Model assumes that protein sequences can be viewed as a series of observed amino acid residues, generated by a set of unobservable "hidden states." Each hidden state is associated with a specific secondary structure (such as α-helix, β-sheet, and random coil), and the model represents the relationships between different secondary structures in the sequence through state transition probabilities and emission probabilities. A key advantage of HMM is its ability to simultaneously capture local patterns and inter-sequence dependencies, making it particularly suitable for handling

complex biological sequence data. Although HMM has shown good performance in protein secondary structure prediction, its ability to model long-range interactions is limited, and its performance is highly dependent on the quality and quantity of the training data [11].

2.3 Protein Structure Prediction Using Deep Learning

2.3.1 Convolutional Neural Networks

When handling low-similarity sequences, challenges often arise in achieving sufficient accuracy. Yagoubi et al. [6] proposed an innovative method based on Convolutional Neural Networks (CNN). This method uses the PSI-PRED tool to convert the amino acid sequence into the predicted secondary structure sequence, then encodes it into a binary matrix as the input data, and then uses 1D-CNN to automatically extract features from the input data and perform classification. Experimental results on multiple low-similarity datasets (25PDB, 640, 1189, and FC699) demonstrated that this model significantly improved prediction performance. However, the model cannot capture long-range dependencies or complex sequence information within protein sequences, which may affect the prediction of certain protein classes. Figure 6 shows the general schema of the proposed approach.

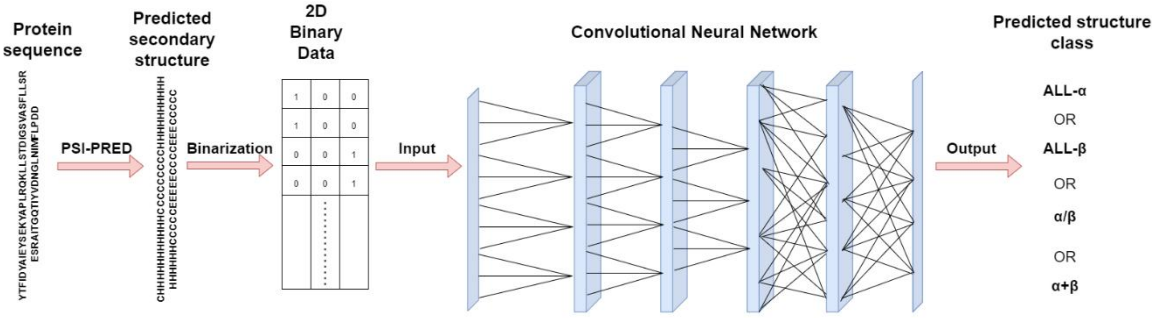


Figure 6 Yagoubi's schema of the approach [6]

2.3.2 Long Short-Term Memory

Wang et al. [12] proposes a protein secondary structure prediction method based on Long Short-Term Memory (LSTM). The method uses Position Specific Scoring Matrix (PSSM) to encode protein residues, transforming them into a two-dimensional data plane. It then designs three LSTM models—horizontal LSTM, vertical LSTM, and combined LSTM—to process different dimensional information of the PSSM matrix. Finally, an ensemble method integrates the prediction results of the three models,

achieving a Q3 accuracy of 77.9% on the CB513 dataset, significantly outperforming the performance of individual models. While this approach improves prediction performance by integrating the three LSTM models, it also increases the complexity of the model and may lead to overfitting issues. Figure 7 shows the combined LSTM model.

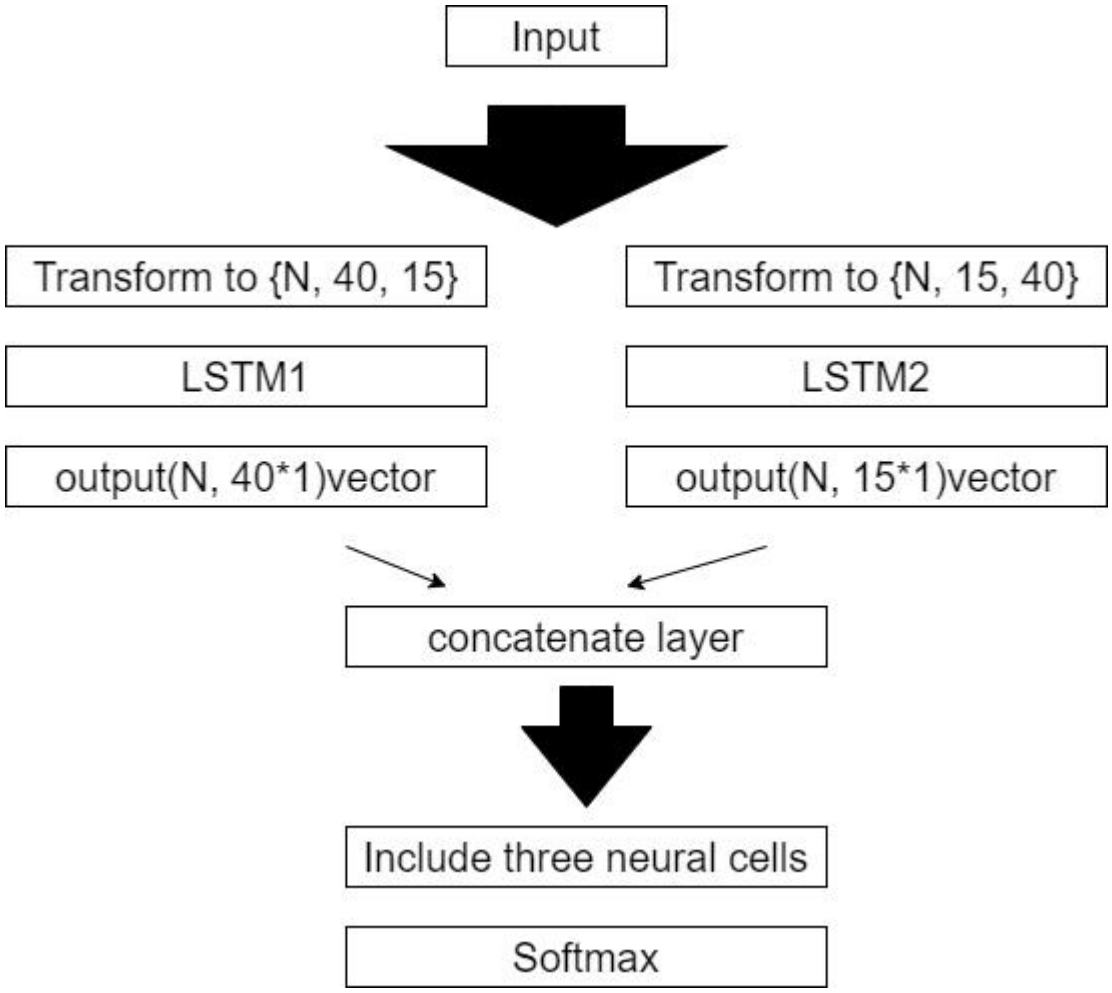


Figure 7 Wang's combined LSTM model [12]

2.3.3 Hybrid Convolutional Neural Networks and Long Short-Term Memory

Cheng et al. [13] proposes a protein secondary structure prediction method based on the integration of CNN and LSTM models. CNN is used to extract features from protein sequences, while LSTM is employed to capture long-range interaction features between amino acids. Cross-validation experiments on the 25PDB dataset show that the proposed model achieves an accuracy of 80.18%, outperforming the results of using a single model alone. Figure 8 shows the EN-CSLR model.

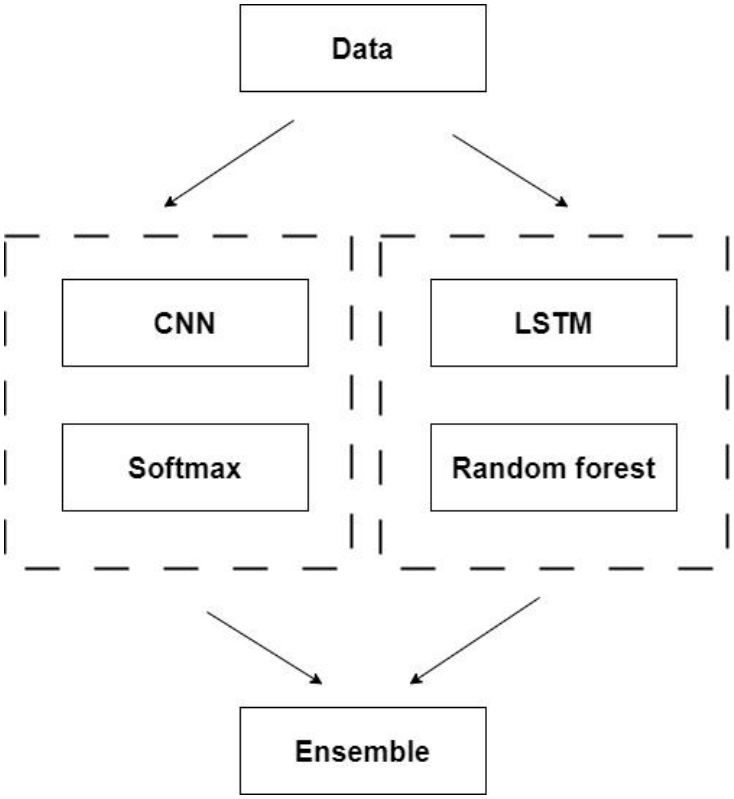


Figure 8 Cheng's EN-CSLR model [13]

A summary of the different researchers and their findings and possible results can be found in Table 1.

Table 1 Summary of background review

Author	Datasets	Methods & Models	Result	Limitation
Garnier and Robson [10]	Protein crystal structure data from the Brookhaven database	GOR method	After removing homologous proteins, the prediction accuracy rates of GOR I, GOR II and GOR III methods for the three states were 56.9%, 57.7% and	The sensitivity of the prediction method to homologous proteins requires the removal of the protein to be predicted from

			61.7% respectively. After merging the three methods, the accuracy rate has been improved to 65.5%.	the dataset to improve accuracy
Chelvi and Rangarajan [11]	SCOP database	Structural Concealed Markov Model (SCMM). Map the primary structure of proteins to their 2D fold.	Provide precise 2D protein folding with reduced sequence gaps of protein interactions.	SCMM assumes that the complex structure of a protein can be represented by the sequence of local structures, a simplification that does not fully capture the complexity of protein folding
Yagoubi et al. [6]	25PDB, 640, 1189 and FC699 datasets	Convolutional Neural Networks model	High prediction accuracy has been achieved on multiple low-similarity datasets, especially performing well in the α+β category.	The structure of the CNN model is rather complex, the training time is long, and it requires a large amount of computing resources.
Wang et al.	CB513 dataset	Long Short-Term	The ensemble	Only CB513

[12]		Memory networks model	method achieved a Q3 score of 77.9%, which is an improvement over the individual LSTM models (horizontal LSTM at 76.6%, vertical LSTM at 76.7%, and combined LSTM at 76.8%).	dataset is used, and the dataset is not representative
Cheng et al. [13]	25PDB	integrates Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks.	Features extracted from CNN and LSTM models can effectively improve the accuracy of protein secondary structure prediction.	The dataset is limited in size and under representative. The model is complex, the calculation cost and training time are high.

Chapter 3 Methodology

3.1 Approach

In this project, two datasets, namely Protein Secondary Structure and Protein Data Bank (PDB), are adopted. The models employ Convolutional Structural Predictor (CSP), Memory-based Structure Network and Self-Attention Convolutional Structural Predictor. The two datasets and the models used will be elaborated in detail.

3.2 Dataset

In this project, two independent datasets from Kaggle were utilized, namely Protein Secondary Structure and Protein Data Bank (PDB). The Protein Secondary Structure dataset was employed for model training, while the PDB dataset was used for evaluating the generalization capability of the model. The training dataset assisted the model in learning patterns within the data, whereas the validation dataset was used to test the model's predictive ability on new data, thereby verifying whether the model was overfitting and capable of generalizing to unseen data. This division of labor ensured the sufficiency of model training and the accuracy of evaluation.

3.2.1 Dataset 1 - Protein Secondary Structure dataset

Protein Secondary Structure dataset consists of three Comma-Separated Values (CSV) files: the basic dataset cb513, the training set and the validation set. These three files (cb513, training, validation) contain 513, 8678, and 2170 data samples, respectively. Each sample in the dataset consists of a peptide sequence (seq) and its corresponding secondary structure. The secondary structure information includes three-states labels (sst3) and eight-state labels (sst8), of which sst3 is divided into three categories: H (α-helix), E (β-sheet) and C (ring and irregular elements). The addition of B (β -bridge), G (3-helix), I (π -helix), T (Turn) and S (Bend) further subdivides the secondary structure into sst8. In addition, the protein sequence length of protein secondary structure dataset is concentrated in 20-1632. The diversity of sequence lengths enhances the robustness of the trained models.

3.2.2 Dataset 2 - Protein Data Bank (PDB) dataset

The PDB dataset contains 393,732 data samples. The structure of the sample is the same as that of dataset 1, consisting of three parts: seq, sst3 and sst8. However, the sequence length of PDB dataset covers 3-5037. Compared with dataset 1, this further expands the diversity of the data, which can better verify the generalization capability of the model.

Choosing these two datasets can cover more sequences, making the model prediction more accurate. These two datasets are crucial for understanding and predicting the secondary structure of proteins, and help to study the function and stability of proteins. Figure 9 shows examples of datasets, including sequences and their corresponding secondary structure.

Figure 9 Examples of Dataset

3.3 Data preprocessing

After the datasets are divided, the data undergoes several preprocessing steps to prepare it for deep learning model training. First, the complex eight-state labels (sst8) are simplified into three-state labels (sst3) through label mapping. This step reduces the complexity of the classification task. Next, the discrete amino acid sequences are converted into numerical tensors using one-hot encoding, which is essential for adapting the input sequences to the model's requirements. The three-state labels are also transformed into a one-hot format to facilitate the model's learning process. Finally, all sequences and labels are processed to ensure uniform length through padding or truncation, which is crucial for efficient batch training. These preprocessing steps provide a standardized input and output format, enabling the subsequent deep learning model training to proceed smoothly.

3.4 Data Split

3.4.1 Dataset 1 - Protein Secondary Structure dataset

The Protein Secondary Structure dataset consists of three Comma-Separated Values (CSV) files. These three files (cb513, training, validation) contain 513, 8678, and 2170 data samples, respectively. Due to the large difference in the amount of data in each file, in order to balance the data, the three files are first merged, and then the training set, validation set and test set are divided according to the proportion of 80%, 10% and 10%.

This ensures that the model has enough training data and prevent overfitting. The structure of the data separation is provided in Figure 10.

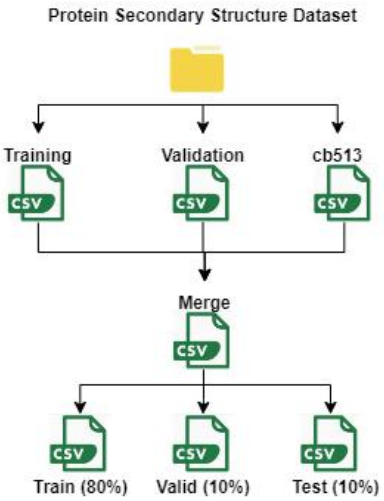


Figure 10 Protein Secondary Structure Dataset after Separation

3.4.2 Dataset 2 - Protein Data Bank (PDB) dataset

The PDB dataset contains 393,732 data samples, and the huge dataset increases the demand for computing resources, which is not conducive to the training of the model. Therefore, the PDB dataset is filtered. The length of peptide sequences in the PDB dataset ranged from 3-5037. Compared with the Protein Secondary Structure dataset, the PDB dataset contained more long sequences.

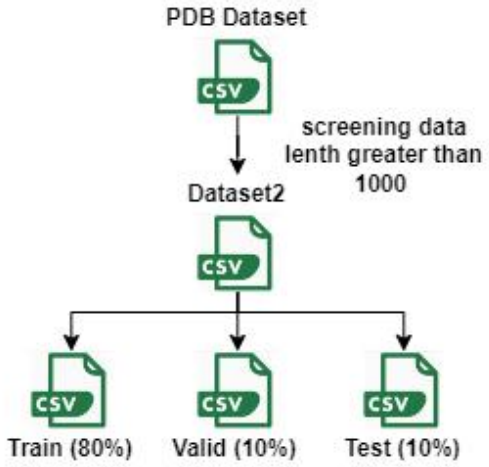


Figure 11 PDB Dataset after Separation

Therefore, in order to increase the diversity of the data, sequences with sequence lengths greater than 1000 in the PDB dataset were filtered. The filtered PDB dataset

contains 3511 sequences. Furthermore, delete the irrelevant columns in the PDB dataset to keep the dataset clean and concise. Finally, the filtered PDB dataset is divided into the training set, the validation set and the test set in the proportions of 80%, 10% and 10% respectively. The structure of data separation is shown in Figure 11.

3.5 Proposed Model Structure

3.5.1 Convolutional Structural Predictor (CSP)

The model primarily designed for classifying protein sequence features. The model utilizes a series of one-dimensional convolutional layers (Conv1D) to extract local patterns from sequences, with multiple convolutional kernels progressively capturing deeper features. Figure 12 shows the structure of CSP model.

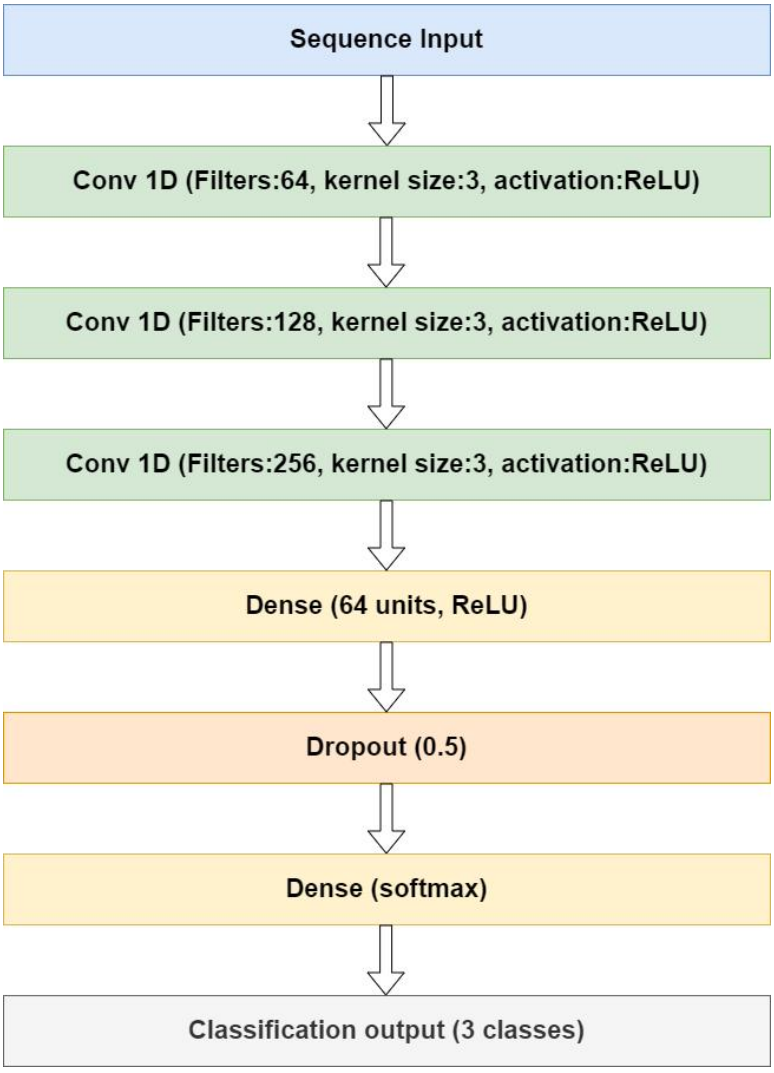


Figure 12 Convolutional Structural Predictor

And this is followed by fully connected layers and Dropout to enhance generalization, then a SoftMax layer to output the probability distribution of predicted classes. The Convolutional Structural Predictor model is lightweight, efficient, and excels at extracting local features, making it well-suited for protein secondary structure prediction.

3.5.2 Self-Attention Convolutional Structural Predictor (Attention-CSP)

This model incorporates the self-attention mechanism on the basis of the Convolutional Structure Prediction (CSP) model. After the convolutional layer, the self-attention mechanism is added to globally model the convolutional features. In the self-attention mechanism, the input sequence is first linearly transformed to generate Query (Q), Key (K) and Value (V). Then, through dot product, SoftMax normalization, and weighted summation, the output of the self-attention mechanism is obtained.

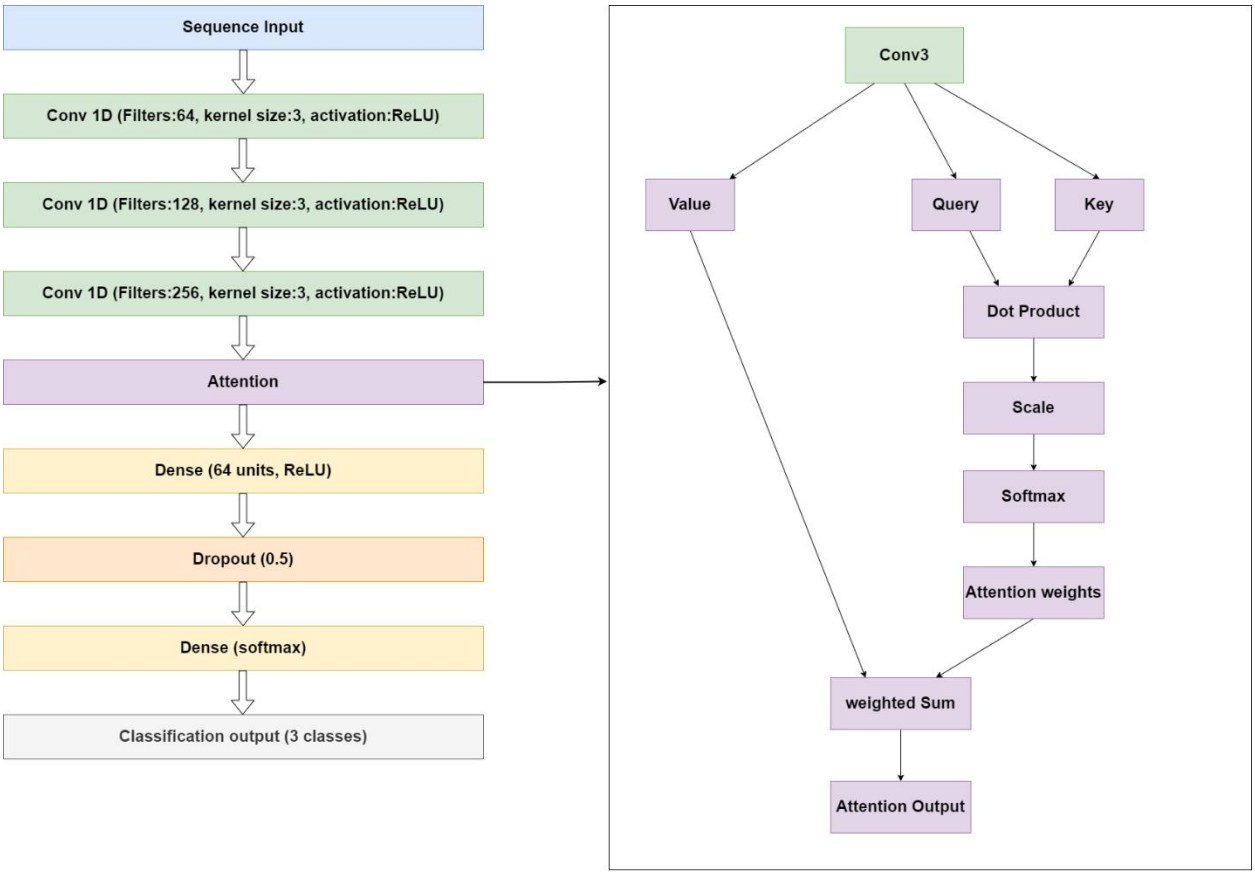


Figure 13 Self-Attention Convolutional Structural Predictor

This enables the calculation of the correlation between different positions in the sequence to dynamically adjust the importance of features, thereby capturing long-range dependencies. Next, the model further processes the features through the time

distribution fully connected layer and uses Dropout to improve the generalization ability of the model. Finally, the SoftMax output layer of the time distribution generates a probability distribution for each sequence position. This model efficiently extracts local features through CSP and enhances the ability to capture global information through the self-attention mechanism, making it suitable for protein secondary structure prediction. Figure 13 shows the model structure after adding self-attention.

3.5.3 Memory-based Structure Network

The main goal of this model is to learn patterns from the input time-series data and make classification predictions. The model is able to consider both forward and backward information in the sequence, better capturing long-range dependencies within the time series. Next, the model introduces non-linear transformations through a fully connected layer and uses a Dropout layer to prevent overfitting. The final output layer employs a SoftMax activation function for a three-class classification task. During training, the Adam optimizer is used to adjust the model's weights, and categorical cross-entropy is used as the loss function to optimize classification accuracy. Figure 14 shows the structure of Memory-based Structure Network model.

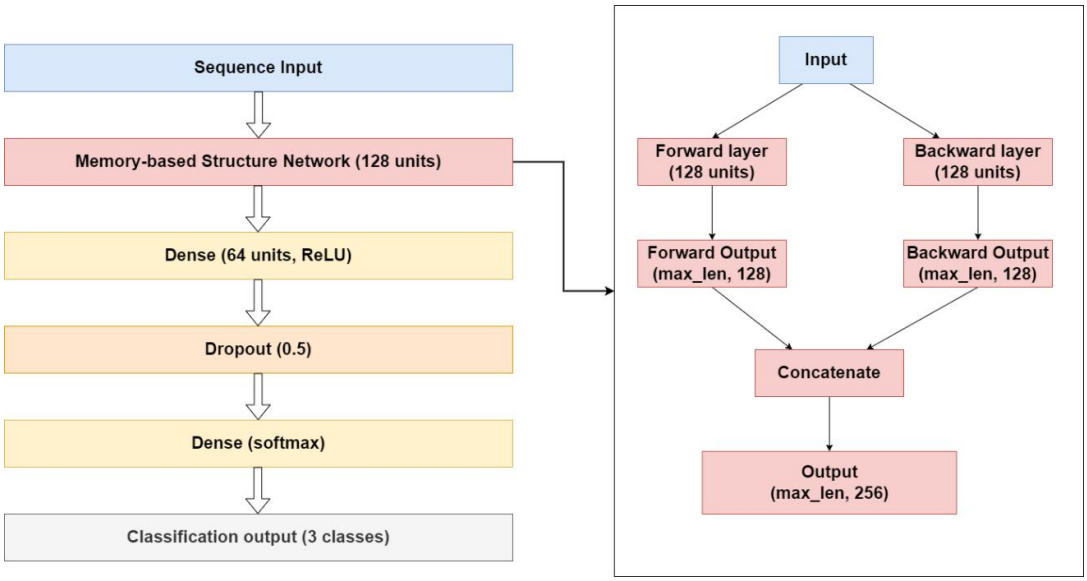


Figure 14 Memory-based Structure Network

3.5.4 Triple Fusion Explainable Model

This model integrates Convolutional Structural Predictor (CSP) model, Memory-based Structure Network model, and Self-Attention Convolutional Structural Predictor (Attention-CSP) model. Furthermore, SHAP was incorporated to explain the model's

results, and the contribution values of each amino acid to the model's prediction were calculated. Figure 15 shows the structure of triple fusion explainable model.

Figure 15 Triple Fusion Explainable Model

First, three 1D convolutional layers are employed to extract local features from the input sequence, with ReLU activation functions enhancing non-linear representation

capabilities while maintaining the same sequence length as the input. Next, a Memory-based Structure Network layer captures bidirectional dependencies within the sequence, generating time-step-level feature representations. Building on this, the Attention mechanism dynamically focuses on key feature regions, further improving the modeling of long-range dependencies. High-dimensional features are extracted through a dense layer, with Dropout applied to prevent overfitting. Finally, a SoftMax output layer classifies each time step into one of three secondary structure classes: alpha-helix (H), bata-sheet (E), and coil (C), producing probability distributions. In the SHAP section, force plot, summary plot, waterfall plot, dot plot and bar plot are selected to analyze the results.

3.6 Experimental Setup and Technology

The experimental setup consists of multiple parts. In terms of data, the dataset is divided into the training set, the validation set and the test set to ensure that the model is evaluated on unseen data. During the training process, it is set that the model can be trained for up to 50 epochs, and the size of each batch is 32. And it is trained using the training set, and the performance is evaluated on the validation set after each epoch. In addition, an early booting mechanism has been implemented. If the validation loss does not improve in ten consecutive epochs, the training will be stopped to prevent overfitting.

The technology this project will be using is displayed in Table 2.

Table 2 Summary of Relevant Technology involved in this project

Software	Framework	Tensorflow
	Language	Python
	Libraries	Numpy, Keras, Matplolib
Hardware	Central processing unit (CPU)	11th Gen Intel® CoreTM i7-1165G7 @ 2.80GHz
	Graphic Processing Unit (GPU)	NVIDIA GeForce MX450 2GB

3.7 Evaluation Metrics

This project uses eight evaluation metrics: Loss function, Confusion matrix, Accuracy, Precision, Recall/Sensitivity, F1-score, Specificity, and ROC Curve. Evaluation metrics are mainly used to measure the performance of the model and can help understand the

model's performance on training and validation data. The results of the evaluation metrics can guide the optimization and selection of the model. The following will provide a detailed introduction to each evaluation metrics.

3.7.1 Loss Function

The model uses categorical_crossentropy as the loss function, which is one of the most commonly used loss functions in multi-class classification tasks. It optimizes the model's classification performance by measuring the cross-entropy loss between the predicted probability distribution and the actual one-hot encoded labels.

$$Loss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C}[\, y_{true}, i, j * \log\left(y_{pred}, i, j\right)]\#\#\#\# \quad (1)$$

3.7.2 Confusion matrix

The confusion matrix presents a detailed distribution of the model's prediction results in matrix form, including the specific cases of correct and incorrect classifications, helping to gain deeper insights into the model's performance. The confusion matrix contains four parts. True Positive (TP) is the number of positive samples correctly predicted as positive (correctly classified positive samples). True Negative (TN) is the number of negative samples correctly predicted as negative (correctly classified negative samples). False Positive (FP) is the number of negative samples incorrectly predicted as positive (incorrectly classified as positive negative samples), also known as a false positive or Type I Error. False Negative (FN) is the number of positive samples incorrectly predicted as negative (incorrectly classified as negative positive samples), also known as a false negative or Type II Error. Figure 16 shows the basic style of the confusion matrix.

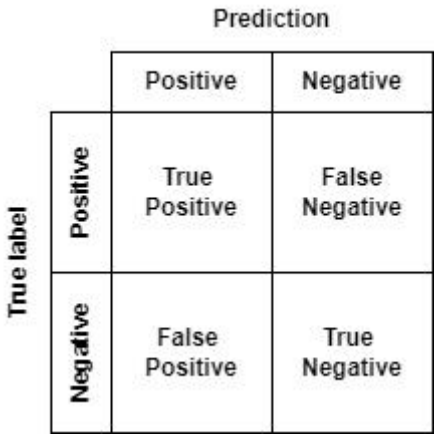


Figure 16 Basic style of the confusion matrix

3.7.3 Accuracy

Accuracy measures the proportion of correctly predicted samples out of the total samples.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \quad (2)$$

3.7.4 Precision

Precision represents the proportion of true positive samples among all the samples predicted as positive by the model.

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

3.7.5 Recall/ Sensitivity

Recall represents the proportion of actual positive samples that are correctly predicted as positive by the model.

$$Recall = \frac{TP}{TP+FN} \quad (4)$$

3.7.6 F1-score

F1-score is the harmonic mean of Precision and Recall, used to balance the importance of both.

$$F1 = 2 * \frac{Precision*Recall}{Precision+Recall} \quad (5)$$

3.7.7 Specificity

Specificity represents the proportion of actual negative samples that are correctly predicted as negative by the model.

$$Specificity = \frac{TN}{TN + FP} \quad (6)$$

3.7.8 ROC Curve

The ROC curve is a tool used to evaluate the performance of a binary classification model by adjusting the classification threshold. It shows the model's performance at different thresholds.

$$AUC = \int_0^1 TPR(FPR)d(FPR) \quad (7)$$

True Positive Rate (TPR), also known as Recall, which indicates the model's ability to correctly identify positive class samples.

The vertical axis (y) $TPR = \frac{TP}{TP+EN}$ (8)

The False Positive Rate (FPR) refers to the probability that the model incorrectly classifies a negative sample as positive.

The horizontal axis (x) $FPR = \frac{TP}{TP+FN}$ (9)

Chapter 4 Implementation and Result Analysis

To evaluate the accuracy and stability of the model in the study, the Protein Secondary Structure dataset and PDB dataset are used. In order to compare the different models, 7 experiments are designed in the project. Firstly, the results of using the protein secondary structure dataset on different models will be introduced to verify the predictive ability of the models for the classification of protein secondary structures. Then, the results of the PDB dataset on different models are introduced to prove the generalization capability of the model.

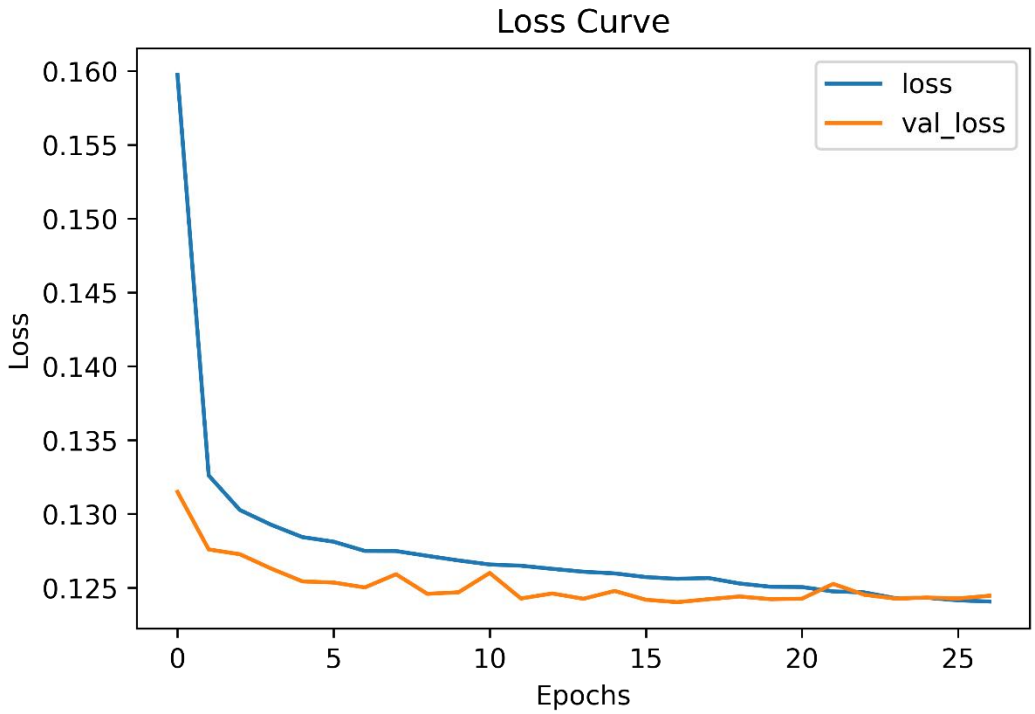
4.1 Convolutional Structural Predictor (CSP)

In the model implementation, local features were extracted through three convolutional layers (with the number of filters being 64, 128, and 256 respectively). Then, a 64-unit fully connected layer with ReLU activation was used to further process the features, and a 0.5 Dropout layer was added to prevent overfitting. Ultimately, the output layer uses the SoftMax activation function to output the probabilities of the three secondary structure categories. During the model training, the model is trained for a maximum of 50 epochs, with each batch size being 32, and an early stop mechanism is adopted to monitor and verify the loss.

4.1.1 Protein Secondary Structure dataset experimental result using Convolutional Structural Predictor

a) Loss

According to Figure 17, the initial rapid decrease in loss indicates that the model is learning effectively. Subsequently, the loss continues to decrease and gradually stabilizes, while the validation loss, despite some fluctuations, remains overall stable, demonstrating good performance on both the training and validation sets. Ultimately, both losses reach low values, indicating that the model has been trained effectively.

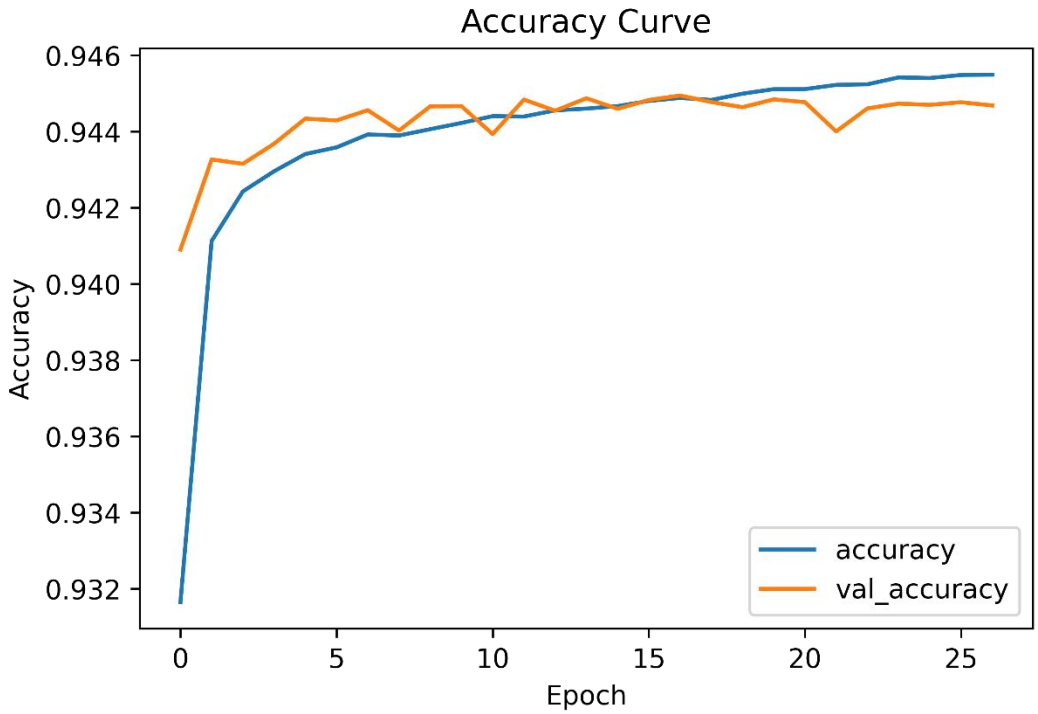


Train Loss = 0.1241, Valid Loss = 0.1245

Figure 17 Loss Curve of the Protein Secondary Structure dataset on CSP model

b) Accuracy

According to Figure 18, both the training accuracy and the validation accuracy increase with the increase of training rounds and eventually tend to be stable, indicating that the model has learned effective feature representations on both the training set and the validation set. The validation accuracy is slightly lower than the training accuracy, but the difference is not significant, indicating that there is no obvious overfitting phenomenon in the model.



Train Acc = 0.9455, Valid Acc = 0.9447

Figure 18 Accuracy Curve of the Protein Secondary Structure dataset on CSP model

c) ROC Curve

The Figure 19 ROC curve is close to the upper left corner, and the AUC value of each category is close to 1, indicating that the classification performance of the model in each category is very good.

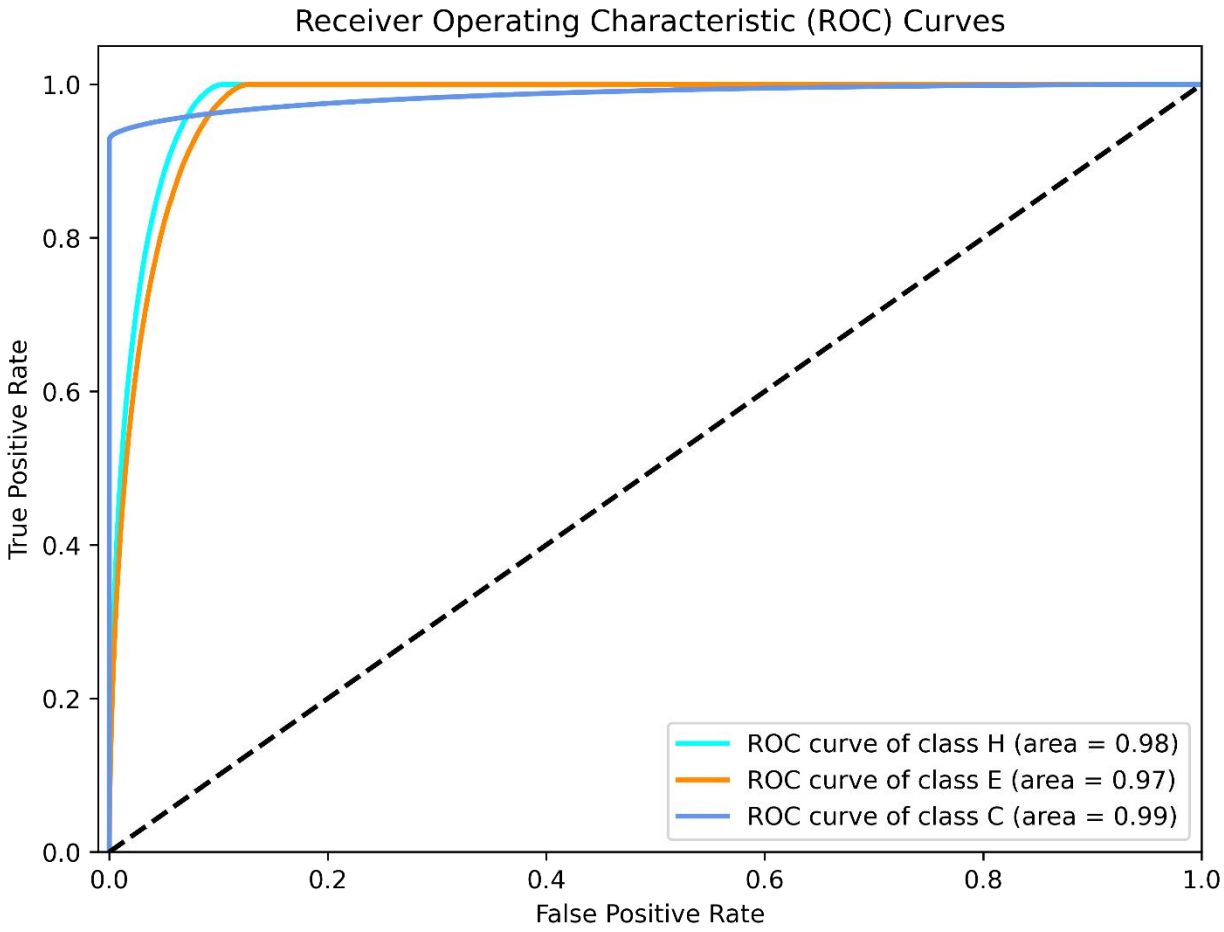
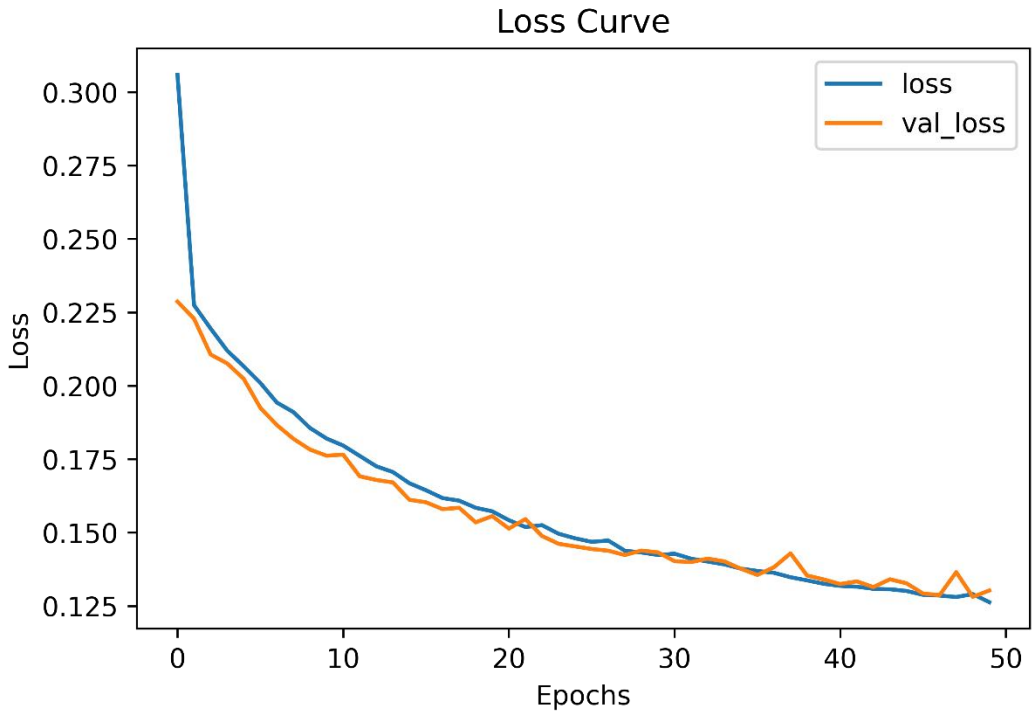


Figure 19 ROC Curve of the Protein Secondary Structure dataset on CSP model

4.1.2 PDB dataset experimental result using Convolutional Structural Predictor

a) Loss

In Figure 20, both the training loss and the validation loss decreased with the increase of training rounds and eventually tended to be stable, further verifying the effectiveness of the model on the PDB dataset. The validation loss is slightly higher than the training loss, and with a slight fluctuation, but the curves of the two are close, indicating that the model is not overfitted on the PDB dataset.

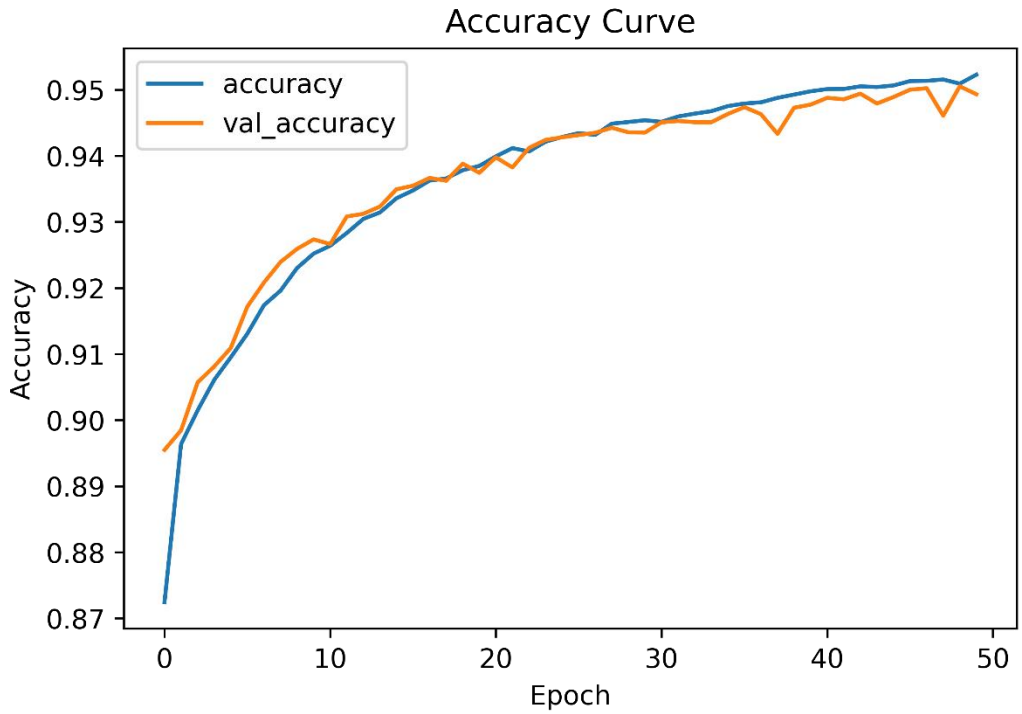


Train Loss = 0.1263, Valid Loss = 0.1302

Figure 20 Loss Curve of the PDB dataset on CSP model

b) Accuracy

Figure 21 shows the training accuracy and the validation accuracy rate also increase with the increase of training rounds and eventually tend to be stable, indicating that the model also has a good learning ability on the PDB dataset. The validation accuracy is close to the training accuracy, further proving the generalization capability of the model.



Train Acc = 0.9523, Valid Acc = 0.9574

Figure 21 Accuracy Curve of the PDB dataset on CSP model

c) ROC Curve

Figure 22, ROC curve is close to the upper left corner, and the AUC values are all equal to 0.99, indicating that the classification performance of the model on each category of the PDB dataset is also very good.

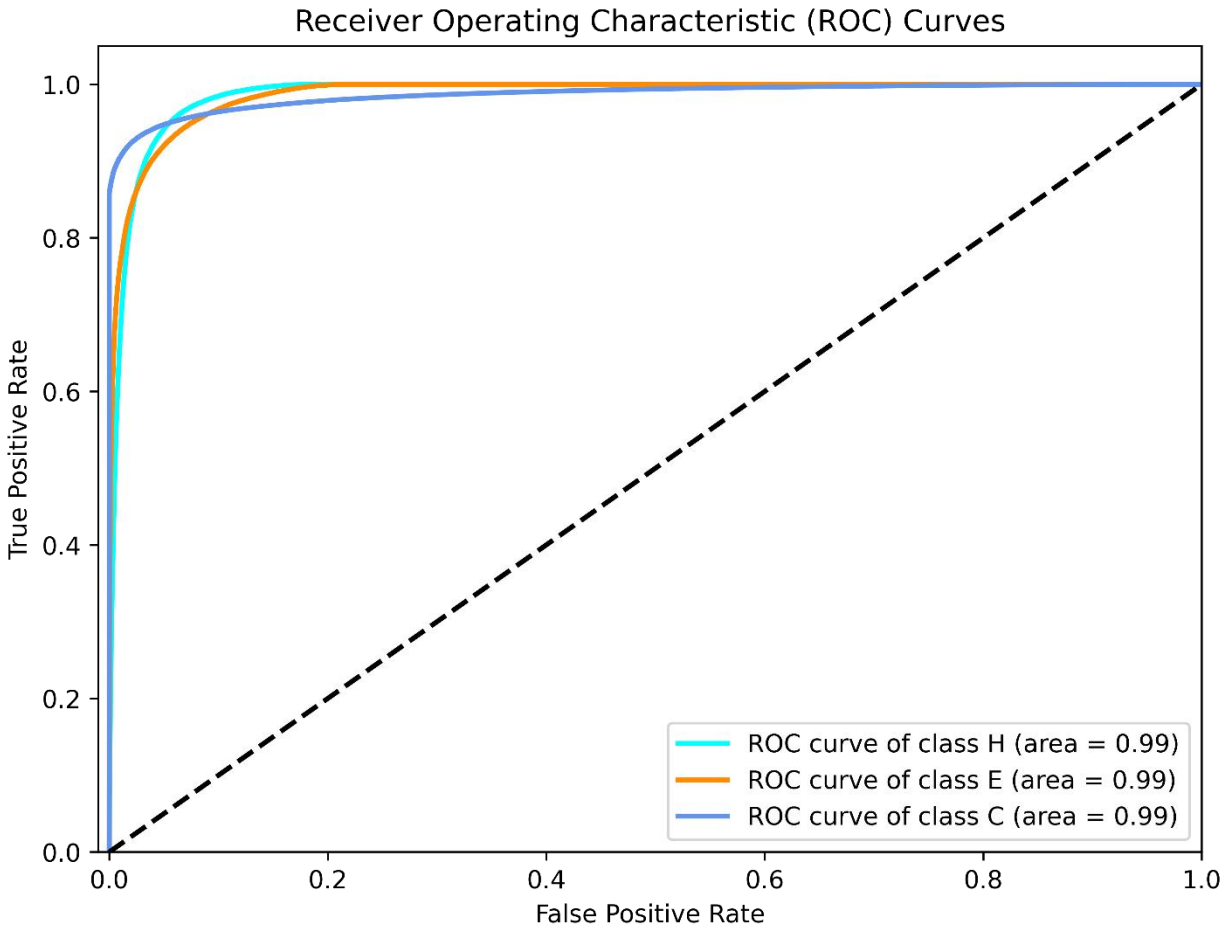


Figure 22 ROC Curve of the PDB dataset on CSP model

4.2 Self-Attention Convolutional Structural Predictor

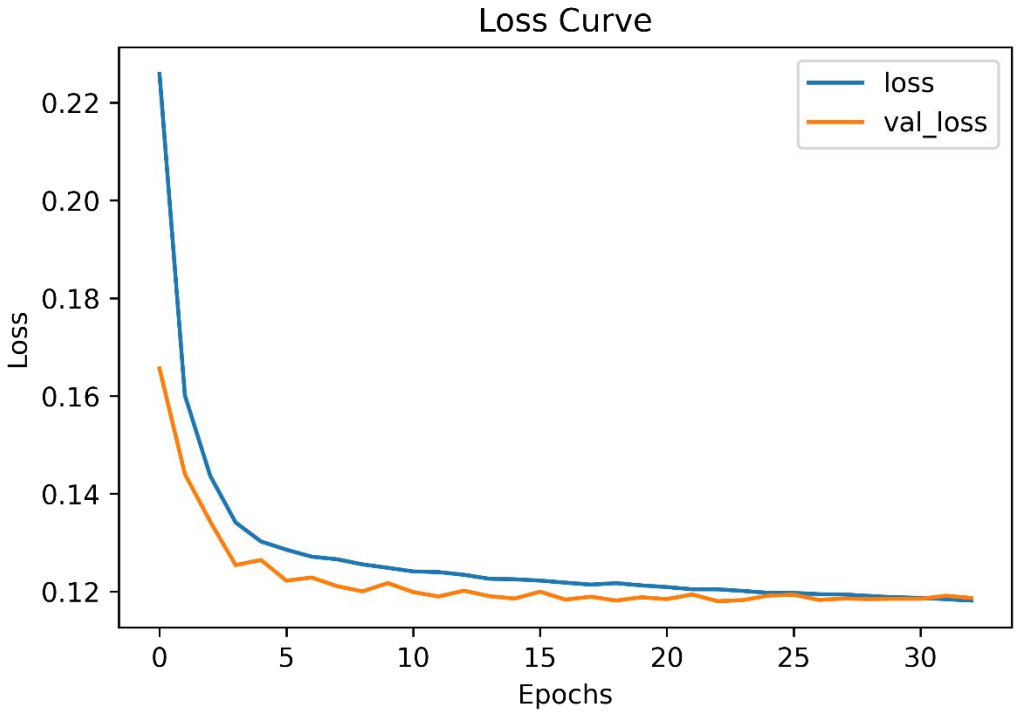
Based on the good performance of the Convolutional Structural Predictor model in predicting the secondary structure of proteins, the Self-Attention Convolutional Structural Predictor model integrates the self-attention mechanism. To enhance the model's ability to capture remote dependencies and global context in sequential data

4.2.1 Protein Secondary Structure dataset experimental result using Self-Attention Convolutional Structural Predictor

a) Loss

According to the Figure 23, the model's performance on the training set continues to improve, while its performance on the validation set also shows improvement, albeit with some fluctuations. Ultimately, both the training loss and validation loss reach low values,

and the two almost coincided, indicating that the model performs well on both the training and validation sets without significant overfitting.

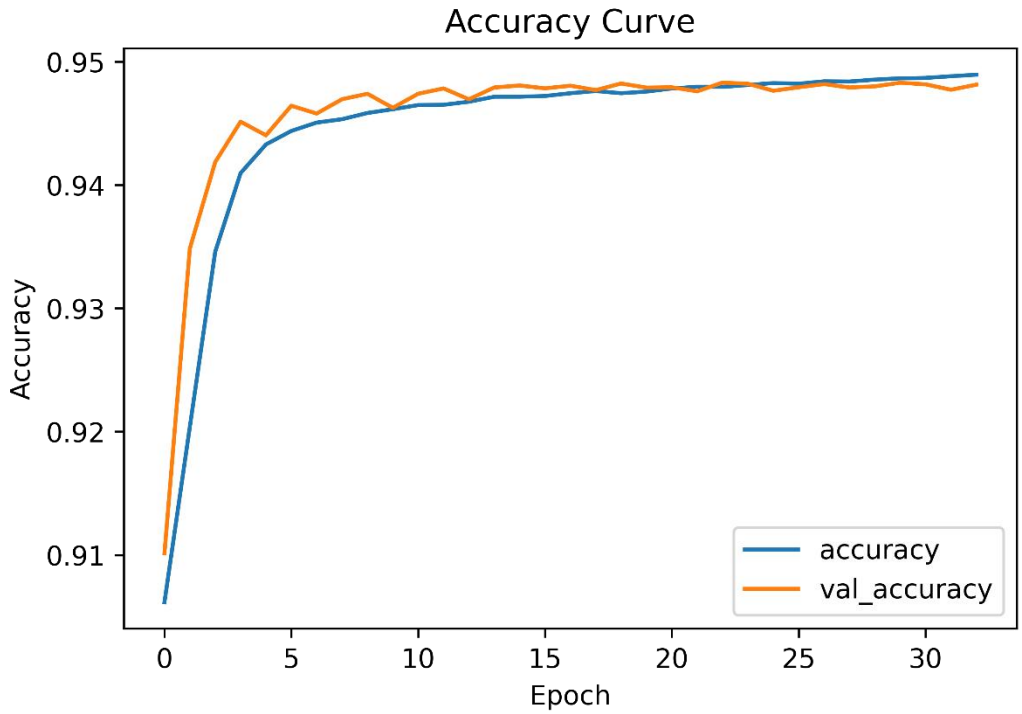


Train Loss = 0.1181, Valid Loss = 0.1187

Figure 23 Loss Curve of the Protein Secondary Structure dataset on Attention-CSP model

b) Accuracy

Figure 24, in the early stages of training, both metrics increase rapidly, indicating that the model is quickly learning and improving its classification capability. As training progresses, the training accuracy and validation accuracy stabilize, with a small gap between them, suggesting that the model performs well on both the training and validation sets without significant overfitting. Ultimately, both training and validation accuracies reach high levels.



Train Acc = 0.9489, Valid Acc = 0.9481

Figure 24 Accuracy Curve of the Protein Secondary Structure dataset on Attention-CSP model

c) ROC Curve

Following Figure 25, the Area Under the Curve (AUC) for class H and class E are both 0.98, while the AUC for class C is 0.99. This indicates that the model demonstrates excellent classification performance across all three classes, with its best performance observed in class C.

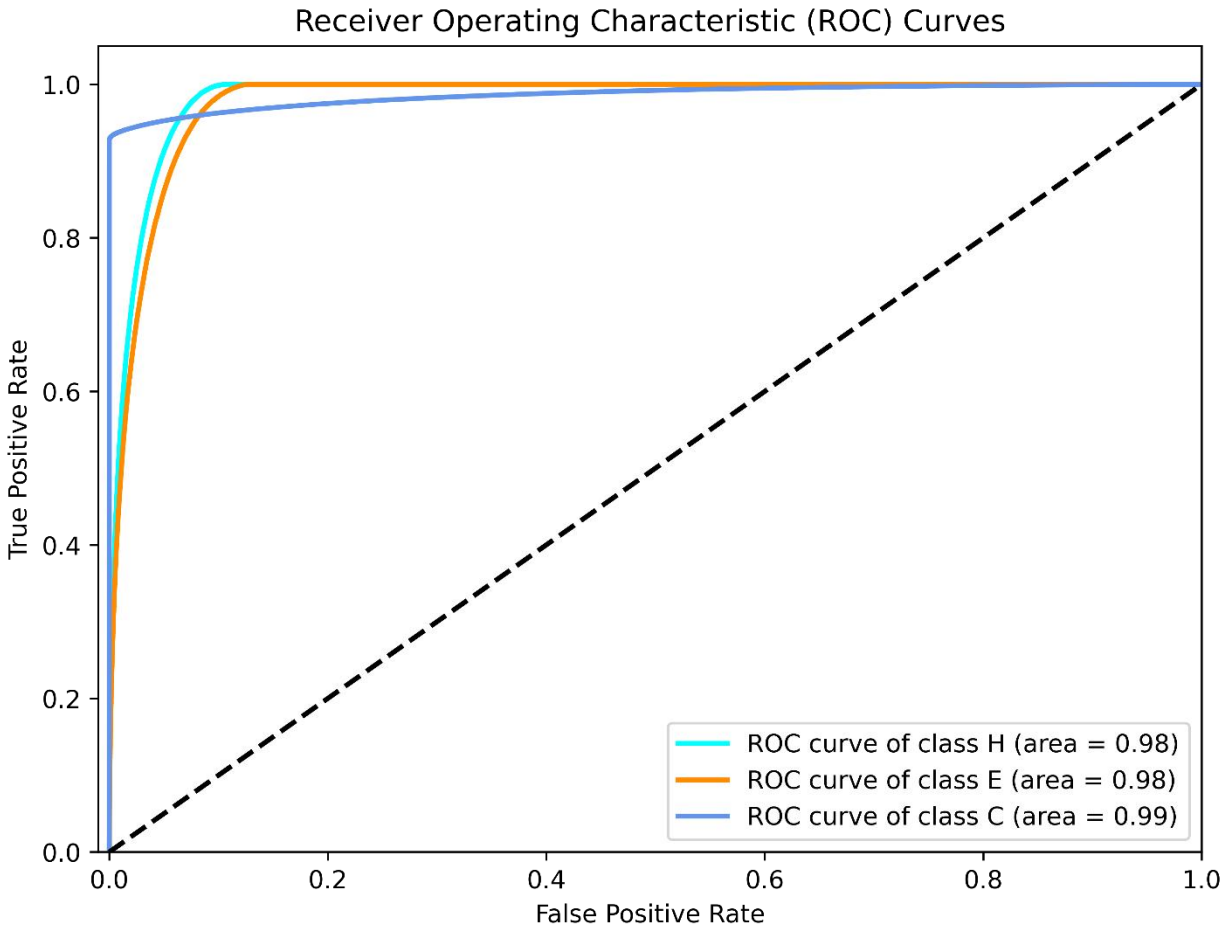


Figure 25 ROC Curve of the Protein Secondary Structure dataset on Attention-CSP model

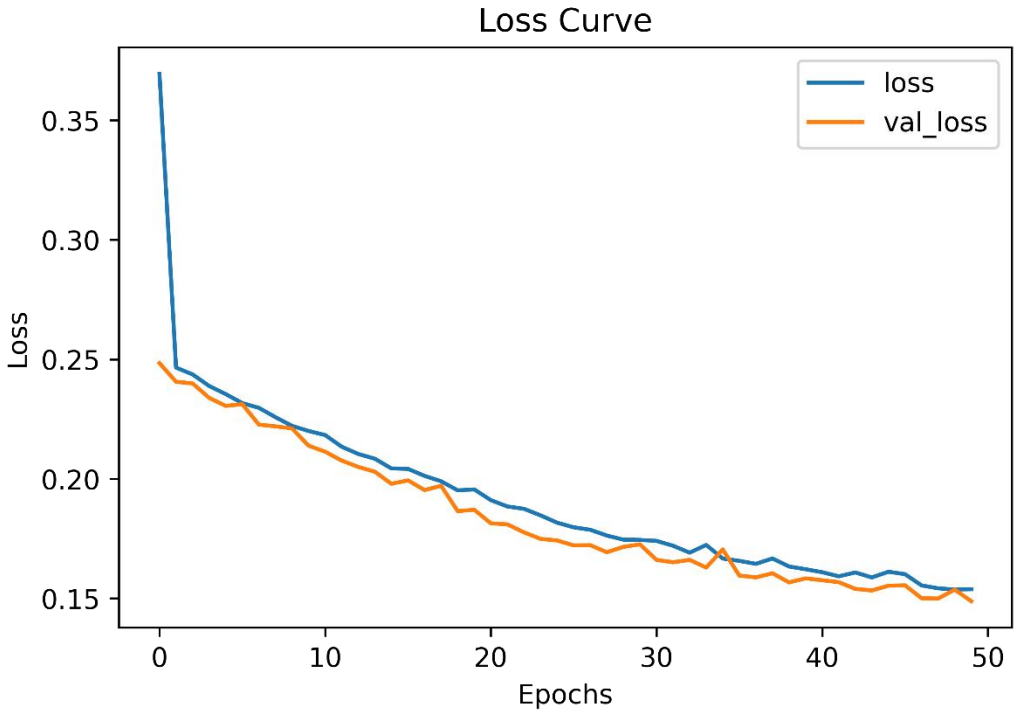
Compared with the Convolutional Structural Predictor model, after adding the self-attention mechanism, the loss values of both the training set and the validation set have decreased, and the accuracy has increased. This further indicates that the self-attention mechanism improves the performance of the model

4.2.2 PDB dataset experimental result using Self-Attention Convolutional Structural Predictor

a) Loss

According to Figure 26, in the initial stage, both the training loss and the validation loss were relatively high. As the training progressed, both decreased significantly, indicating that the model is effectively learning and reducing the prediction error. In the later stage of training, the loss value tends to be stable, and the training loss and validation loss

curves are close, indicating that the model performs consistently on the training set and the validation set, and there is no obvious overfitting phenomenon

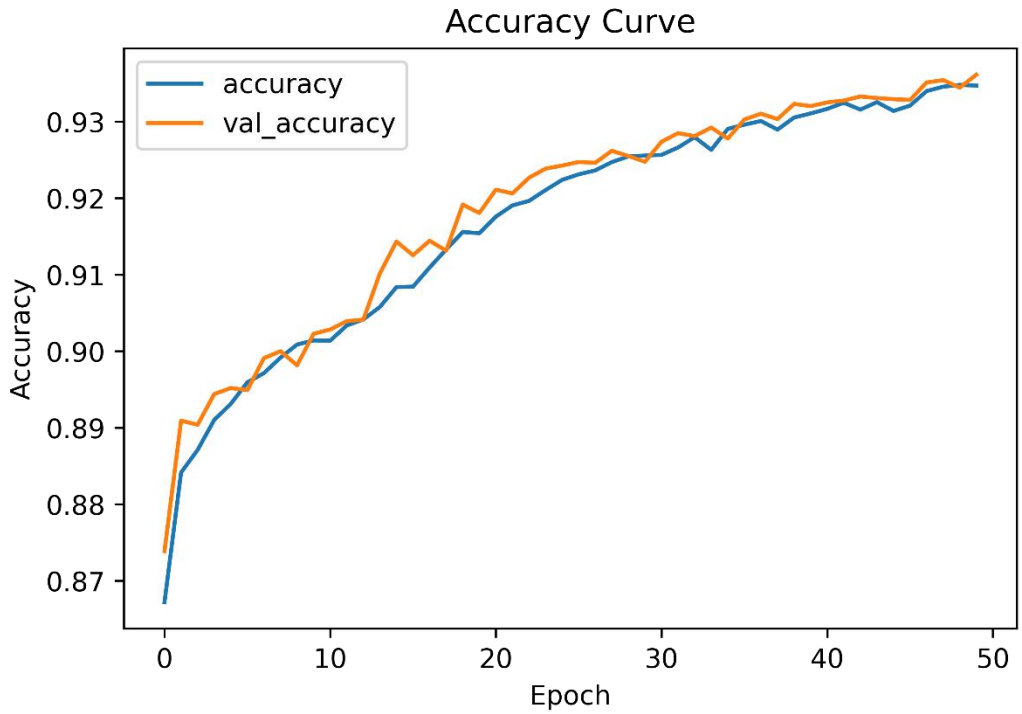


Train Loss = 0.1539, Valid Loss = 0.1489

Figure 26 Loss Curve of the PDB dataset on Attention-CSP model

b) Accuracy

For figure 27, as training progresses, both training accuracy and validation accuracy stabilize, with a small gap between them, indicating that the model's performance is improving on both the training and validation sets without significant overfitting. Ultimately, both training and validation accuracies reach high levels, demonstrating that the model has good generalization ability. Although the validation accuracy shows fluctuations at certain points, it remains generally consistent with the training accuracy overall.



Train Acc = 0.9347, Valid Acc = 0.9361

Figure 27 Accuracy Curve of the PDB dataset on Attention-CSP model

c) ROC Curve

According to Figure 28, the Area Under the Curve (AUC) of the ROC for three classes are all 0.98. This indicates that the model demonstrates high classification performance across all three classes.

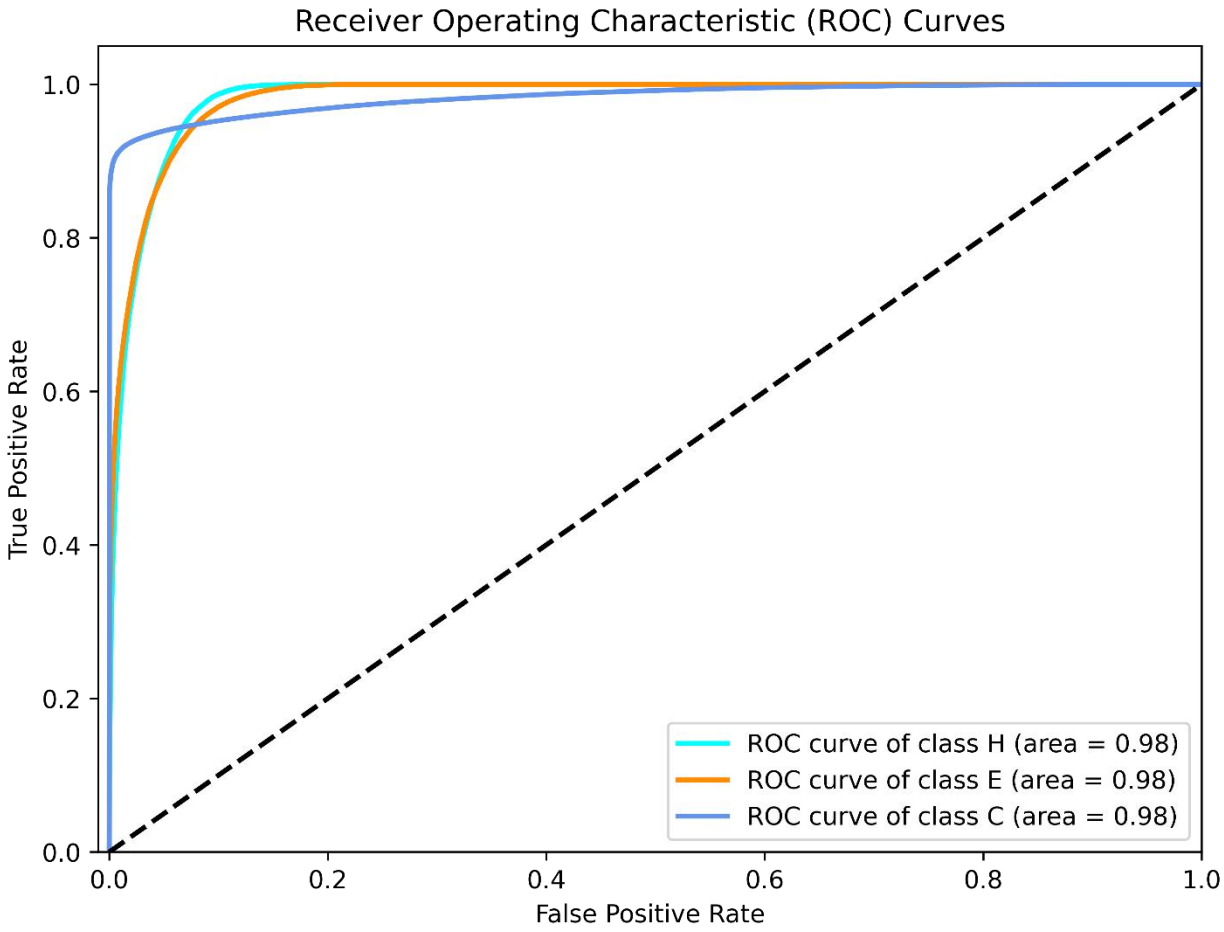


Figure 28 ROC Curve of the PDB dataset on Attention-CSP model

The various evaluation indicators of the PDB dataset in the Self-Attention Convolutional Structural Predictor have not changed significantly, indicating that this model has a high ability of accurate prediction and generalization

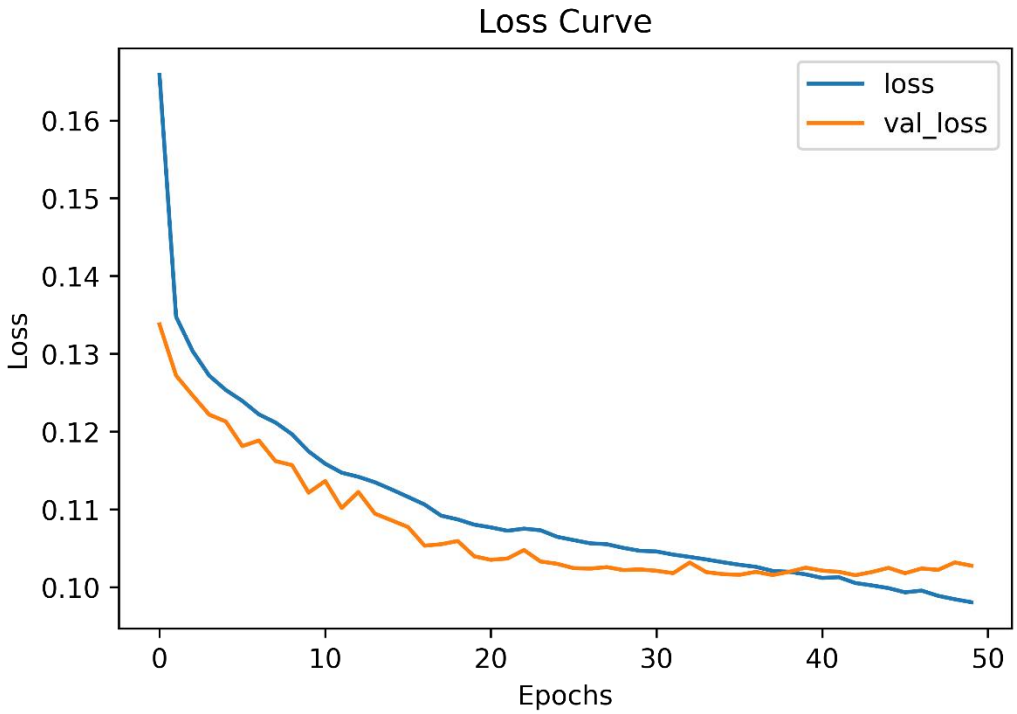
4.3 Memory-based Structure Network

This part will verify Memory-based Structure Network has a positive effect on the prediction of protein secondary structure.

4.3.1 Protein Secondary Structure dataset experimental result using Memory-based Structure Network

a) Loss

Figure 29 shows with the increase of training rounds, both the training loss and the validation loss showed a significant downward trend and eventually decreased to a relatively low and stable level, indicating that the model training effect was good.

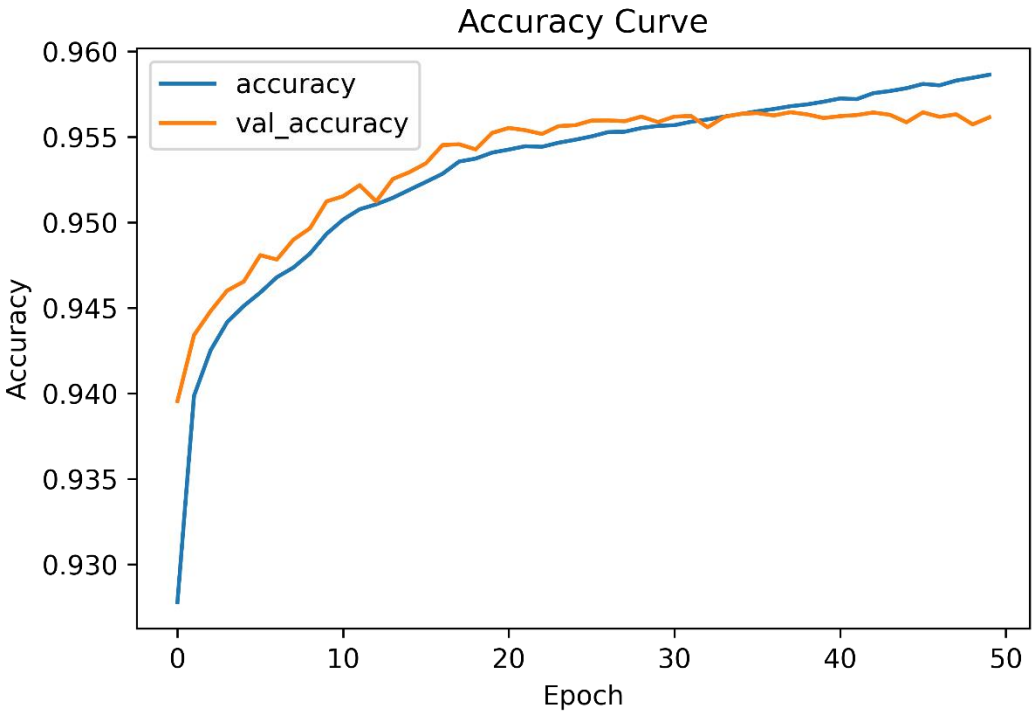


Train Loss = 0.0981, Valid Loss = 0.1028

Figure 29 Loss Curve of the Protein Secondary Structure dataset on Memory-based structure network model

b) Accuracy

Figure 30 indicates that both the training accuracy and the validation accuracy show an upward trend, which suggests that the model is learning and gradually improving its predictive ability. The accuracy rate of the final training set is higher than that of the validation set, but the difference between the two is not significant. The model has no obvious overfitting problem, indicating that the model has high accuracy and stability



Train Acc = 0.9586, Valid Acc = 0.9561

Figure 30 Accuracy Curve of the Protein Secondary Structure dataset on Memory-based structure network model

c) ROC Curve

The AUC values (Figure 31) for all categories are very high, close to 1, indicating that the model performs very well in distinguishing each category.

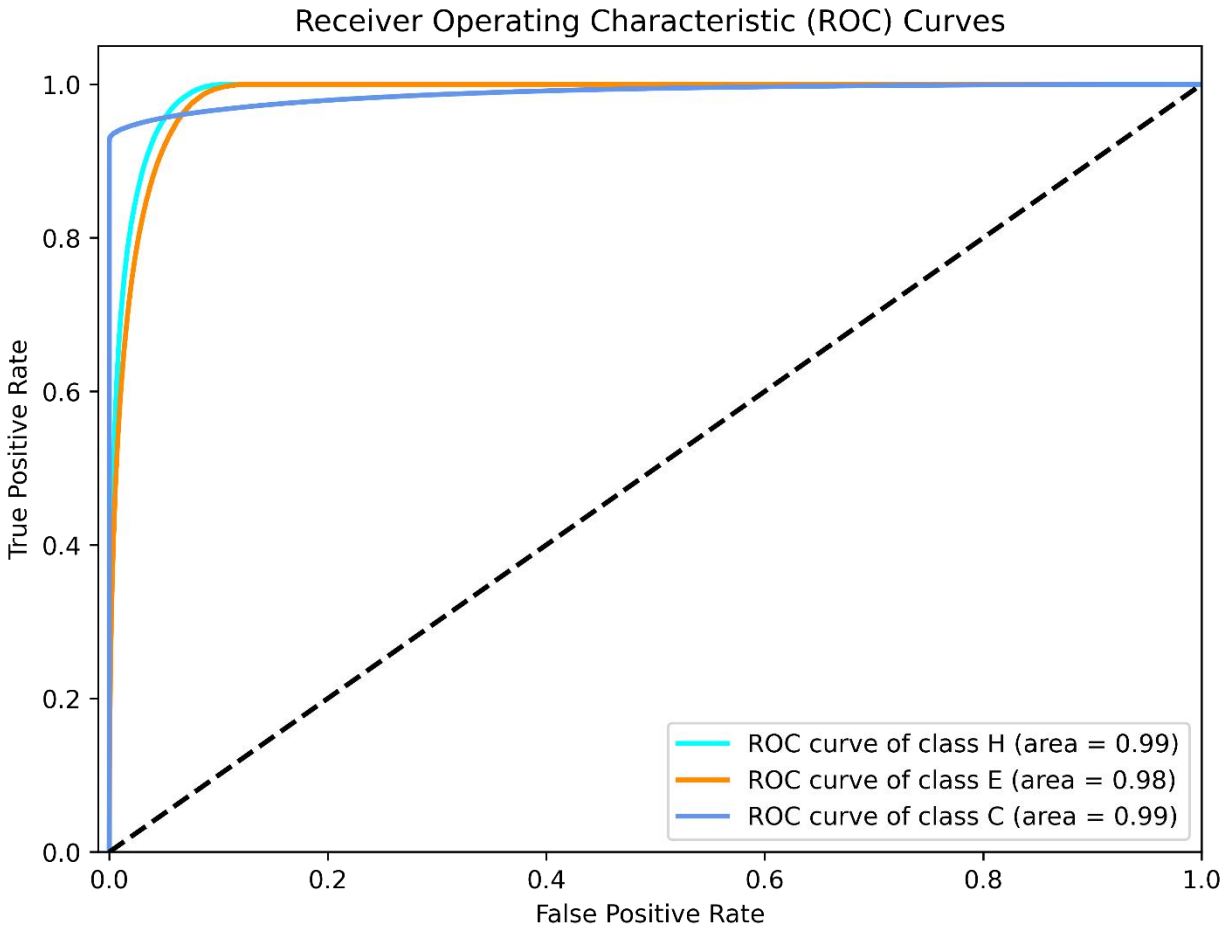


Figure 31 ROC Curve of the Protein Secondary Structure dataset on Memory-based structure network model

By comprehensively analyzing various indicators, the Memory-based Structure Network model also has high accuracy in the prediction of protein secondary structure

4.4 Triple Fusion Explainable Model

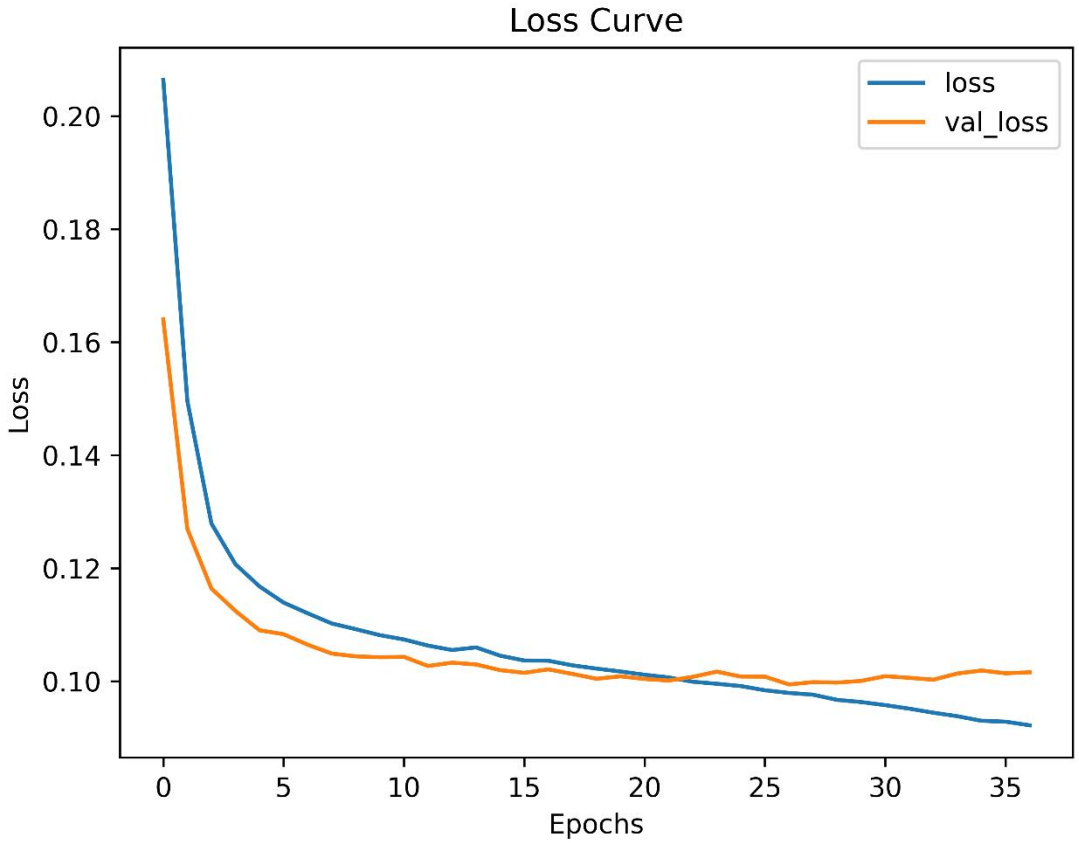
The analysis shows that the previous single model has high accuracy and stability in predicting the secondary structure of proteins. This model will integrate individual models to analyze the effect of combining the use of Convolutional Structural Predictor (CSP), Memory-based Structure Network, and Self-Attention Convolutional Structural Predictor (Attention-CSP), in predicting the secondary structure of proteins. By integrating these models, this method aims to leverage the advantages of each model: CSP is used for automatic feature extraction of protein sequences, Attention-CSP is used to focus on the important parts of the sequence, and Memory-based Structure Network is used to capture long-term dependencies and sequence patterns. This

combination is expected to enhance the model's ability to accurately predict the secondary structure of proteins by considering local and global sequence information.

4.4.1 Protein Secondary Structure dataset experimental result using Triple Fusion Explainable Model

a) Loss

According to Figure 32, both curves show that the loss value decreases with the increase of training rounds, indicating that the model performance is improving. The training loss and validation loss curves are close, indicating that the model is not overfitted

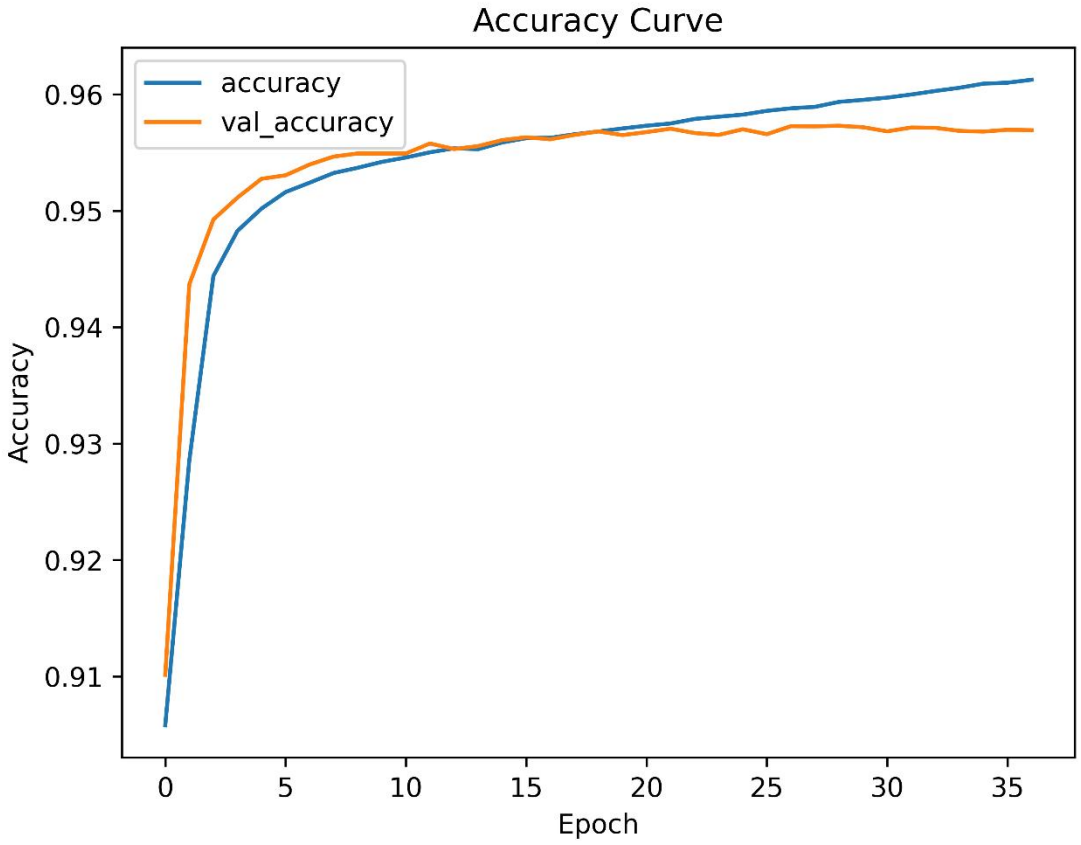


Train Loss = 0.0926, Valid Loss = 0.1016

Figure 32 Loss Curve of the Protein Secondary Structure dataset on Triple Fusion Explainable model

b) Accuracy

According to Figure 33, the accuracy increases rapidly during the initial epochs, indicating that the model is quickly improving its predictive capability. After a few epochs, the training accuracy continues to rise gradually, while the validation accuracy stabilizes around 0.955. The small gap between training and validation accuracy further confirms that the model does not exhibit overfitting.



Train Acc = 0.9610, Valid Acc = 0.9569

Figure 33 Accuracy Curve of the Protein Secondary Structure dataset on Triple Fusion Explainable model

c) ROC Curve

The ROC curves (Figure 34) for the three categories (H, E, C) all have AUC values of 0.99, indicating that the model achieves excellent classification performance across all categories.

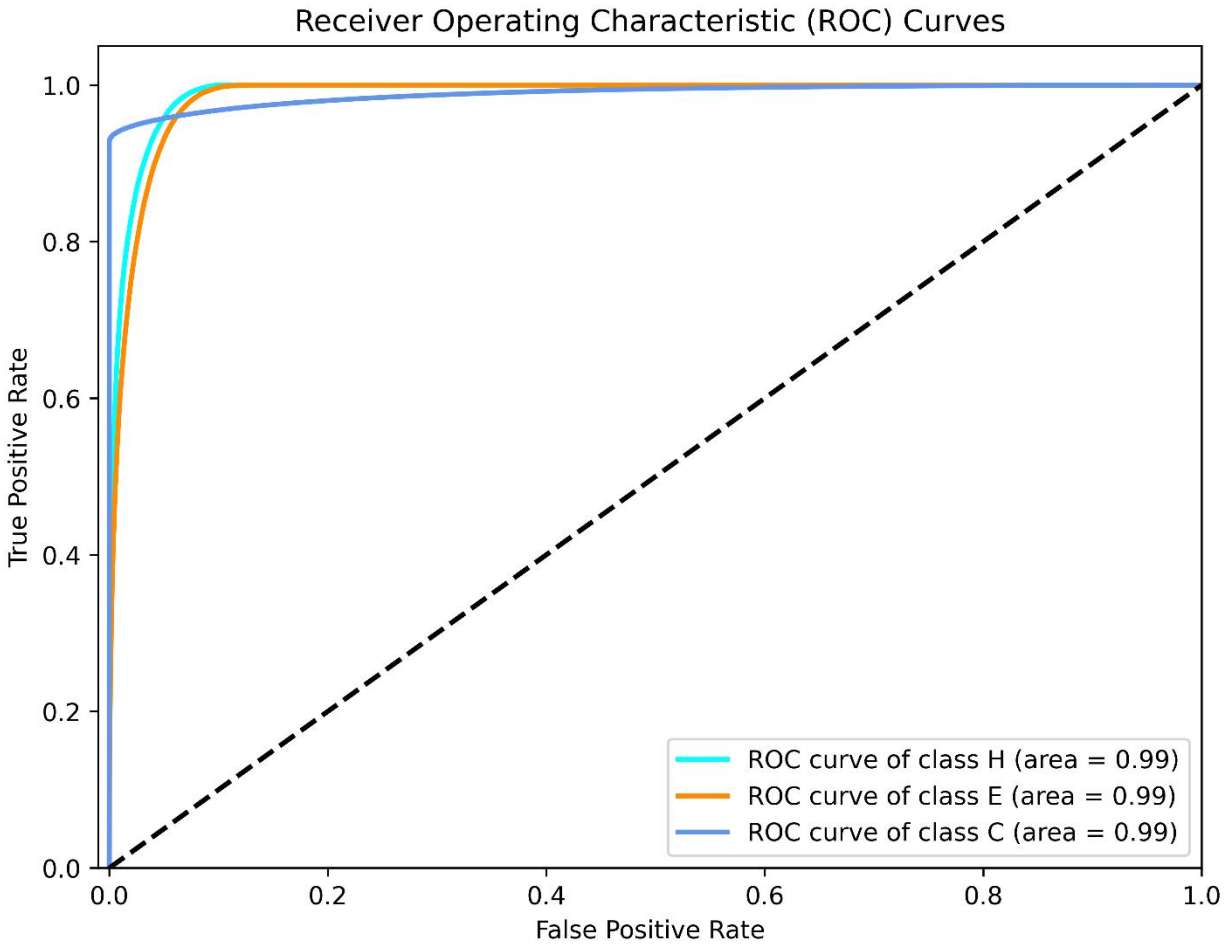
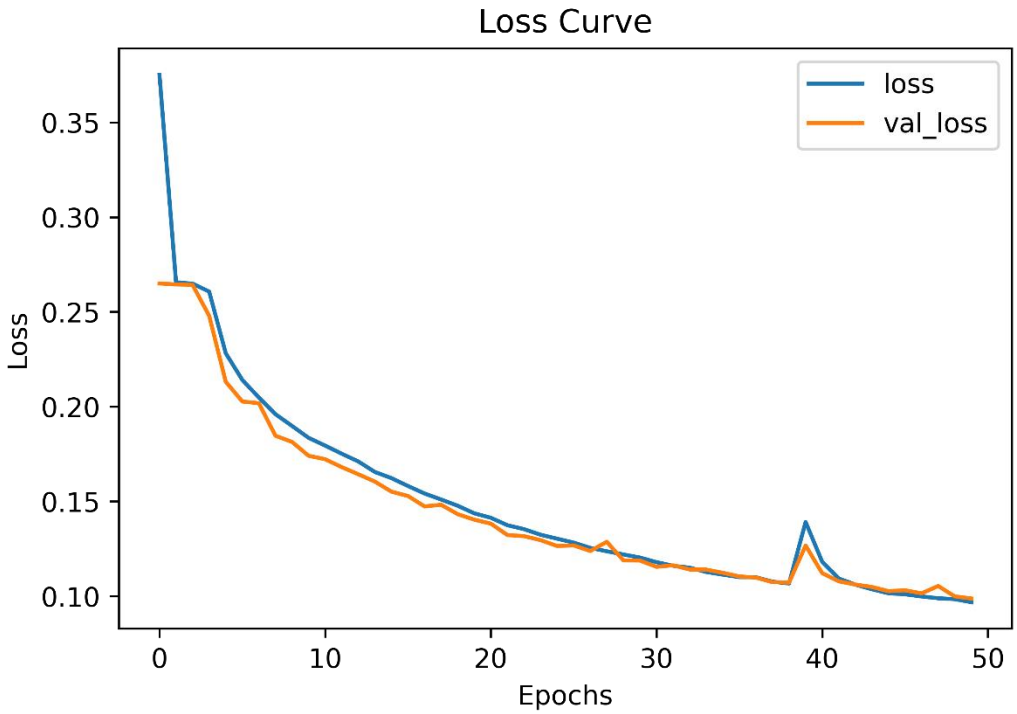


Figure 34 ROC Curve of the Protein Secondary Structure dataset on Triple Fusion Explainable model

4.4.2 PDB dataset experimental result using Triple Fusion Explainable Model

a) Loss

Figure 35 shows that both loss curves are in a downward trend, indicating that the model performance is improving. However, in the later stage of training, the validation loss increased, indicating the risk of overfitting, and the model can be further optimized

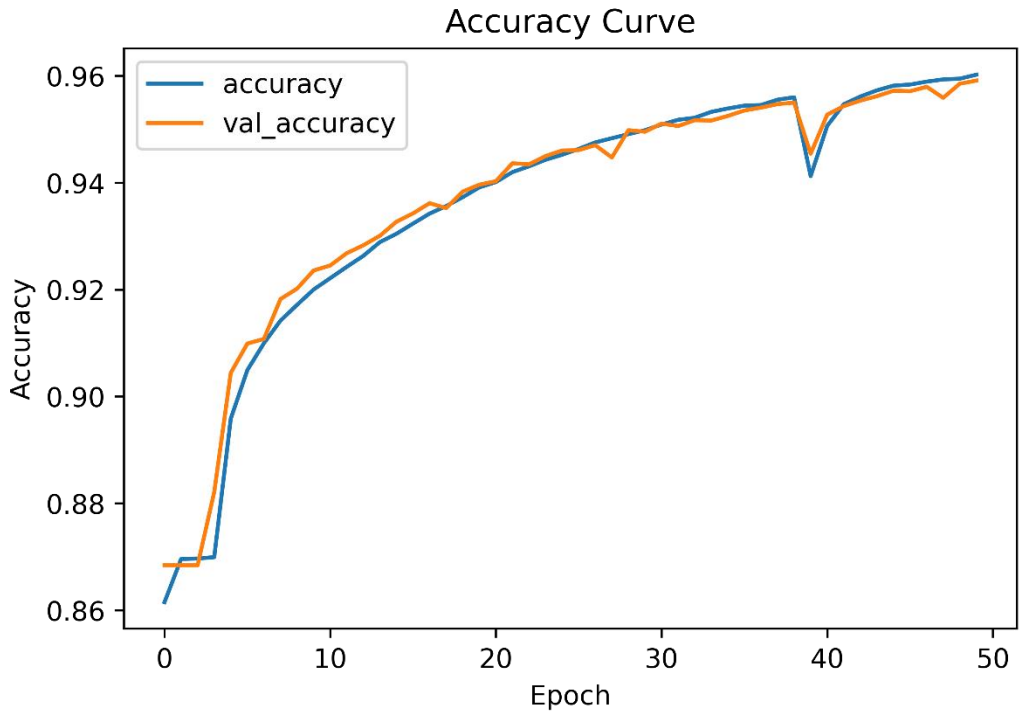


Train Loss = 0.0968, Valid Loss = 0.0988

Figure 35 Loss Curve of the PDB dataset on Triple Fusion Explainable model

b) Accuracy

In Figure 36, the two curves gradually rise and fluctuate in the later stage. However, the two curves are nearly coincident, and the risk of overfitting is relatively low. The model can be further optimized to reduce the fluctuations.



Train Acc = 0.9602, Valid Acc = 0.9591

Figure 36 Accuracy Curve of the PDB dataset on Triple Fusion Explainable model

c) ROC Curve

Figure 37 shows the ROC curves for the three categories (H, E, C) all have AUC values of 0.99, indicating that the model achieves excellent classification performance across all categories.

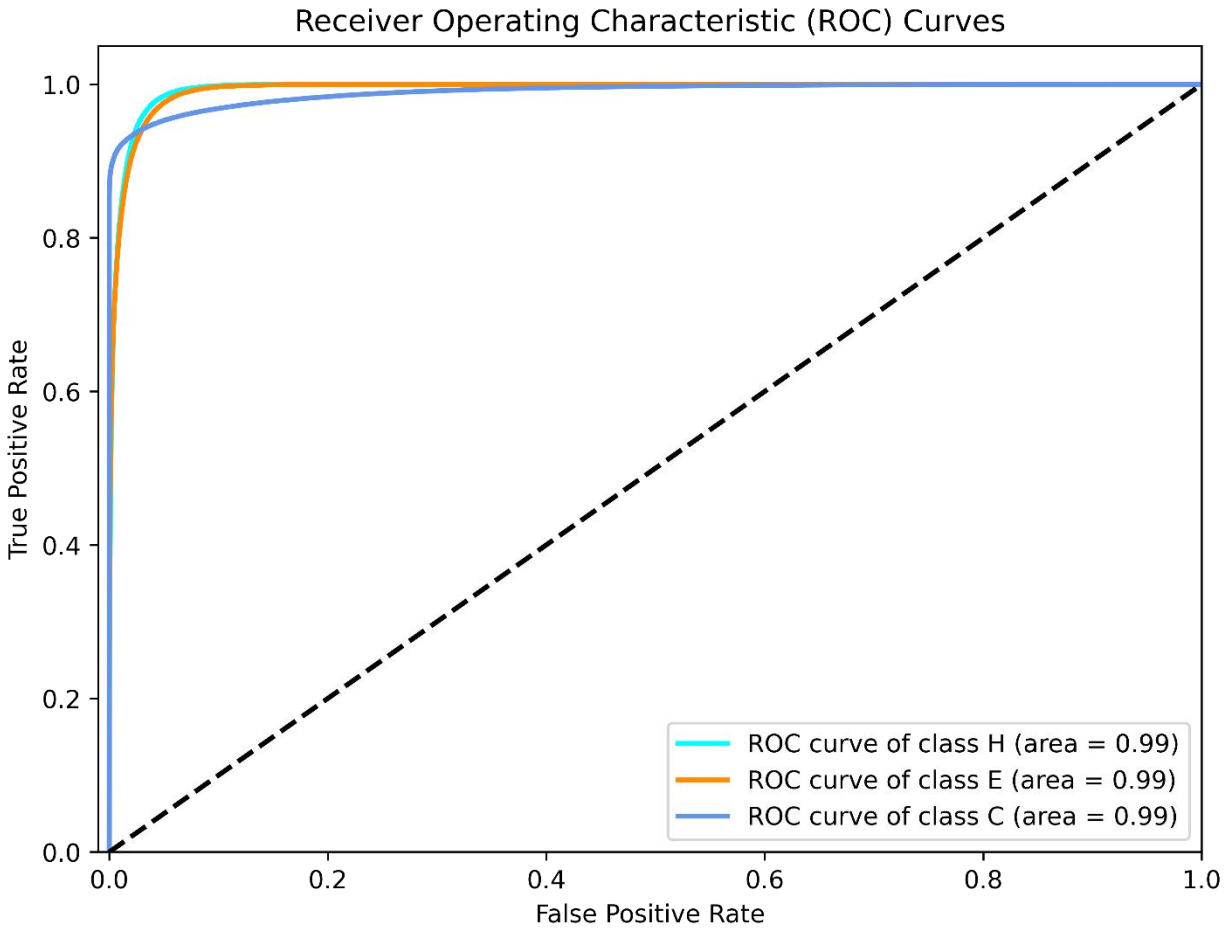


Figure 37 ROC Curve of the PDB dataset on Triple Fusion Explainable model

In this project, when training the Protein Secondary Structure dataset, four different models were adopted, namely the CSP model, the Attention-CSP model with self-attention mechanism, the Memory-based Structure Network model using bidirectional memory-based structure, and the Triple Fusion Explainable model integrating the first three models. The accuracy rates of these models were 94.37%, 94.68%, 95.52%, and 95.64% respectively, showing a gradually improving trend. This result not only validates the enhancement of the model's predictive ability but also proves the effectiveness of the Triple Fusion Explainable model in the protein secondary structure prediction task.

Furthermore, to evaluate the model's generalization ability, the PDB dataset was used for verification. The results showed that the accuracy rate on the PDB dataset was nearly the same as that on the Protein Secondary Structure dataset, which further proved that the model not only performed well on specific datasets but also had good predictive performance for long sequence protein structures in the PDB dataset. This

further demonstrated that the Triple Fusion Explainable model has good generalization ability.

The table below provides a detailed summary of the evaluation results for each model.

Table 3 Result of different model and dataset

Dataset	Model	Category	Accuracy	Loss	Precision	Recall	Specificity	F1-Score	ROC Curve
Dataset 1: Protein Secondary Structure	CSP	Class H			0.6455	0.6758	0.9774	0.6603	
		Class E			0.5899	0.4034	0.9901	0.4792	
		Class C			0.9723	0.9809	0.7225	0.9766	
		Overall	0.9437	0.1263	0.9404	0.9437	0.8966	0.9414	0.9803
	Attention-CSP	Class H			0.7013	0.6667	0.9827	0.6836	
		Class E			0.6092	0.4997	0.9886	0.5490	
		Class C			0.9718	0.9814	0.7174	0.9766	
		Overall	0.9468	0.1206	0.9438	0.9468	0.8962	0.9451	0.9827
	Memory-based Structure	Class H			0.9793	0.7559	0.9854	0.7576	
		Class E			0.6993	0.5458	0.9917	0.6131	

	Network	Class C			0.9748	0.9832	0.7484	0.9790	
		Overall	0.9552	0.1048	0.9530	0.9552	0.9085	0.9538	0.9874
	Triple Fusion Explainable Model	Class H			0.7977	0.7321	0.9887	0.7635	
		Class E			0.6658	0.6135	0.9891	0.6386	
		Class C			0.9756	0.9835	0.7557	0.9795	
		Overall	0.9564	0.1016	0.9548	0.9564	0.9112	0.9555	0.9882
Dataset 2: PDB	CSP	Class H			0.7282	0.8822	0.9717	0.7978	
		Class E			0.8003	0.7607	0.9907	0.7800	
		Class C			0.9829	0.9667	0.8836	0.9748	
		Overall	0.9504	0.1280	0.9542	0.9504	0.9487	0.9517	0.9870
	Attention-CSP	Class H			0.7480	0.7153	0.9793	0.7313	
		Class E			0.7280	0.6553	0.9880	0.6898	
		Class C			0.9650	0.9740	0.7550	0.9695	
		Overal	0.9386	0.14	0.9368	0.93	0.9074	0.93	0.98

		I		36		86		76	28
	Triple Fusion Explainable Model	Class H			0.8475	0.8708	0.9865	0.8590	
		Class E			0.8484	0.7582	0.9933	0.8008	
		Class C			0.9765	0.9796	0.8362	0.9780	
		Overall	0.9606	0.0946	0.9603	0.9606	0.9387	0.9603	0.9924

4.5 Triple Fusion Model Explainability

To enhance the interpretability of our machine learning model, we employed SHAP (SHapley Additive exPlanations) values to visualize the contribution of each feature to the model's output. SHAP values quantify the impact of each feature on the prediction, providing insights into the decision-making process of the model. This project uses force, waterfall, bar, dot and summary plots to analyze the model results.

4.5.1 Bar plot for each class

This part selects a single sample from each category for explanation, and analyzes the role of each amino acid in different categories. The blue bars indicate that this amino acid has a negative impact on the result, that is, it reduces the probability of the sample being predicted as that category. The red bars indicate a positive impact, that is, it increases the probability of the sample being predicted as that category.

Figure 38 shows that a single sequence was selected for analysis in category C. As can be seen from the figure, the two types of amino acids, Alanine (A) and Aspartic acid (D), have a positive effect on predicting category C. Methionine (M), Glycine (G), X (Unknown or filled amino acids), and Serine (S) have a negative effect. Among them, Alanine has the greatest positive effect and Methionine has the greatest negative effect.

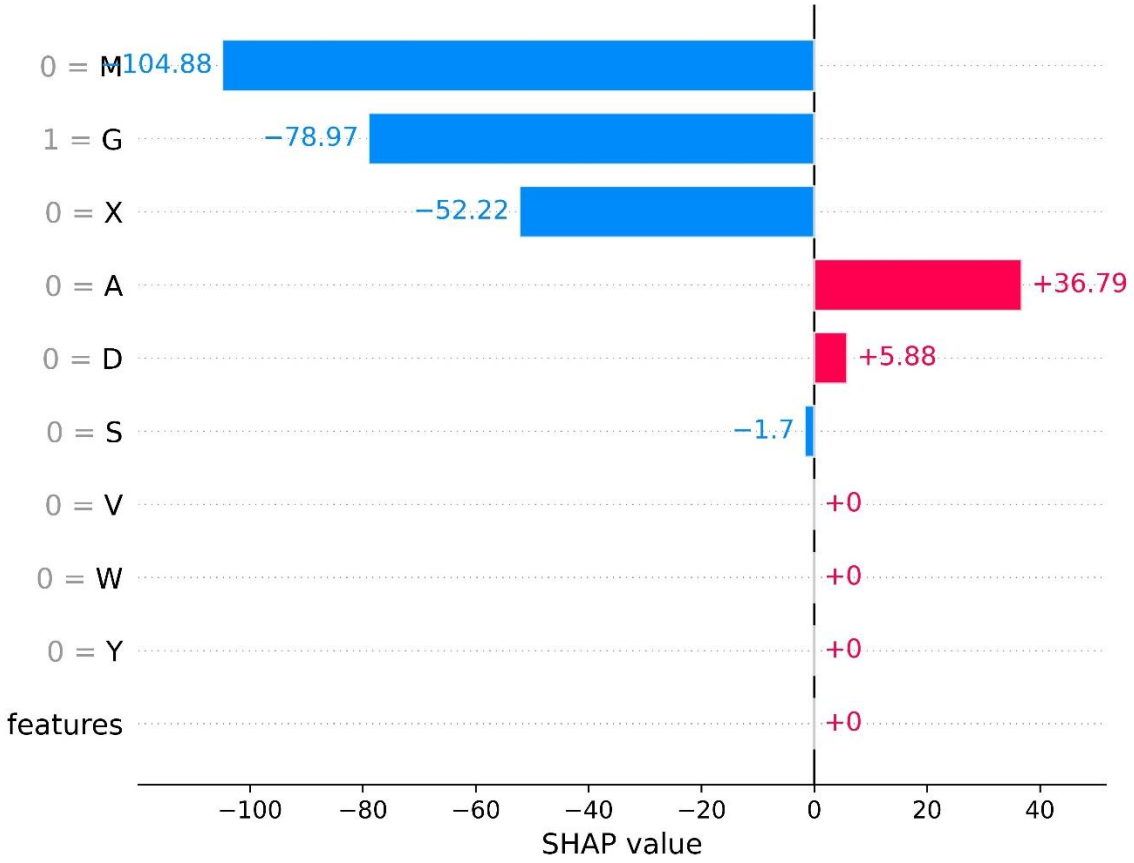


Figure 38 Bar plot for class C

Figure 39 shows that a single sequence was selected for analysis within category E. As can be seen from the figure, the three types of amino acids, namely Alanine (A), Aspartic acid (D), and Serine (S), have a positive effect on predicting as E category, while Methionine (M), Glycine (G), and Unknown or filled amino acids (X) have a negative effect. Among them, Alanine (A) has the greatest positive effect, and Methionine (M) has the greatest negative effect.

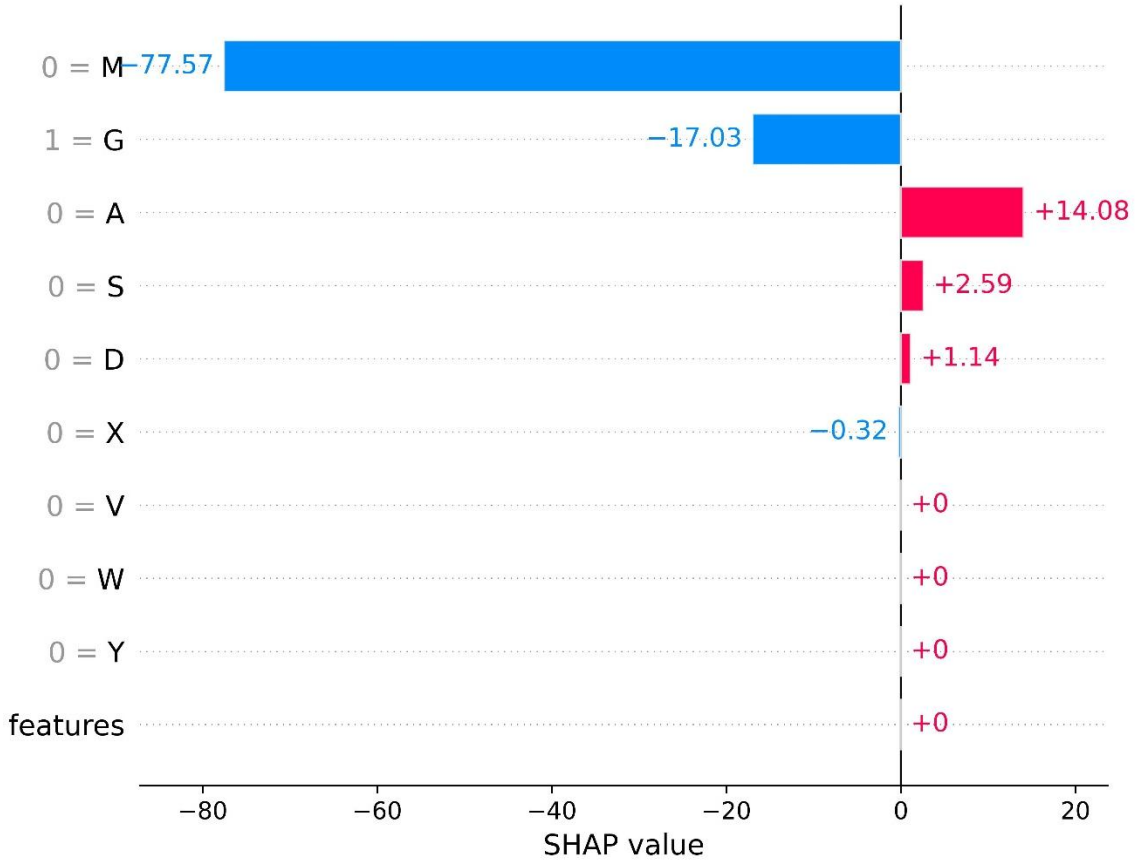


Figure 39 Bar plot for class E

Figure 40 shows that a single sequence was selected from category H for analysis. As can be seen from the figure, the three types of amino acids, namely M (Methionine), G (Glycine), and X (Unknown or filled amino acids), have a positive effect on predicting as H category. A (Alanine), D (Aspartic acid), and S (Serine) have a negative effect. Among them, M has the maximum positive effect and A has the maximum negative effect.

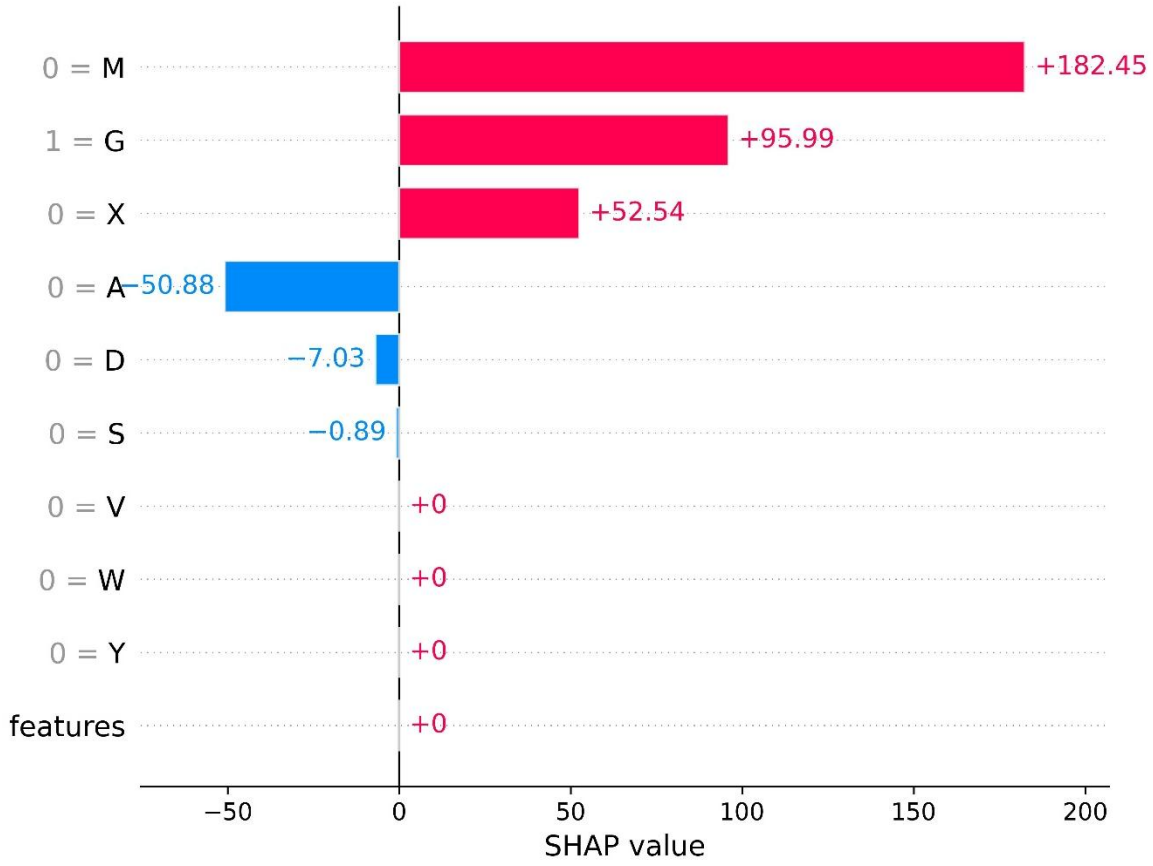


Figure 40 Bar plot for class H

4.5.2 Force plot for each class

The force plot is used to explain the prediction results of a single sample, showing how the contribution of each feature to the predicted value gradually changes through SHAP values. The term base value refers to the predicted value of the model when no influence from any features is present. The marked above is the output value, which is obtained by adding the SHAP values of all features to the base value. Red color indicates the features that pull the predicted result towards a higher value, while blue color represents the features that pull the predicted result towards a lower value. Furthermore, the longer the arrow indicates that the absolute value of the SHAP value for that feature is larger, meaning that the impact of that feature on the prediction result of the current sample is more significant.

Figure 41 illustrates that Glycine (G) has the greatest contribution value to the prediction, with an impact value of 1. This has lowered the prediction score of class C. The

contributions of Alanine (A), Methionine (M) and X (Unknown or filled amino acids) are relatively small.

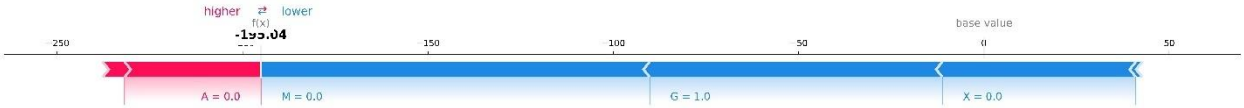


Figure 41 Force plot for class C

Figure 42 illustrates that Glycine (G) has the greatest contribution value to the prediction, with an impact value of 1. This has lowered the prediction score of class E. The contributions of Alanine (A), Methionine (M), D (Aspartic acid), and S (Serine) are relatively small.

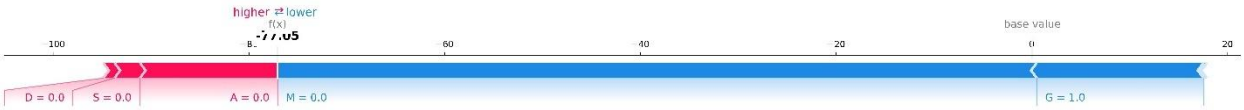


Figure 42 Force plot for class E

Figure 43 illustrates that Glycine (G) has the greatest contribution value to the prediction, with an impact value of 1. This has higher the prediction score of class H. The contributions of Alanine (A), Methionine (M), and X (Unknown or filled amino acids) are relatively small.

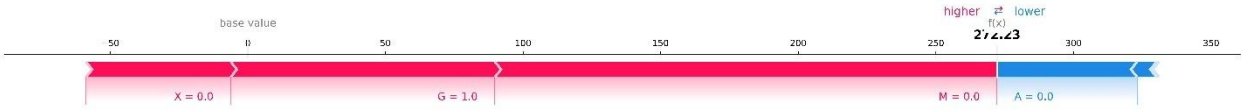


Figure 43 Force plot for class H

4.5.3 Dot plot for each class

In the dot plot, the right side of the horizontal axis indicates that this feature makes the model more inclined to predict as that category, while the left side represents a greater tendency not to predict as that category. The color of the dots represents the magnitude of the feature value; red indicates a high value and blue indicates a low value.

Figure 44 indicates that in class C, the X (Unknown or filled amino acids) feature has the most significant influence on the model and has a high eigenvalue. P (Proline), L (Leucine), G (Glycine) and I (Isoleucine) followed, while the influence of other amino acid characteristics was relatively weak.

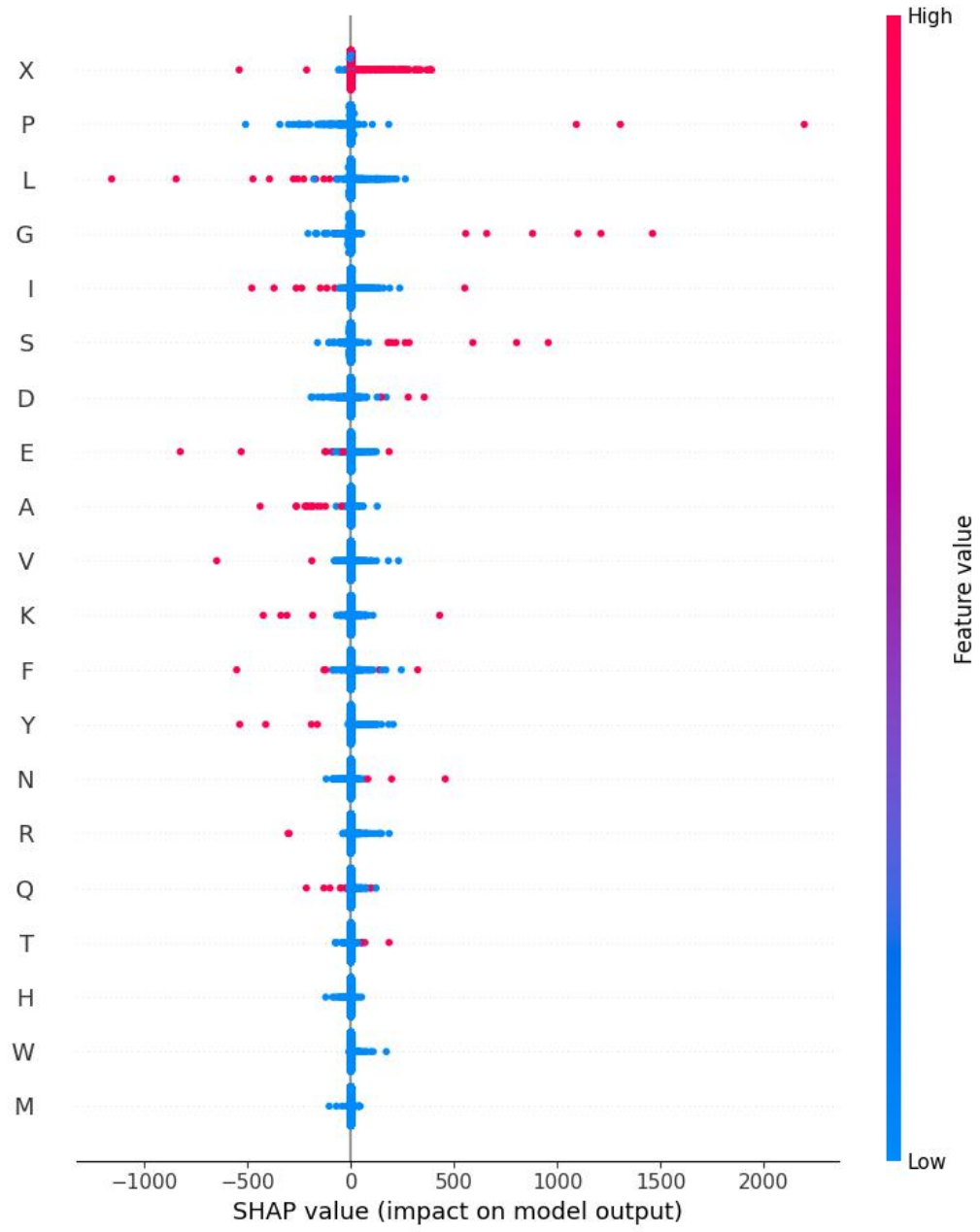


Figure 44 Dot plot for class C

Figure 45 indicates that in class E, the X (Unknown or filled amino acids) feature has the most significant influence on the model and has a high eigenvalue. V (Valine), L

(Leucine), G (Glycine) and I (Isoleucine) followed, while the influence of other amino acid characteristics was relatively weak.

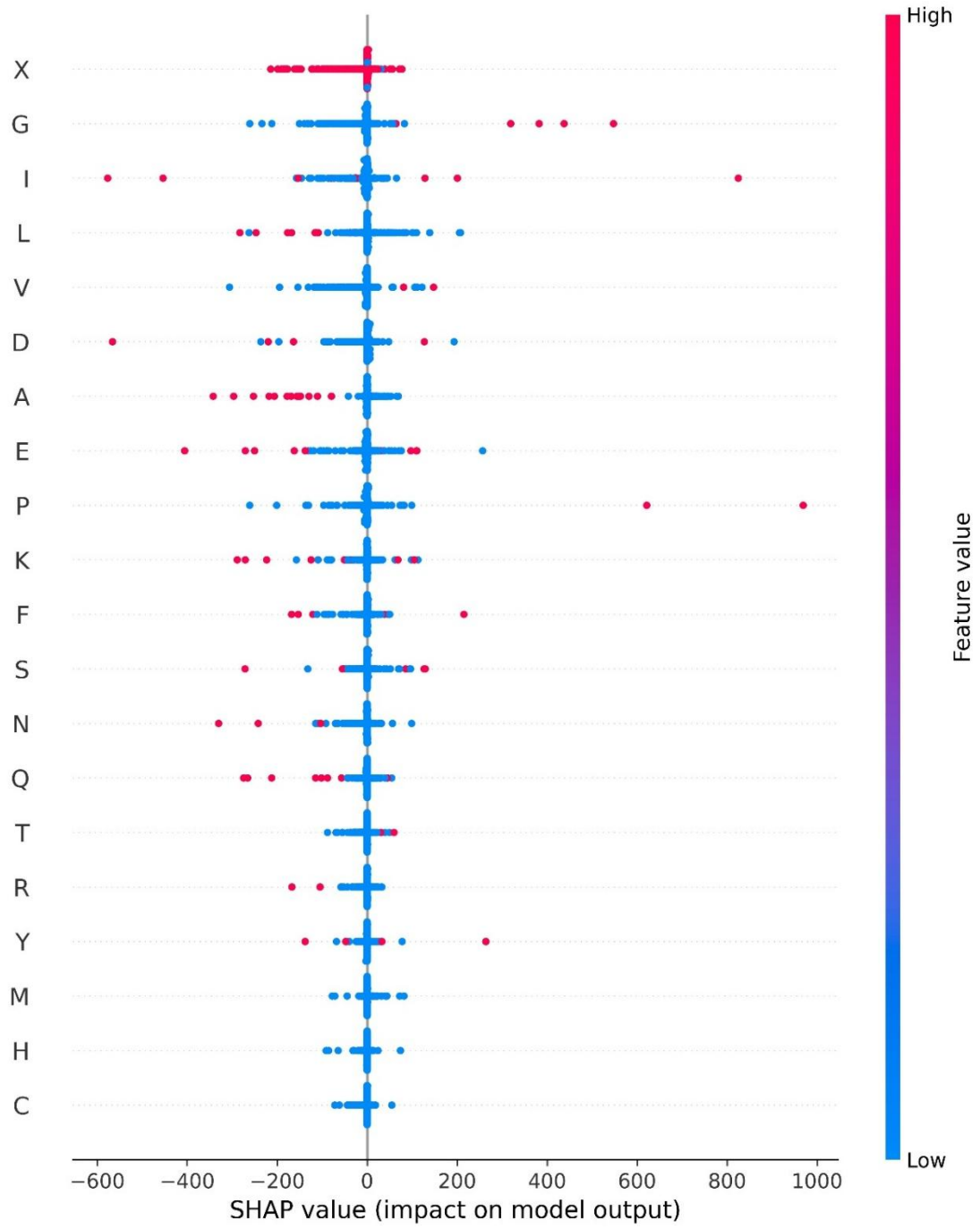


Figure 45 Dot plot for class E

Figure 46 indicates that in class H, the X (Unknown or filled amino acids) feature has the most significant influence on the model and has a high eigenvalue. P (Proline), L

(Leucine), G (Glycine) and I (Isoleucine) followed, while the influence of other amino acid characteristics was relatively weak.

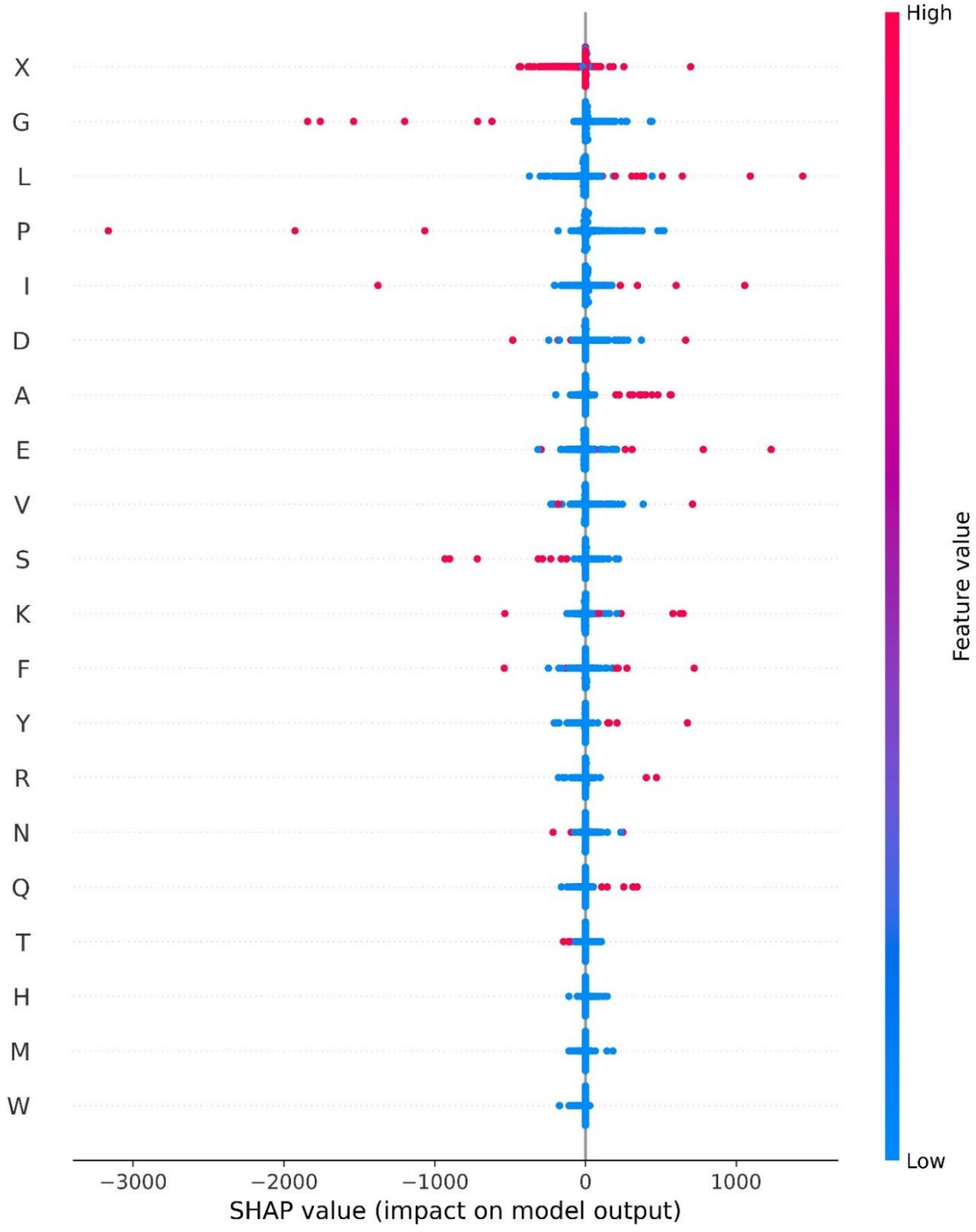


Figure 46 Dot plot for class H

4.5.4 Waterfall plot for each class

In the waterfall plot, there is also an expected value. The red color indicates that this feature has a positive effect on the prediction result, while the blue color represents a negative effect with a reduction in the prediction result. The waterfall plot illustrates how each feature influences the output result through its contribution.

Figure 47 shows that M (Methionine), G (Glycine), X (Unknown or filled amino acids) and S (Serine) all have negative effects on the prediction of category C, while A (Alanine) and D (Aspartic acid) have positive effects. Among them, the negative effect of M is the greatest, while the positive effect of A is the greatest.

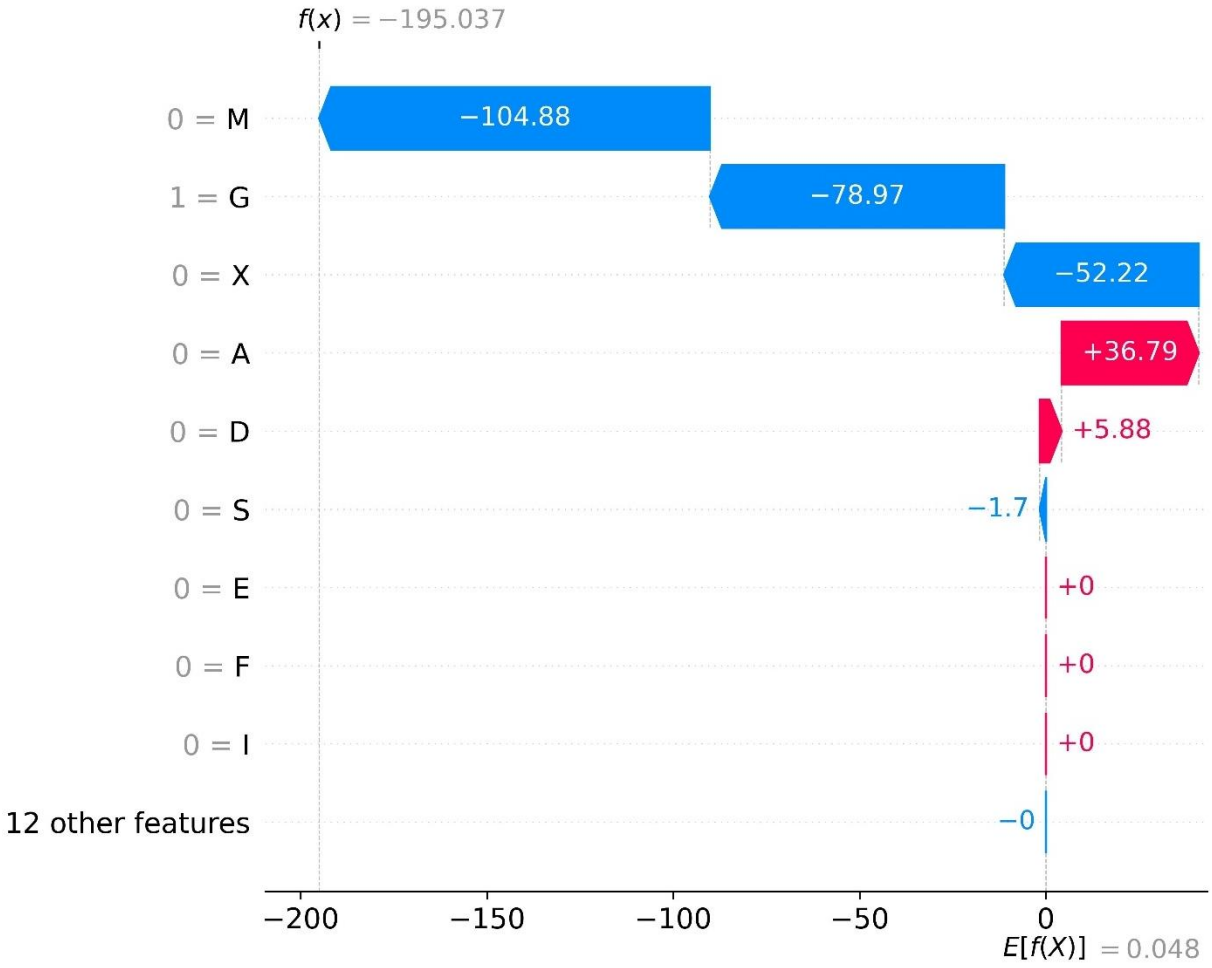


Figure 47 Waterfall plot for class C

Figure 48 shows that M (Methionine), G (Glycine), and X (Unknown or filled amino acids) all have negative effects on the prediction of category E, while A (Alanine), S (Serine)

and D (Aspartic acid) have positive effects. Among them, the negative effect of M is the greatest, while the positive effect of A is the greatest.

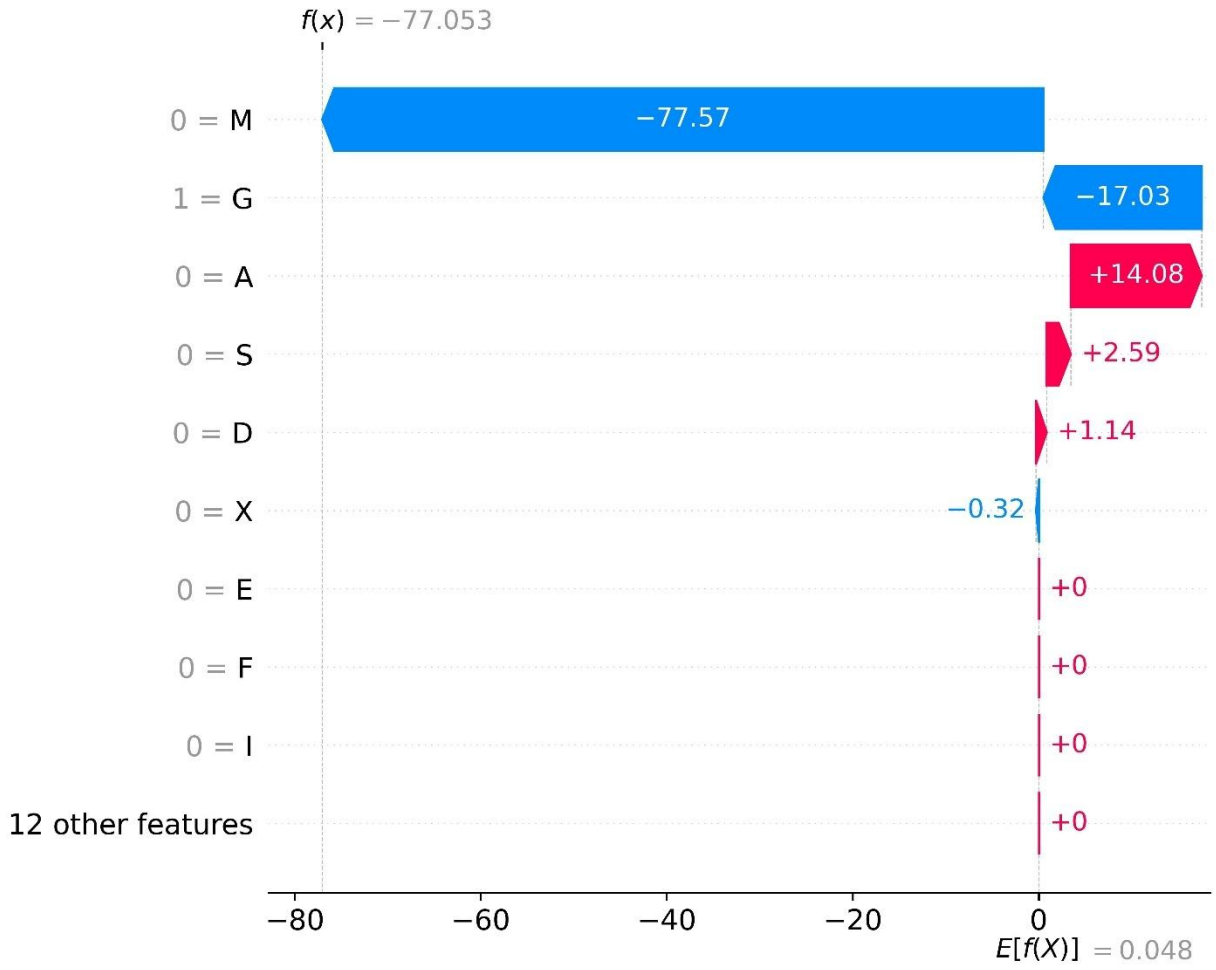


Figure 48 Waterfall plot for class E

Figure 49 shows that A (Alanine), S (Serine) and D (Aspartic acid) all have negative effects on the prediction of category H, while M (Methionine), G (Glycine), and X (Unknown or filled amino acids) have positive effects. Among them, the negative effect of A is the greatest, while the positive effect of M is the greatest.

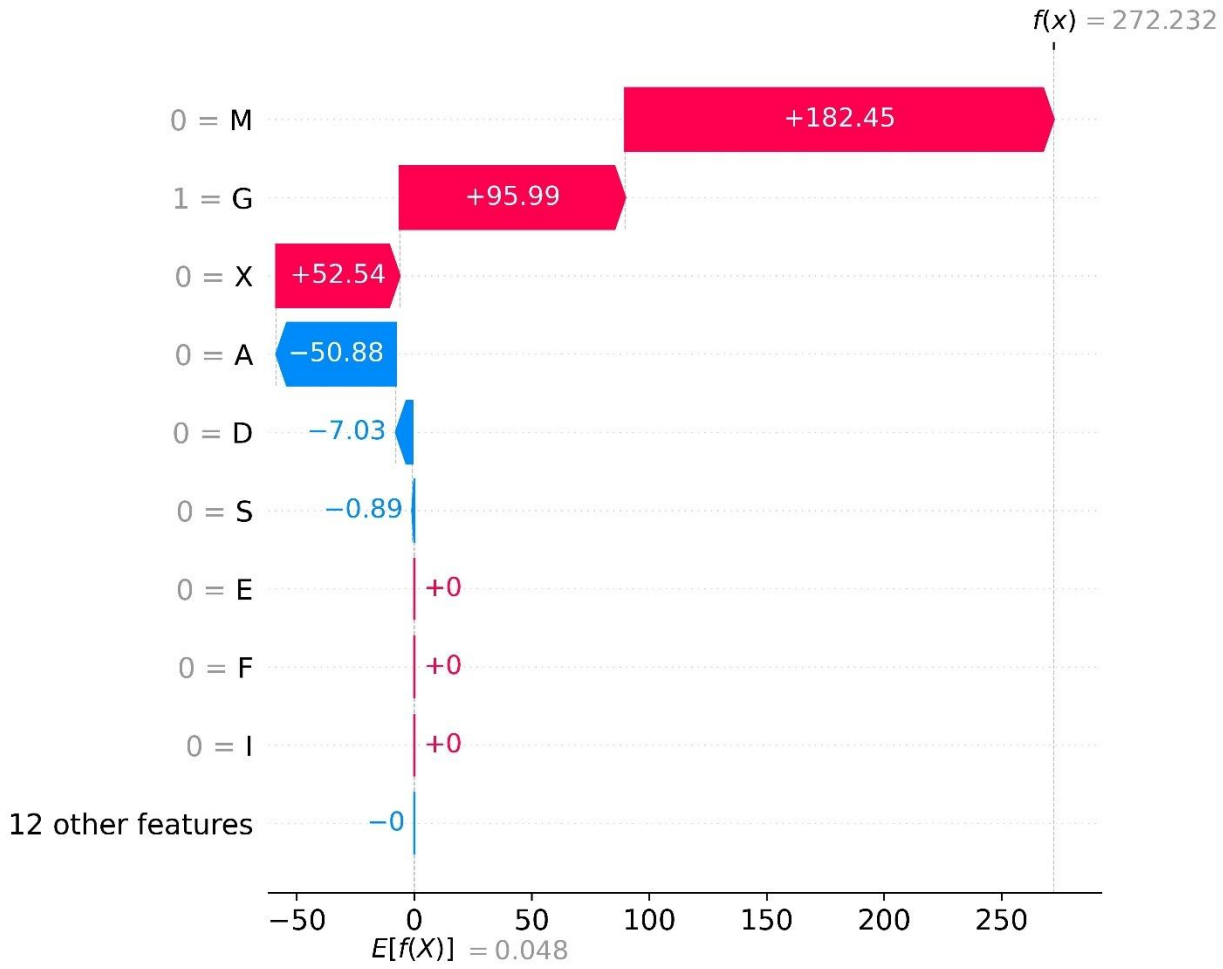


Figure 49 Waterfall plot for class H

4.5.5 Summary plot

The summary plot is used to illustrate the influence of different amino acid features on each category. Different colors represent the influence on different categories, and the length of the bars indicates the magnitude of the influence.

Figure 50 shows the influence magnitudes of different amino acids in different categories. Among them, X (Unknown or filled amino acids) has the greatest influence in each category, while M (Methionine) and W (Tryptophan) have relatively small influences.

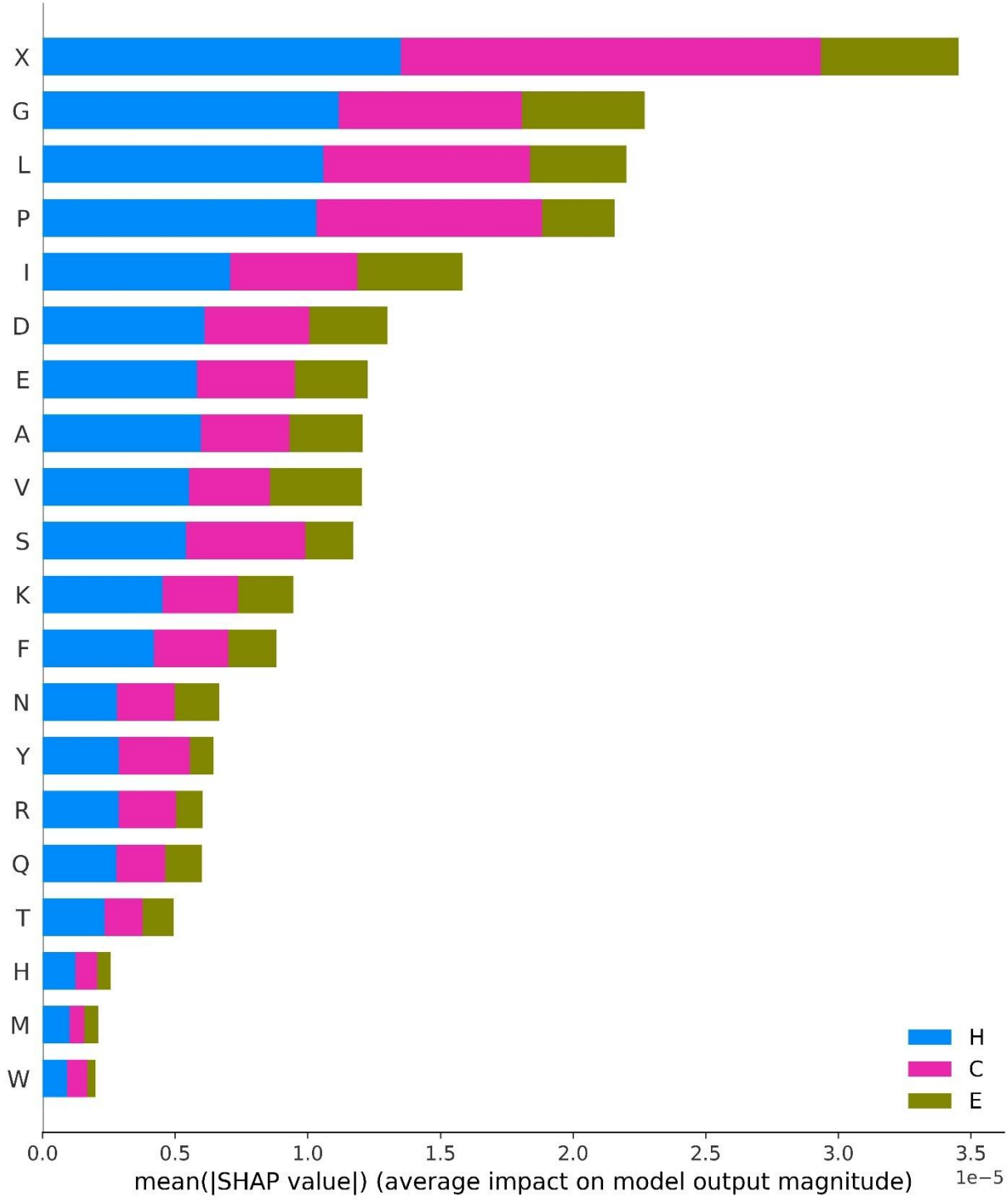


Figure 50 Summary plot

4.6 Model Deployment

The provided figures showcase the deployment GUI for proposed model.

The homepage shows the introduction of the project, how to use the GUI to make predictions, the model used in the project and the type of protein secondary structure, so that users can have a basic understanding of the GUI. In addition, the left sidebar can switch between different functions of the GUI. Figure 51 shows the homepage of the GUI.

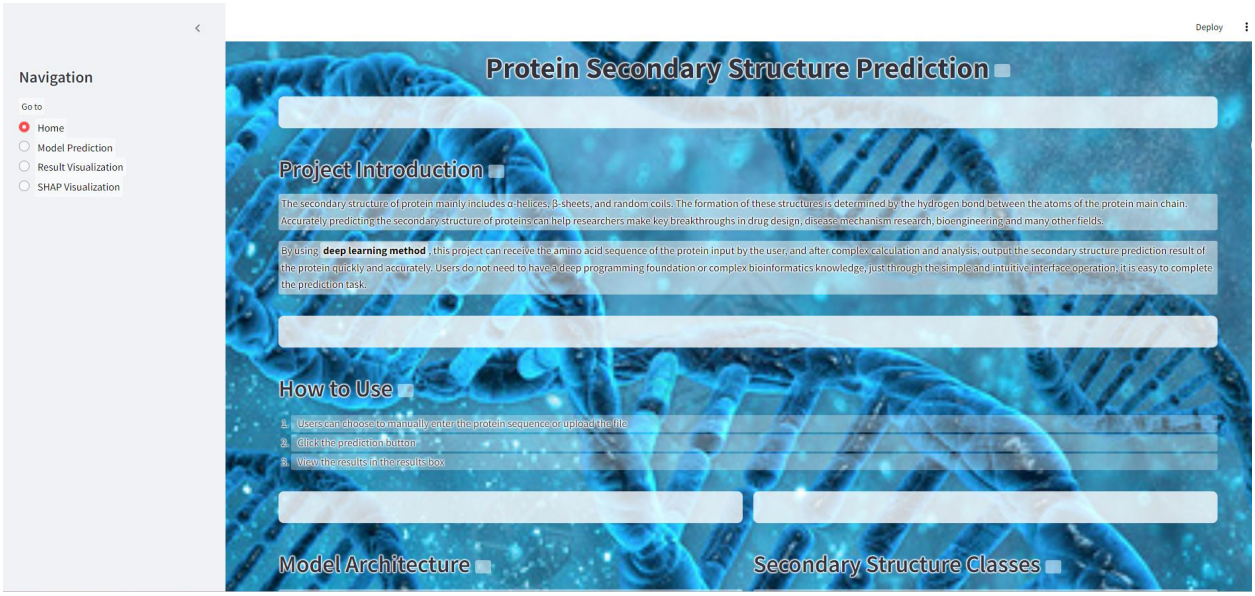


Figure 51 Homepage of the GUI

The prediction page, users can choose to manually enter a single protein sequence for prediction or upload a CSV file for batch prediction. Figures 52,53 show the manual input page and the upload file page, respectively. After entering or uploading the file, click the predict button to make predictions. For a single sequence, the result can be displayed directly. For batch forecasting, only some results can be displayed on the page due to page restrictions. Users need to download the forecast result file to view all predict results.

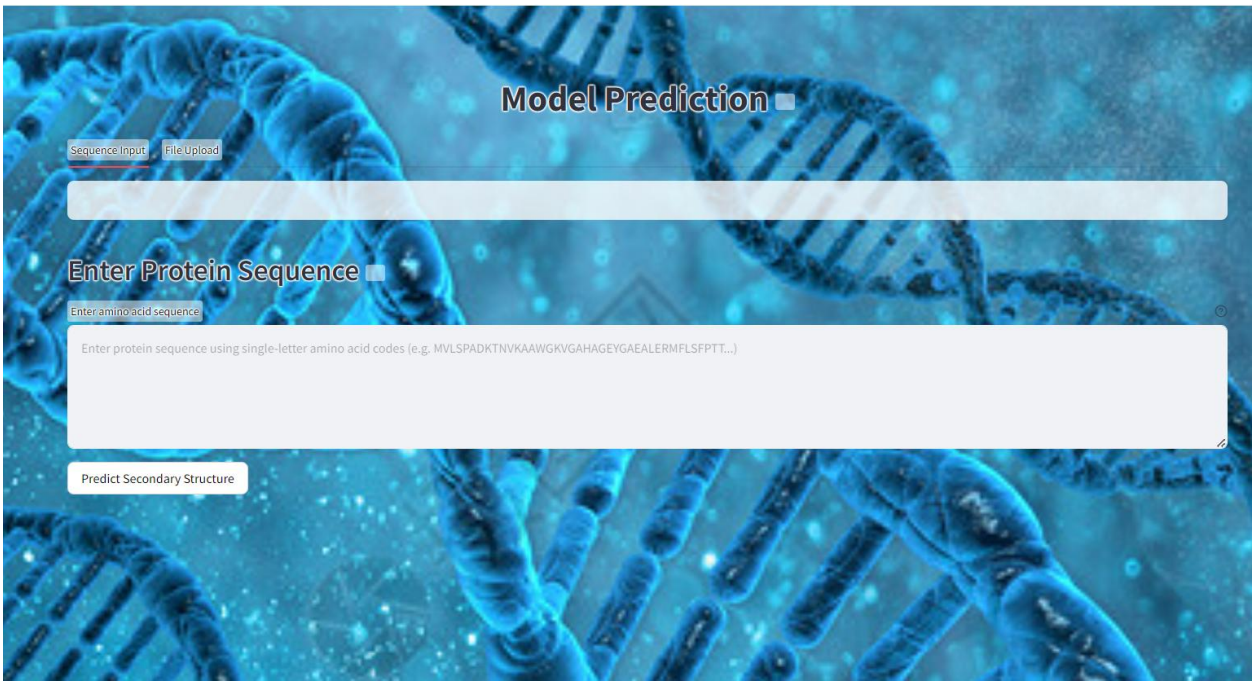


Figure 52 Manual input page

Figure 53 Upload file page

The results visualization page, the Loss, Accuracy, Confusion Matrix, and ROC Curves images of the model are displayed. Users can select different options to view the images and have a more intuitive understanding of the model performance. Figure 54 shows the resulting visualization page.

Figure 54 Results visualization page

The SHAP visualization page, dot, bar, summary, waterfall, force graphs for each class are shown, and users can select different types to view different SHAP results. The slide down page also includes an explanation of each graph and an analysis of amino acid contributions. Figure 55,56 shows the SHAP visualization page.

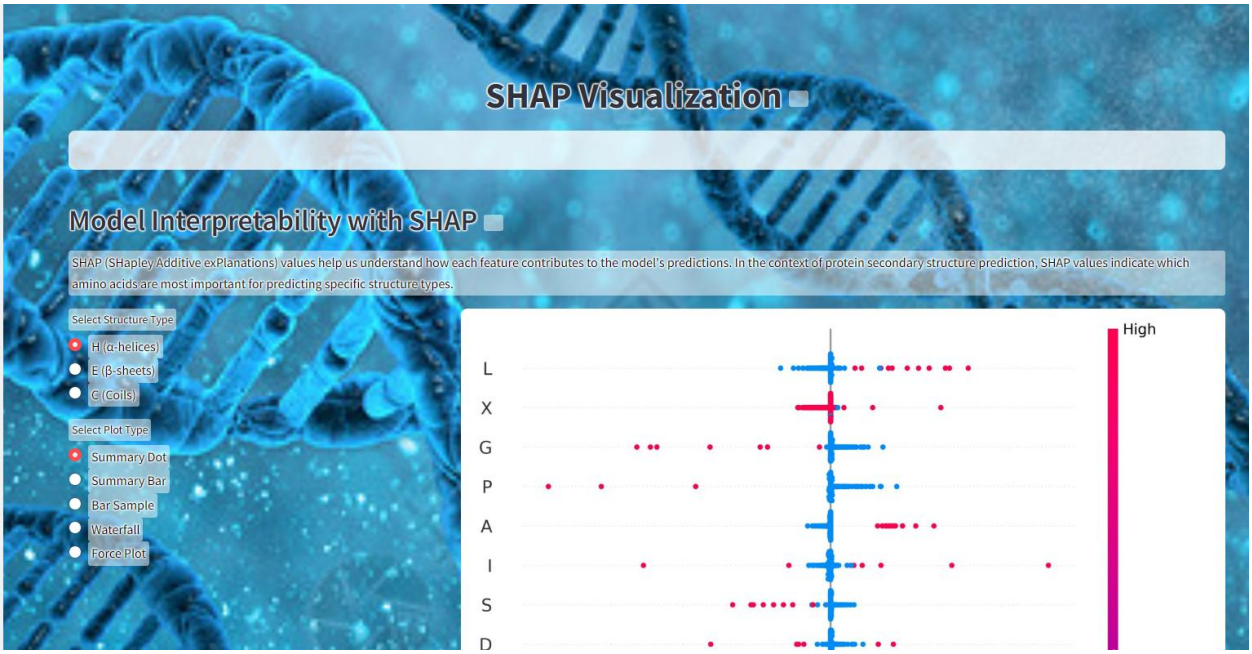


Figure 55 SHAP visualization page

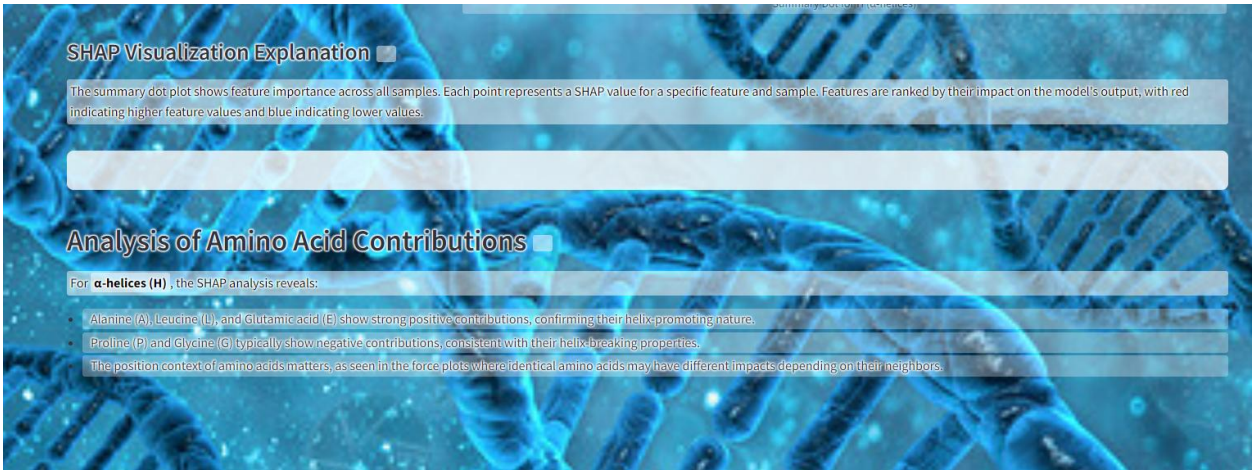


Figure 56 SHAP visualization page

Chapter 5 Professional Issues

5.1 Project Management

This part will include four sections: activities, schedule, project data management, and project deliverables. To comprehensively present the arrangement of the entire project.

5.1.1 Activities

The entire project is divided into several parts for implementation. Table 4 shows each part and the specific contents of it.

Table 4 Activities of the project

Activities	Details
Collection of documents (Completed)	Determine a topic, Collect and study some literature
Project Proposal (Completed)	Complete project proposal document
Model prepare (Completed)	Research models, find suitable methods to solve the problem
Model implementation (Completed)	Build model, analyze and train the model
Model improvement (Completed)	Make code improvements based on the shortcomings
Data prepare (Completed)	Search and collect dataset
Data processing (Completed)	One-hot is used to process the data and determine the training and test data
Thesis writing (Completed)	Conduct essay writing
Thesis modify (Completed)	Make thesis reversions
Presentation preparation (Completed)	Prepare for the presentation

5.1.2 Schedule

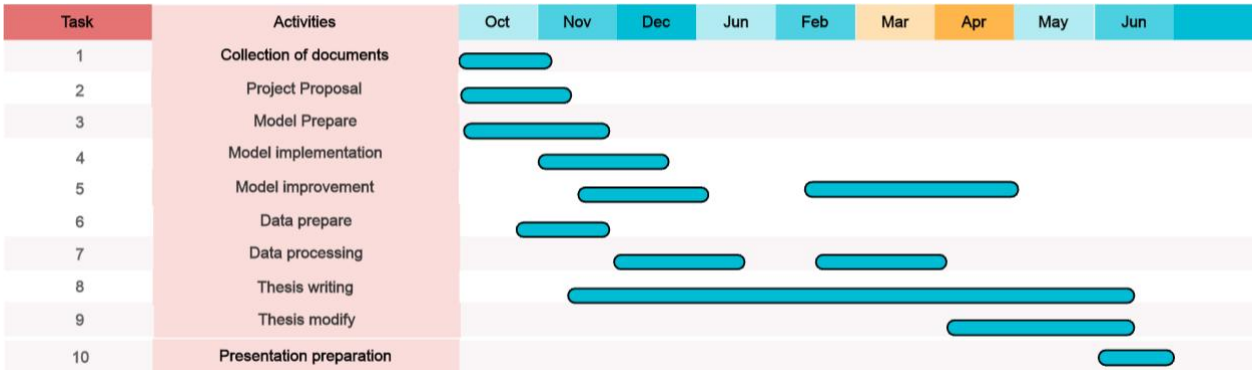


Figure 57 Gantt Chart

5.1.3 Project Data Management

To manage the different versions of codes modification, I plan to use Github as the version management tools for keeping code updated and secure.

URL is as follow: https://github.com/Cecilia-chu/Deep-learning-project

Table 5 Version Control Progress

Version Number	Code Name	Content	Result
1	Individual model code for basic model testing	Data training for individual models	The training results, csv file that contains each epoch for later comparison
2	Few combinations of models	Integrate multiple models and train data	CSV file that contains each epoch for later comparison
3	Compare model and choose final model	Compare different models to determine the final model	CSV file that contains each epoch for later comparison

5.1.4 Project Deliverables

This section lists what needs to be delivered throughout the project.

- The project proposal
- Progress Report
- Final Project Report
- Project codes
- Project presentation slides
- Project presentation

5.2 Risk Analysis

Table 6 Risk Analysis

Risk ID	Potential Risk	Cause ID	Potential Causes	Severity	Likelihood	Risk	Mitigation ID	Mitigation
R1.1	Dataset quality issues	C1.1.1	The dataset may contain incomplete data, or mislabeled entries.	4	2	8	M1.1.1	Verify and correct data labels, and validate model robustness with multiple datasets.
R1.2	Model overfitting	C1.2.1	The model is highly complex, potentially performing well on training data but poorly on test data	4	3	9	M1.2.1	Use regularization techniques (e.g., Dropout, L2 regularization), increase dataset diversity, and implement early stopping.

R1.3	Long training time	C1.3.1	The combination of CSP, Attention-CSP, and Memory-based Structure Network models has high computational complexity, leading to prolonged training time.	3	1	3	M1.3.1	Optimize model structure to reduce redundancy, use distributed training or hardware acceleration (e.g., GPUs, TPUs).

5.3 Professional Issues

In the realm of bioinformatics research, particularly when applying deep learning technologies, several professional issues must be carefully navigated to ensure responsible and impactful development. The following will be explained from the aspects of legal, social and ethical.

Legal Issue

The datasets used in this project for protein secondary structure classification are sourced from publicly available resources. It is essential to ensure that the use of these datasets complies with relevant laws and regulations, such as GDPR. During data handling, it is important to avoid infringing on the copyright or unauthorized intellectual property rights of the original data.

Social Issue

The application of deep learning in bioinformatics can have a significant impact on society. For example, this technology can support the healthcare field by improving the diagnosis and treatment of genetic diseases. However, it is critical to ensure the fairness

and reliability of the model to prevent any unfair outcomes caused by dataset bias or model errors, which could negatively affect certain groups. Therefore, it is necessary to use diverse datasets in the research to ensure the generalizability of the results and prevent the amplification of social issues.

Ethical Issue

The project may involve data related to sensitive biomedical information. Therefore, strict ethical standards must be followed to ensure privacy protection during data processing. Researchers must guard against potential misuse of deep learning technology, such as reverse-engineering sensitive information through the model. Moreover, the development and deployment of the model must include transparency about its interpretability and limitations to avoid misleading decision-makers and the public. Adhering to professional codes of conduct, such as those outlined by BCS and ACM, is particularly important, including principles like "protecting the public interest" and "maintaining integrity."

Environmental Issue

The training and testing of deep learning models require substantial computational resources, which may have environmental impacts. In particular, the combination of CSP, Attention-CSP, and Memory-based Structure Network models often necessitates high-performance computing devices, whose energy consumption may result in significant carbon emissions. Therefore, it is advisable to consider using more energy-efficient model structures or minimizing unnecessary computational operations. Additionally, techniques such as model pruning and quantization can be employed to optimize model performance and reduce the environmental burden.

Chapter 6 Conclusion

Throughout the project, a novel deep learning model (Triple Fusion Explainable Model) was successfully constructed and validated on Protein Secondary Structure dataset and PDB dataset, aims to improve the ability of protein secondary structure prediction. The model combines CSP, Attention-CSP, and Memory-based Structure Network models resulting in impressive performance metrics. The model achieved an accuracy of 95.96%, an AUC of 98.89%, and an F1-Score of 95.93%, specificity of 95.93% and overall, 95.03% precision, indicating its potential as a critical tool in protein secondary structure prediction. The results of this model on the Protein Secondary Structure dataset and PDB dataset are similar, which shows the generalization ability of the model. At the same time, SHAP value is used to analyze the influence of each feature on the model prediction results, which provides ideas for the subsequent model improvement.

Although the various evaluation metrics of the model perform well, it still has certain limitations. When the model was using the PDB dataset, both Loss and Accuracy fluctuated, indicating that the model still has limitations when dealing with long sequences and further optimization of the model is needed. Furthermore, as the length of protein sequences increases, the requirements for hardware also keep rising, and a large enough memory is needed to support the operation of the model.

For the future work, this project only trained protein sequences with a length of less than 5,000. In the future, it is necessary to increase the diversity of data, select longer protein sequences for training, and improve the model's prediction ability for long sequences. In addition, when predicting the secondary structure of proteins, this project adopts a three-classification strategy. In the future, the multi-category classification ability of the model can be improved, and the three-category classification can be converted into eight-category classification, so as to further improve the precision of prediction. This is more conducive to the understanding and research of protein secondary structure, and provides a powerful tool for related scientific research.

References

[1] M. A. Sofi and M. A. Wani, ‘Improving Prediction of Protein Secondary Structures using Attention-enhanced Deep Neural Networks’, in *2022 9th International Conference on Computing for Sustainable Global Development (INDIACom)*, Mar. 2022, pp. 664–668. doi: 10.23919/INDIACom54597.2022.9763114.

[2] D. P. Ismi, R. Pulungan, and null Afiahayati, ‘Deep learning for protein secondary structure prediction: Pre and post-AlphaFold’, *Comput. Struct. Biotechnol. J.*, vol. 20, pp. 6271–6286, 2022, doi: 10.1016/j.csbj.2022.11.012.

[3] M. Sofi and M. ArifWani, ‘Improving Prediction of Amyloid Proteins using Secondary Structure based Alignments and Segmented-PsSm’, *Int. Conf. Comput. Sustain. Glob. Dev.*, 2021, Accessed: Mar. 11, 2025. [Online]. Available: https://www.semanticscholar.org/paper/Improving-Prediction-of-Amyloid-Proteins-using-and-Sofi-ArifWani/8c3cfb5235c919265753005c89af677ad6770812

[4] G. Karypis, ‘YASSPP: better kernels and coding schemes lead to improvements in protein secondary structure prediction’, *Proteins*, vol. 64, no. 3, pp. 575–586, Aug. 2006, doi: 10.1002/prot.21036.

[5] ‘Exploiting the past and the future in protein secondary structure prediction | Bioinformatics | Oxford Academic’. Accessed: Mar. 16, 2025. [Online]. Available: https://academic.oup.com/bioinformatics/article/15/11/937/249908?login=true

[6] R. Yagoubi, A. Moussaoui, A. Dabba, and M. B. Yagoubi, ‘PSCP-CNN: Protein Structural Class Prediction using a Convolutional Neural Network’, in *2022 5th International Symposium on Informatics and its Applications (ISIA)*, Nov. 2022, pp. 1–6. doi: 10.1109/ISIA55826.2022.9993605.

[7] J. Cheng, Y. Liu, and Y. Ma, ‘Protein secondary structure prediction based on integration of CNN and LSTM model’, *J. Vis. Commun. Image Represent.*, vol. 71, p. 102844, Aug. 2020, doi: 10.1016/j.jvcir.2020.102844.

[8] J. Jumper *et al.*, ‘Highly accurate protein structure prediction with AlphaFold’, *Nature*, vol. 596, no. 7873, pp. 583–589, Aug. 2021, doi: 10.1038/s41586-021-03819-2.

[9] W. Wang, S. Nema, and D. Teagarden, ‘Protein aggregation—Pathways and influencing factors’, *Int. J. Pharm.*, vol. 390, no. 2, pp. 89–99, May 2010, doi: 10.1016/j.ijpharm.2010.02.025.

[10] J. Garnier and B. Robson, ‘The GOR Method for Predicting Secondary Structures in Proteins’, in *Prediction of Protein Structure and the Principles of Protein Conformation*, G. D. Fasman, Ed., Boston, MA: Springer US, 1989, pp. 417–465. doi: 10.1007/978-1-4613-1571-1_10.

[11] T. Kalai Chelvi and P. Rangarajan, ‘Analysis of Protein Folding using Structural Concealed Markov Model’, in *INTERNATIONAL CONFERENCE ON SMART STRUCTURES AND SYSTEMS - ICSSS’13*, Mar. 2013, pp. 92–97. doi: 10.1109/ICSSS.2013.6623008.

[12] J. Wang, J. Cheng, Z. Zhao, and W. Lu, ‘Protein Secondary Structure Prediction Using Ensemble of LSTM Neural Networks’, in *2019 2nd International Conference on Information Systems and Computer Aided Education (ICISCAE)*, Sep. 2019, pp. 241–244. doi: 10.1109/ICISCAE48440.2019.221626.

[13] J. Cheng, Y. Liu, and Y. Ma, ‘Protein secondary structure prediction based on integration of CNN and LSTM model’, *J. Vis. Commun. Image Represent.*, vol. 71, p. 102844, Aug. 2020, doi: 10.1016/j.jvcir.2020.102844.