DEEP LEARNING TO PREDICT PROTEIN SECONDARY STRUCTURE

by
Sai Teja Kairamkonda

A THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science
in Computer Science
in the Graduate School of
Troy University

TROY, ALABAMA

March 2023

DEEP LEARNING TO PREDICT PROTEIN SECONDARY STRUCTURE

Submitted by Sai Teja Kairamkonda in partial fulfillment of the requirements
for the degree of Master of Science
in Computer Science
in the Graduate School of
Troy University

Accepted on behalf of the Faculty of the Graduate School by the thesis committee:

Dr. Bill Zhong, Ph.D.	04/11/23 Date
Dr. Xiaoli Huan, Ph.D.	04/11/23 Date
Dr. Alexander Kofman, Ph.D.	04/11/23 Date
Dr. Suman Kumar, Ph.D. Department Chair, Computer Science	4/11/23 Date
Steven L. Taylor, Ph.D. Dean, College of Arts and Sciences	4/13/23 Date
Mary Anne Templeton, Ph.D. Associate Provost and Dean of the Graduate School	4/14/23 Date

ABSTRACT

DEEP LEARNING TO PREDICT PROTEIN SECONDARY STRUCTURE

Protein secondary structure prediction is a basic problem in bioinformatics and drug industry, and disease diagnosis. The methods have mainly focused on predicting secondary structure of one-dimensional sequence space. The goal of the protein structure data is to predict the secondary structure and tertiary structure by using features derived from the 3D structure of the protein by using its amino acid sequence. Many researchers are trying to find out the 3D structure of the protein by using amino acids which is quite difficult because of the different structures and shapes in the field of bioinformatics. Amino acids play a vital role in protein structure formation by linking the primary sequence with the tertiary protein structure. Moreover, primary protein structure has very high computational cost along with maintaining the accuracy as well.

Machine learning has successfully used in various domains to train and predict the results with different data sets. Artificial neural networks (ANNs) serve as a foundation for deep learning, which is used in a wide range of applications, including healthcare, self-driving cars, Demographic and Election Predictions. In our project, deep learning architectures have been proposed to predict the performance of the protein structure, including recurrent neural networks (RNNs) and convolutional neural

networks (CNNs) to predict protein secondary structure. Compared to other studies that simply used amino acid sequences to predict protein secondary structure, we add two additional protein properties as features to deep learning models. The water solvent accessibility of a protein is its first characteristic and thermal stability as its second characteristic. The protein charge, which has two characteristics—positive charge (N-terminal) and negative charge is used for the protein structure prediction and next model will use the thermal stability we exploited to study the structure of the protein. Next, we integrate the characteristics of two proteins, and the deep learning model also makes use of them.

ACKNOWLEDGEMENT

Thank you to my parents for your endless support. You have always stood behind me, and this was no exception.

Most importantly, I am grateful for my family's unconditional, unequivocal, and loving support.

I would like to express my deep gratitude and sincere appreciation to my advisor, Dr. Jiling Zhong, for their invaluable guidance and support throughout the entire process of completing this thesis. Their expertise, insights, and feedback were essential in shaping this work and helping me achieve my academic goals.

I am also thankful to the members of my thesis committee, Dr. Huan and Dr. Kofman, for their valuable suggestions and constructive criticisms, which helped me to improve the quality of this thesis.

I am grateful to my parents and brother, for their unconditional love and support throughout my academic journey. Their sacrifices and encouragement have been a constant source of motivation and inspiration for me.

Finally, I would like to thank all the people who participated in my research project, without whom this thesis would not have been possible.

HUMAN OR ANIMAL SUBJECTS REVIEW FORM

for

Sai Teja Kairamkonda

Name of Student

DEEP LEARNING TO PREDICT PROTEIN SECONDARY STRUCTURE

Title of Research Project

This research project has been reviewed by the Institutional Review Board and approved as follows (the appropriate block must be checked by either the Thesis chair or the Chair of the Institutional Review Board):

☒ **Neither humans nor animals** will be used, and this research is certified exempt from Institutional Review Board review by the thesis committee chair.

- **Human participants** will be used, and this research is certified exempt from Institutional Review Board review by the Chair of the Institutional Review Board.

- **Human participants** will be used, and this research was reviewed and is approved by the Institutional Review Board.

- **Animal participants** will be used, and this research was reviewed and is approved by the Animal Research Review Board.

Signature of Thesis Committee Chair	Date
Signature of Chair of Institutional Review Board	Date

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE: INTRODUCTION

Proteins are complex molecules which are essential to all the living organisms. They are made up of chains of smaller molecules called amino acids, which are linked together through peptide bonds to form polypeptide chains. The sequence and placement of the amino acids will help to create each protein's unique structure and function. There are 20 different types of amino acids commonly found in living organisms, each with a unique side chain or R-group. The side chain gives each amino acid its distinct chemical properties, such as its polarity, acidity, or basicity. Some amino acids are hydrophobic and others are hydrophilic and few have both hydrophobic and hydrophilic properties. According to Zhang and Gladyshev (2017), these two amino acids are typically the 21st and 22nd amino acids; however, they are not utilized in our study. These 20 amino acids are used to create the vast majority of proteins in all forms of life, from bacteria to humans. Protein structure is one of the essential research in computational biology.

Overall, the study of proteins and amino acids is essential to understanding the biochemistry and molecular biology of living organisms. Researchers continue to study these molecules in order to gain a deeper understanding of the mechanisms underlying various biological processes and to develop new treatments for diseases.

There are 20 standard amino acids that are commonly found in living organisms. These building blocks of proteins include:

1) Alanine

2) Arginine

3) Asparagine

4) Aspartic acid

5) Cysteine

6) Glutamic acid

7) Glutamine

8) Glycine

9) Histidine

10) Isoleucine

11) Leucine

12) Lysine

13) Methionine

14) Phenylalanine

15) Proline

16) Serine

17) Threonine

18) Tryptophan

19) Tyrosine

20) Valine.

Each of these amino acids has a unique chemical structure, characterized by its side chain or R-group, that influences its properties and functions. These amino acids can be combined in different sequences and arrangements to create the vast array of proteins

found in all forms of life. Each protein may have multiple amino acids to form the structure but might not use all amino acids. Because of this method, protein can be formed by unlimited amino acids. DNA is a genetic molecule that contains the genetic code of an organism. Nucleotide is a structural form or blocks of DNA. It is based on four chemicals that are Cytosine, Thymine, Adenine, and Guanine. Each gene's code combines these nucleotides in a triplet which encodes an amino acid known as codon. Each group of three-nucleotide makes one amino acid is required to make a protein. The sequence and formation structure of the amino acid is important it decides the protein function based on these characteristics. Protein has been divided into four essential structures: primary, secondary, tertiary, and quaternary. Moreover, secondary structure has folding patterns of the polypeptide chain which are in alpha helices and beta sheets[1]. Protein structure determines the solubility, stability, and interaction with other molecules in the cell.[2]

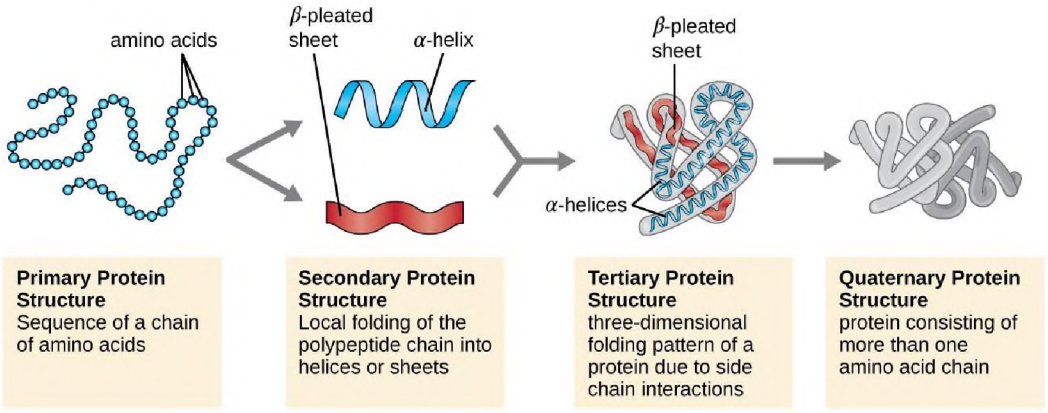


Figure 1 shows the Protein structure[3]

[1] https://byjus.com/chemistry/alpha-helix-and-beta-sheet/

[2] https://www.researchgate.net/publication/11252882_Protein_Structure_Prediction_in_2002

[3] https://microbenotes.com/protein-structure-primary-secondary-tertiary-and-quaternary/

There are various databases like RCSB database, Nucleotide database, and wwPDB database which contains protein sequence from various resources such as RefSeq, GenBank, and TPA. Protein Data Bank (PDB) is a database which is built for storing the 3D structure of biological molecules and it returns over 100,000 structures as of September 2021. On the other hand, GenBank [4] contains over 1.6 trillion nucleotides in over 300 million sequences in December 2022. By 2021, there are over 160K protein structures have been deposited into the RCSB[5]. By Comparing all the protein sequences among the listed above GenBank has the protein sequence is 1500 times more than the protein structure.

The experiments to find the protein structure is very timely and high cost because it is time-consuming process in labs. In our project, we are going to apply deep learning algorithms process to predict the 3d protein structure from the protein sequence by reducing the cost of the process.

Protein secondary structure prediction started in 1951 by Pauling and Corey proposed a model for nucleic acid which are large molecules made from nucleotides. Max Perutz and John Kendrew have awarded Noble Prize in 1962 for identifying the structure of the protein. In 2002, (Jack Schonbrun, William J Wedemeyer and David Bake, 2002) conducted an experiment to predict the protein structure in which humans versus automated servers. One of the most difficult issues in computational biology is accurately predicting the protein 3D structures from the protein sequences alone.

[4] https://www.ncbi.nlm.nih.gov/genbank/statistics/

[5] https://www.rcsb.org/structure/3P7U

Secondary structure prediction accuracy will be used to predict the accurate protein structure (Fischer & Eisenberg, 1996; Wu, Skolnick & Zhang, 2007). Because it relates the primary sequence and tertiary structure, protein secondary structure prediction offers an alternate way for tertiary structure prediction. (Plaxco, Simons & Baker, 1998; Zhou & Karplus, 1999; Ozkan, Wu & Chodera, 2007). However, protein secondary structure prediction is still a classical problem in bioinformatics (Yang, Gao, Wang, Heffernan, Hanson, Paliwal & Zhou, 2016). In this paper, we attempt to improve the performance of predicting protein's secondary structure. Secondary structure assignment technique which is most commonly used will include Dictionary of Secondary Structure of Proteins (DSSP) and three classes Secondary Structure Prediction (SSP), which automatically assigns the secondary structure into eight states according to hydrogen-bonding patterns (Kabsch & Sander, 1983).

Deep learning and Machine learning has already done outstanding results in various domains in the world by giving close results. Neural network(NN) was introduced by Rost & Sander in 1993 with 69.7% accuracy, PSIPRED in 1999 got 76.5% accuracy. Structural Property prediction was achieved 80% accuracy with an Integrated Neural network (SPINE) in 200 and it was increased to 82% accuracy with an Integrated Neural network (SPIDER2) in 2015, 84% accuracy by Deep Convolution neural field network(DeepCNF) in 2016, 84.6% accuracy was achieved by Multilayer Shift-and-Stitch Deep Convolutional Architecture (Wang, Peng, Ma & Xu, 2016). As, we have seen the accuracy achieved from past few decades using artificial neural network which is similar to deep learning to predict the protein secondary structure. In

our project, we will apply both Convolution Neural Network (CNN) and Recurrent Neural Network (RNN) into the protein secondary structure prediction performance by using amino acids with different protein properties.

CHAPTER TWO: PROTEIN STRUCTURE

Proteins are complex macromolecules which are really necessary for the normal operation of living things. They are made up of polypeptide chains, which are made up of long chains of amino acids joined together by peptide bonds. The specific three-dimensional configurations that these polypeptide chains eventually fold into are essential to their function. Therefore, proteins' biological activity depends heavily on their structure and physical attributes.

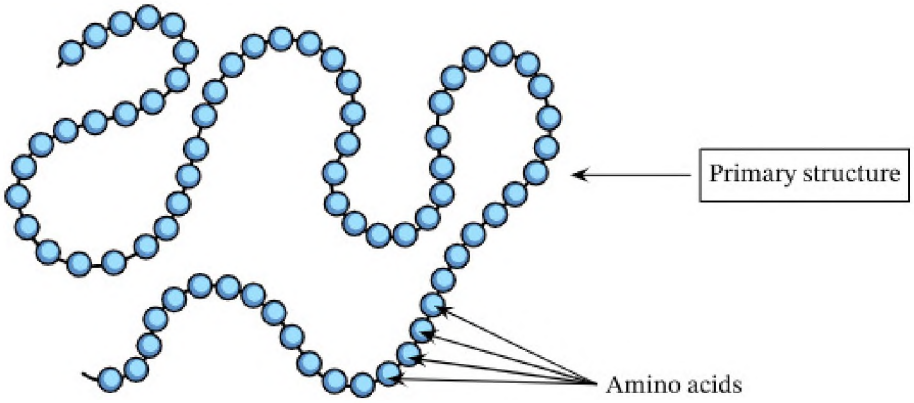


Figure 2 Protein Structure[6]

Protein Properties:

The structure of proteins governs[1] a wide range of characteristics. For instance, the amino acid sequence of a protein is determined by its fundamental structure, which in turn affects the overall charge, hydrophobicity, and reactivity of the protein. A protein's stiffness, flexibility, and capacity to form particular connections with other molecules

[6] https://www.nagwa.com/en/explainers/528127907457/

are all governed by its secondary structure. A protein's overall shape and the particular binding sites that it contains—both of which are essential to its function—are determined by the tertiary structure of the protein. A protein's capacity to interact with other proteins and create useful complexes depends on its quaternary structure.

Protein Structure:

Protein structure is divided into four levels: primary, secondary, tertiary, and quaternary structures. The polypeptide chain's linear amino acid sequence serves as the main structure.

2.1 Protein Primary Structure

The primary structure of a protein refers to the linear sequence of amino acid residues that make up the protein chain. Each amino acid is joined to the next by a peptide bond, forming a chain of amino acids called a polypeptide. The sequence of amino acids in the polypeptide chain is determined by the genetic code and can be represented using a one-letter or three-letter code for each amino acid.

The primary structure of a protein is important because it determines the protein's overall shape and function. Changes in the amino acid sequence can lead to changes in the protein's folding and activity, which can result in various diseases and disorders. Primary structure of a protein structure is shown in Figure. In 1973, Chris Anfinsen proved that a protein's amino acid sequence is the only factor that can influence the higher level of structure (Anfinsen, 1973). It helps to determines the protein functionality. In 2005, Karp and Geer mentioned the change in the sickle cell anemia is brought on by a unique twist in the sequence of hemoglobin. A protein's chain's

structure is dependent on the amino acid sequence it contains. The primary structure of 1,000 proteins has so far been discovered and investigated.

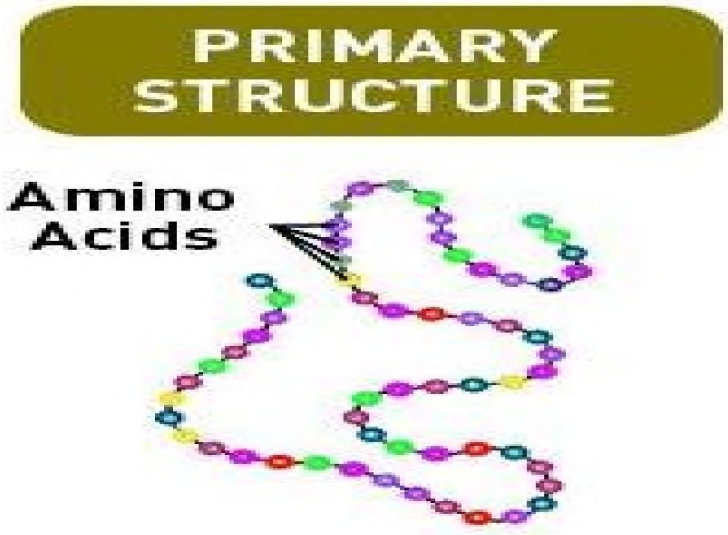


Figure 2 Primary protein structure[7]

2.2 Secondary Protein Structure

The secondary structure in protein structure refers to the specific spatial configuration of the polypeptide backbone. Alpha helices and beta sheets are the two most prevalent forms of secondary structure. In addition to providing more details about the protein's nature, the 3D structural information accessible at this level.

In Figure 2, there are two common types of secondary structure, α-helix and β-pleated sheet.

[7] https://byjus.com/chemistry/protein-structure-and-levels-of-protein/

- Alpha-helices: The amide groups in the polypeptide chain's backbone form hydrogen bonds, which result in the coiling structures known as alpha-helices. The structure is stabilized and given a helical shape by the hydrogen bonds.
- Beta sheets: Beta-sheets are extended structures that come about as a result of hydrogen bonds forming between adjacent polypeptide chain segments. These sections might either be pieces of various polypeptide chains or adjacent strands of a single polypeptide chain.

Because it affects the protein's overall folding and stability, a protein's secondary structure is crucial. The 3D structure and, ultimately, the function of the protein can be significantly influenced by the arrangement of the secondary structural components. The below figure shows the secondary protein structure.

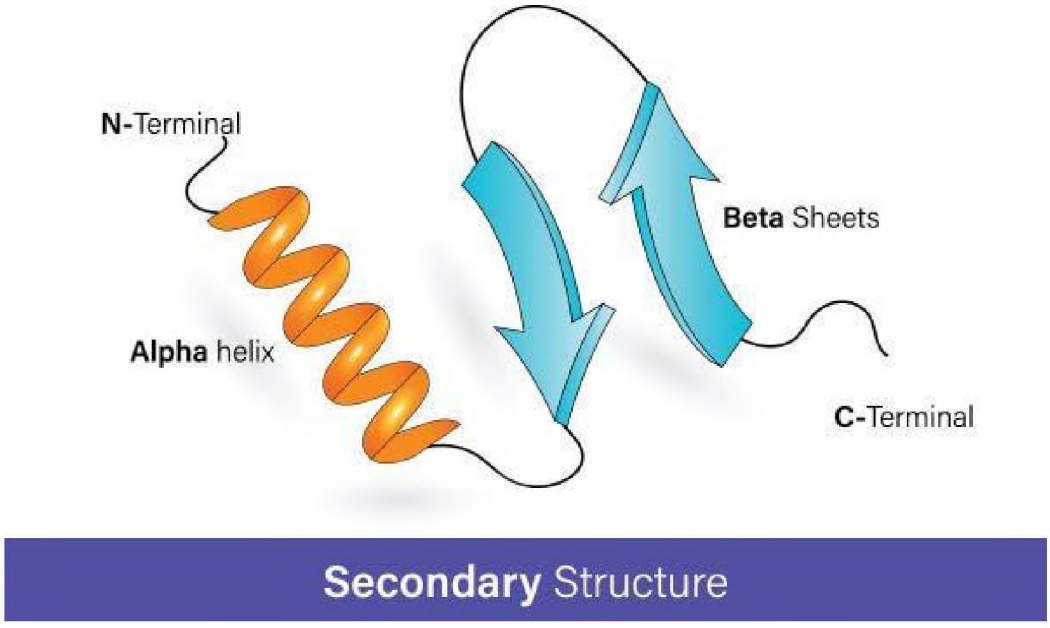


Figure 3 Secondary Protein structure[8]

[8] https://biologydictionary.net/protein-structure/

2.3 Tertiary Protein Structure

The total 3D arrangement of the protein's atoms, including any other structural components like its alpha-helices, beta-sheets, twists, and loops, is referred to as the protein's "tertiary structure" in the context of its structure. A protein's amino acid makeup and the environment in which it is folded both have an impact on the protein's tertiary structure. To function biologically, complex molecules like proteins must fold into a functional 3D structure. The forces that keep the protein in its native state, such as hydrogen bonds, hydrophobic interactions, and electrostatic interactions, promote protein folding. The tertiary structure of a protein may have a substantial impact on how well it functions. For instance, because enzymes have certain pockets and active sites, they can connect to and catalyze specific chemical reactions. Figure Protein tertiary structure.

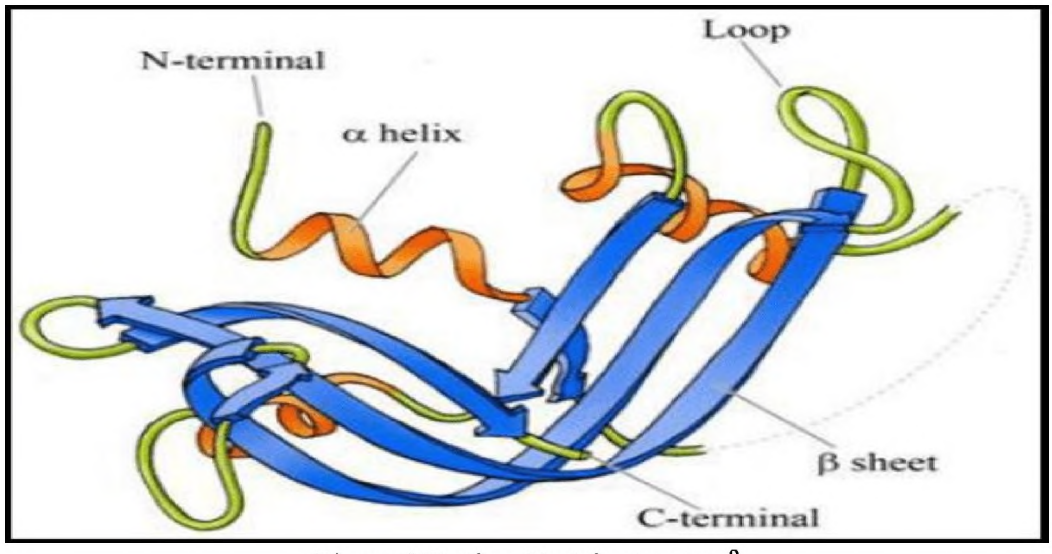


Figure 4 Tertiary Protein structure[9]

[9] https://www.researchgate.net/figure/Schematic-representation-of-the-Tertiary-Structure-of-a-protein-Tertiary-structure-of-a_fig2_348370508

2.4 Quaternary Structure

A protein's quaternary structure describes the spatial organization of various protein subunits, or polypeptide chains, to create a larger functional protein complex. The overall function and structure of the protein will be strongly impacted by the order of these subunits, which may or may not be the same. The quaternary structure of a protein is critical to how it works, and changes to this structure can have detrimental effects. For instance, diseases like sickle cell anemia, cystic fibrosis, and hemophilia can be brought on by quaternary structural alterations brought on by mutations in the genes encoding the subunits of multimeric proteins. Figure 4, the quaternary structure.

For instance, hemoglobin is a tetramer made up of two identical alpha subunits and two identical beta subunits. Hemoglobin is the protein that carries oxygen in red blood cells. Hemoglobin can bind and carry oxygen effectively because to the quaternary structure's careful placement of these subunits. The quaternary structure of a protein can be stabilized by a variety of interactions, including as hydrogen bonds, salt bridges, disulfide bonds, and hydrophobic interactions.

Collagen, the most prevalent protein in the human body, is another example of a protein with quaternary structure. A fibrous protein called collagen gives connective tissues including skin, bone, and cartilage structural support and suppleness. The functional protein collagen has a triple helix structure and is made up of three polypeptide chains, or subunits, that are organized in a quaternary configuration.

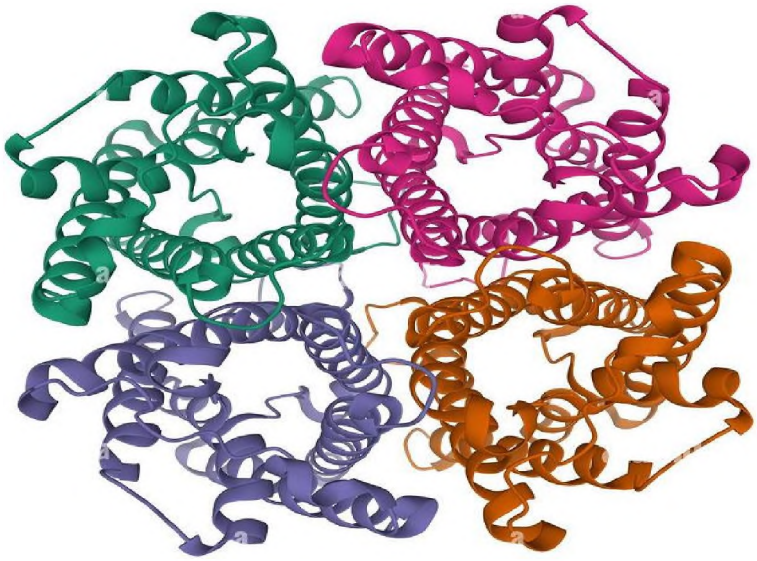


Figure 5. Quaternary Protein structure[10]

[10] https://www.alamy.com/stock-photo/quaternary-structure.html?sortBy=relevant

CHAPTER THREE: PROBLEM FORMATION AND CORRELATED WORK

3.1 Problem Formation

Protein structure prediction is one of the most popular subproblem in the bioinformatics. Predicting 3D structure is still unresolved problem because of the amino acids form unlimited structural folds. Researches have found out four protein types but protein structure can be formed by unlimited amino acid sequence alone and with properties then it's count is beyond so research find it hard to predict 3D structure. Moreover, the amino acid sequence and structure of protein can be formed in various databases like in the gene database, for example, GenBank has more than 2 billion sequences. However, there are currently over 160K protein structures that have been deposited into the RCSB. Even though we have large database of the sequence. If the experiments performed in the laboratory will cost more than $200,000 and time consuming to form each sequence.

Various research has already used to predict protein secondary structure with sequence of amino acids. Protein secondary structure prediction accuracy can be determined by the second structure is correctly. Because protein tertiary structures are categorized into various structural folds based on how secondary structure elements (helices and sheets) are packed and permuted, solving the secondary structure prediction problem is crucial (Yang, Gao, Wang, Heffernan, Hanson, Paliwal & Zhou, 2016). Accuracy of the secondary structure can be predicted by mainly two major category.

- Dictionary of Second Structure of Protein (DSSP): The secondary structure assignment approach, which automatically assigns secondary structure facing eight states based on hydrogen-bonding patterns, is known as Dictionary of Secondary Structure of Proteins (DSSP). The class designations are: H for alpha helix, B for residue in an isolated beta bridge, E for extended strand, G for three-helix, I for five-helix, T for hydrogen-bonded turn, S for bend, and L for loop (Kabsch & Sander, 1983; Lin, Lanchantin & Qi, 2016.
- Second Structure of Protein SSP: Helix, sheet, and coil make up the secondary structure of proteins, which will be a common condensing of eight states of the DSSP into three states (SSP). The most widespread convention asks for the designations of helix as G (310 helix), H (a-helix), and I (p-helix), sheet as B (isolated bridge), and E (extended sheet), and any extra states as coils.

Our project will use protein properties along with amino acids properties sequence to predict the structure of protein structure and train the model. Amino acids are classified into three types of classes: charged amino acids residues, polar amino acids and non-polar acids.

- Charged amino acids: Because to the positive or negative charge on their side chains, these amino acids can interact electrostatically. They are made up of arginine, glutamate, aspartate, histidine, and lysine.
- Thermal stability: One of the elements that can be utilized to deduce a protein's three-dimensional structure is its thermal stability. A protein's thermal stability can be determined experimentally by heating it up and observing how its

structure and function change. When a protein is heated, it goes through a sequence of structural changes that can be seen using several spectroscopic methods such circular dichroism, fluorescence, or NMR. These alterations can be used to calculate the protein's melting temperature (Tm), or the temperature at which half of the protein molecules in a sample have undergone denature. Overall, a protein's thermal stability is an essential variable that can provide vital details about its structure and stability, information that can be helpful in a variety of applications, such as drug development, biotechnology, and structural biology.

- Nonpolar amino acids: These amino acids' hydrophobic side chains prevent them from interacting with water. Tryptophan, alanine, valine, leucine, isoleucine, proline, phenylalanine, and methionine are all present in them. In general, nonpolar side chains, as opposed to polar or charged ones, increase the heat stability of amino acids. This is due to the fact that nonpolar amino acids interact with water molecules less frequently, making them less prone to denature or unfold under high temperatures.
- Polar amino acids: These amino acids can interact with water due to their hydrophilic side chains. Serine, threonine, cysteine, asparagine, and glutamine are some of them. Nevertheless, polar and charged amino acids are frequently more susceptible to heat denaturation due to their interactions with water molecules and other polar or charged molecules in their environment. As a result,

the protein's structure and function may change, which may cause a reduction in activity or even protein aggregation.

In our project, we will use charged properties and thermal stability properties for the deep learning model.

3.2 Correlated Work

Various methods have been already used to predict the protein secondary structure with different accuracy have been achieved by using training such as SPINE-X, MUST-CNN, DeepCNF, SPIDER2 and so on.

SPINE-X (Zhang, Yang & Faraggi 2011) applied deep learning technique to predict the protein structure with the properties of amino acids. Using secondary structures that PSSM and PPAA mutually predicted, solvent accessibility was anticipated. Subsequently, smaller structure projections and solvent accessibility predictions were made using PSSM and PPAA, along with torsion angles.

SPIDER2 (Rost & Sander, 1993) applied deep learning networks with three iterations to improve the accuracy of secondary structure parallely. This method was applied three times to gradually improve solvent accessibility, backbone torsion angles, and secondary structure.

DeepCNF (Wang, Peng, Ma & Xu, 2016) is known as Deep Convolutional Neural Fields was the first approach to predict the secondary structure by stacking deep cnn mutually with random conditional field model on top as the output layer. It was used to predict complex sequence such as the long-range sequential information and adjacent labels interdependencies.

MUST-CNN is a multilayer shift-and-stitch technique is convolution neural network which uses sequence to predict the protein structure. It is a 1D classification system that predicts at the amino acid level using protein sequence input. (Lin, Lanchantin, & Qi, 2016) implemented the algorithm which shifts the amino acid sequence based on pooling in each layer and combine them to get deep model for every amino acid.

Deep Convolutional and Recurrent Neural Network (DCRNN) improves the secondary structure by utilizing a stacked deep learning framework. This model consists of four structures, one feature embedding layer, CNN layer, bidirectional gated recurrent unit (BGRU) layers, and two fully connected layers. DCRNN achieved 73.2% accuracy on the dataset CB6133 (Li, Yu, Shahabi & Liu, 2018).

CHAPTER FOUR: ARTIFICIAL NEURAL NETWORK(ANN)

4.1 Machine learning

It primarily focuses on the development of new algorithms and models which makes computer to think and decide for the task to improve its performance by learning from data. It can used to create programs that analyze and learn from large datasets identify patterns and makes accurate predictions or decisions without being explicitly programmed to do so. It is widely used in many day-to-day fields such natural language processing, patterns prediction, healthcare sector etc. Frank Rosenblatt (1957) developed first machine learning algorithm which was perceptron which simulates the process of learning through trail and error.

There are three types of machine learning: supervised learning, reinforcement learning and unsupervised learning.

Supervised learning is when a model will be trained on a labeled dataset, where each data point will be associated with the known target variable(response or label). The algorithm must learn a mapping from the input features to the target variable as part of supervised learning in order to make accurate predictions. Using a training set, or collection of labeled examples, the algorithm learns the relationship between the input features and the desired outcome. Supervised learning includes algorithms such as linear regressions, random forest, decision tree, neural networks, logistic regression and support vector machines(SVMs) are used for predicting the output. Reinforcement learning, a branch of machine learning, teaches us how to select activities that will maximize a reward signal. The goal of the agent is to find a strategy that maximizes the

predicted cumulative benefit over time. The agent learns by experiencing rewards or penalties for each action it takes. It is used to solve a wide range of problems which includes games, robotics and self driving. Reinforcement algorithm include Q-learning, policy gradients, deep reinforcement learning, and actor-critic methods which are quite popular. Unsupervised learning is another machine learning approach where model will be trained on the unlabeled dataset which no target variables. It is used to design and identify patterns or structure of the data, like dimensionality reduction or clustering. It is used in various fields such as anomaly detection and image or text analysis. As a preprocessing step for supervised learning approaches, it can also be used to reduce the dimensionality of the data or locate clusters of similar data points that can be used to build a classifier or regression model. K-means is a widely used algorithm for unsupervised learning. Below figure 6 illustrates machine learning.

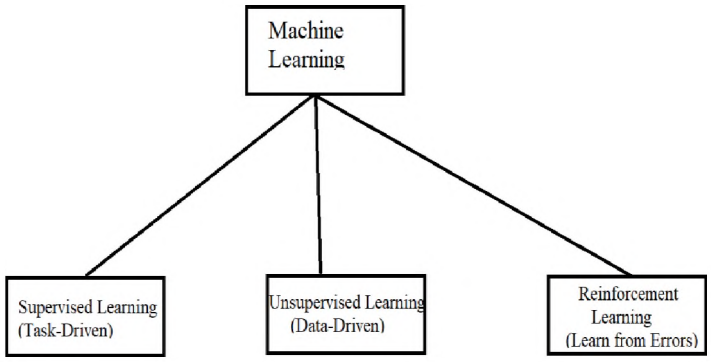


Figure 6. Machine learning types[11]

[11] https://www.potentiaco.com/what-is-machine-learning-definition-types-applications-and-examples/

4.2 Artificial Neural Network

An artificial neural network (ANN) is the most widely used artificial intelligent tools which is used for research domain. It is type of computational model that is based on the design and functionality of neural networks which are present in the human brain. It is a specific type of machine learning algorithm that can be taught to recognize patterns in data and can be used to perform a variety of tasks, including classification, regression, and prediction tasks. It has various interconnected layers of artificial neurons which are also known as nodes or units. Each neuron takes one or more inputs and performs simple computation to produce output. The outputs of some neurons are connected as inputs to other neurons, forming a network of interconnected nodes. Each layer has a different number of neurons or nodes. Input layer consists of total number of nodes which will decide of total features in the training data. It doesn't matter how many number of nodes does hidden layer contains because the number of output layers will be decided by the labels in the input data or number of classes. Figure 7 illustrates the structure of Artificial neural network. ANN has been successful in various applications, but they can have some limitations like they required large amount of labeled data for training purpose, chances of overfitting as well. The weights and biases between the neurons are learned through the process known as training, in such a way that the parameters to minimize the errors between the actual output and predicted output.

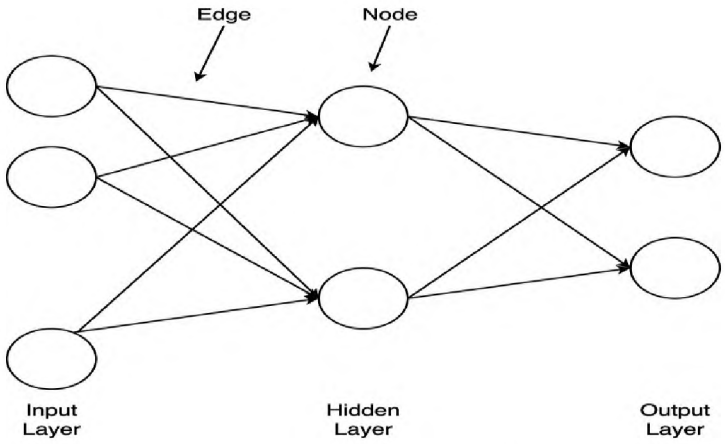


Figure 7. Structure of ANN[12]

There are several types of ANNs such as: Feedforward neural networks, Convolutional Neural Networks (CNNs), Autoencoder, Recurrent Neural Networks (RNNs) and so on. But in our project we have mainly focused on Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs).

4.2.1 Convolutional Neural Network architecture (CNN)

CNN is one of the most widely used method in deep learning algorithm which is used in our daily life. It will input image data, weights and biases as learnable to differenent objects/aspects in the image. This type of neural network is widely used in image recognition, video recognition, handwriting identification, natural language processing, etc. It's method created to label word properties on text data, like the part of speech or category of a named object. It can be applied in bioinformatics for analysis of protein sequence, let us assume the techniques, which treat each amino acid as a word and each protein chain as a sentence are easily transportable. CNNs and

[12] https://towardsai.net/p/l/understand-the-fundamentals-of-an-artificial-neural-network

feedforward neural networks are similar, but CNNs have additional layers designed to process image data. These layers include convolutional, pooling, and fully connected ones. Convolutional layers use the input image as a intial point and process it through different filters (also known as "kernels") to produce a set of feature maps that highlight various aspects of the image, such as edges, corners, and textures. By altering the weights of the filters that it gains during training, the network learns to recognize different feature types to build the model. Figure 8 illustrate sample CNN model.

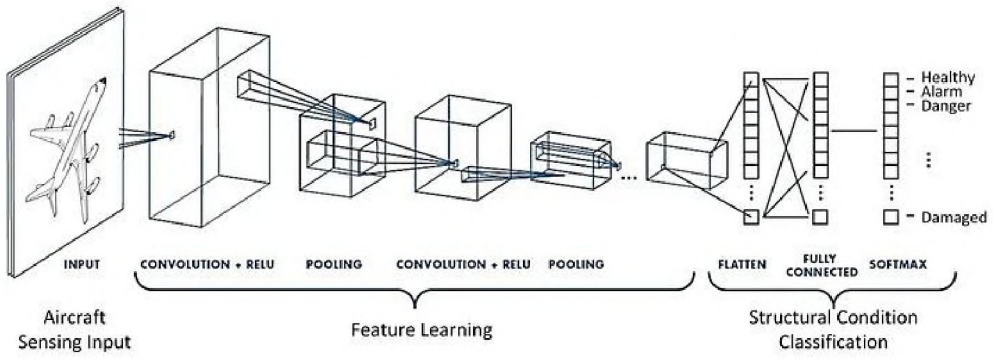


Figure 8. Sample CNN model[13]

CNN model has multiple layers which are combined together to output of the entire network. The above figure shows the architecture of the CNN model. Let's discuss each layer.

- **Convolutional layer**: It is used for image recognition by converting into 3D matrix with height * width * depth. It comes from the three-color channels i.e., (Red-Green-Blue channels) also known as RGB values equal to the number of

[13] https://medium.com/analytics-vidhya/introduction-to-convolutional-neural-network-6942c189a723

filters. A matrix is used to refer for the filter that shares the input weights. It is stacked to form deep neural network in which each layer learns more complex features than the pervious layer. The output of the convolutiaon layer is then passed through an activation layer known as ReLU(Rectified Linear Unit) which include non-linearity and improve the model's ability to describe non-linear interactions between input and output. If the input image couldn't fit in the filter then zero-padding technique is used to pad images with zeros.

- **Pooling layer:** This function comes after convolutional layer which continuously minimize the spatial size of the image by reducing the computation time and parameters to avoid the overfitting problem by discarding the irrelevant or redundant information. There are two types of pooling used in CNNs: max pooling and average pooling. Max-pooling has a filter which is applied on input feature and the maximum value within the filter region is considered as output. This process with reduce the size of the feature map while preserving the most prominent features. The average pooling will consider the average of the values in the filter region which can enhance the feature map's smoothness and lessen the impact of outliers.
- **Dropout:** It is technique which used to prevent overfitting in the model and improve the performance of the model. It is usually in the decimal number form. Dropout is applied on the fully connected layers which contains convolutional and pooling layers in it. Because in fully connected layers might have large number of parameters which can lead to overfit the model. It makes model to

be more robust for large dataset. Excessive dropout for the model can reduce the capacity of the model and can cause underfitting which is quite common for smaller datasets. Thus, dropout rate need to be determined based on the dataset size.

- **Fully connected:** It is used to map the features extracted from the pooling layer and convolutional layer to the output classes. It is present at the end of the CNN. It consists of multiple neurons which are actually depends on the task and complexity of the data. The number of output classes in image classification tasks is equal to the number of neurons in the last fully connected layer, and the probability distribution over the classes is calculated using the softmax function.
- **Softmax:** It is used in the last fully connected layer for multi class classification tasks. It performs computation for the probability distribution of the output classes by normalization the exponential values of the input scores. It provides the results in the form of probabilities.

4.2.2 Recurrent Neural Network Architecture (RNN)

CNNs are more suited to processing data with a grid-like layout and extracting spatial features, RNNs are better suited to processing sequential data and capturing temporal dependencies. It mainly works by storing the memory at each step and used to process the current input to generate the output. It is usually maintained as vector or set of vectors and its value depend on the pervious state and current input. Figure 8 illustrate the architecture of RNN. The data will loop in the cell A where the information will pass from one state to next for the cell. Figure 9 illustrates network H's loop is

unfold to pass a message. Hence, RNN can retain a short-term dependency. However, it is necessary to persist a piece of long-time information. A famous architecture of RNN is Long Short Term Memory (LSTM), which is designed to remedy the short-term issue in RNN.

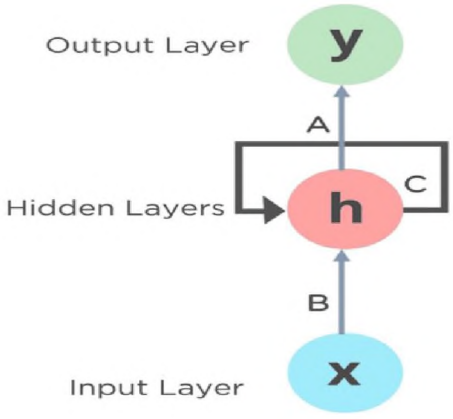


Figure 9. RNN structure[14]

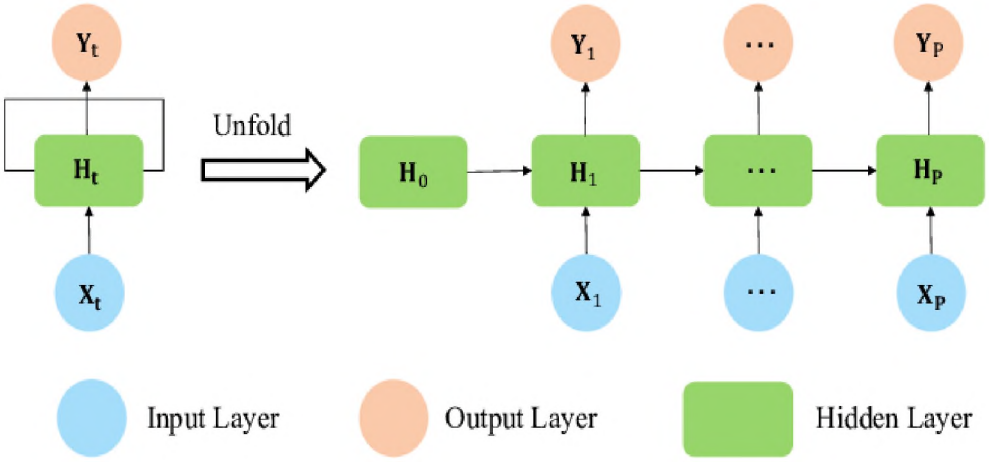


Figure 10. Structure of unfold RNN [15]

[14] https://www.simplilearn.com/tutorials/deep-learning-tutorial/rnn

[15] https://www.researchgate.net/figure/The-folded-and-unfolded-structure-of-recurrent-neural-networks-1-RNN-Similar-to-a_fig5_341639694

The input sequence is represented as a series of input vectors in an unfolded RNN, and each input vector is fed into the network at a separate time step. By integrating the current input with the prior state and using a nonlinear activation function, the network updates its internal state at each time step. With a different set of weights, the network's output can be determined at each time step depending on its internal state and input. Long Short-Term Memory (LSTM), a type of Recurrent Neural Network (RNN), is designed to circumvent the vanishing gradient problem and improve the network's ability to learn and store long-term dependencies in the input sequence. LSTM cell structure is illustrated below figure 11. Gated recurrent unit(GRU) is a type of variation of LSTM with design same and produces almost same results.

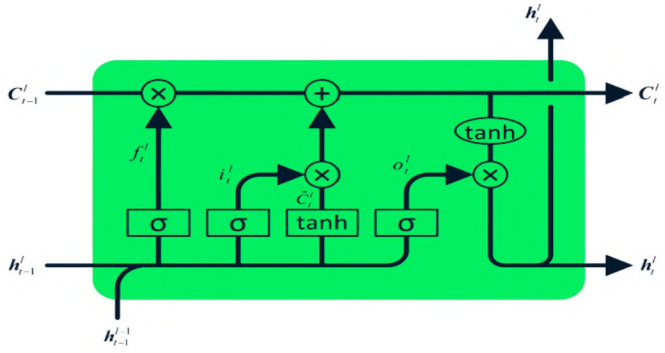


Figure 11. LSTM Cell structure[16]

[16] https://www.datacamp.com/tutorial/tutorial-for-recurrent-neural-network

CHAPTER FIVE: PROTEIN DATASETS

5.1 Datasets

A collection of data that has been organized, formatted, and is prepared for analysis or machine learning is known as a dataset. Many shapes and types of data, including images, text, audio, video, and numbers, can be found in datasets. Many sources, including public databases, online archives, and private collections, offer datasets for retrieval. Many datasets are available for free usage and download. For our project, We have used two large protein datasets in our experiment which are CB6133 and CB513. The first one is used as a training dataset, and the second one is our test data.

5.1.1 CB6133 dataset

This dataset was produced with PISCES CullPDB (Wang and Dunbrack, 2003). It contains 6133 protein in which 5600 proteins used for training purpose, for validation purpose 256 proteins are used and finally for testing 277 proteins are used. This is non-homolgous protein structure dataset and sequence dataset to train the models. It can be reshaped into 5600 proteins * 700 amino acids * 57 features which fits our project model.

5.1.2 CB513 dataset

This data is obtained from (Lin, Lanchantin, & Qi, 2016) which is derived from Zhou and Troyanskaya in 2014. It is a public dataset used for testing. CB513 is a public benchmark dataset that is used for testing. Sequence with more than 25% in CB513 are removed from CB6133. One of the protein sequences contains around 696 amino acids. It contains 513 non-homolgous protein sequences for the testing.

For CB6133 and CB513 dataset contains 57 features in it. The 57 features in the data sets, CB6133 and CB513, are described in Table 3. There are few unknown marks such as the letter 'X' in the amino acid sequence denotes the unknown amino acid and 'Noseq' marks the end of the protein sequence in both amino acid residues and secondary structure. In our project, we will use the protein charge property and thermal stability as additional features in both training and testing.

Features	**Field Description**
[0, 22]	Amino acid residues in the order : 'A', 'C', 'E', 'D', 'G', 'F', 'I', 'H', 'K', 'M', 'L', 'N', 'Q', 'P', 'S', 'R', 'T', 'W', 'V', 'Y', 'X', 'NoSeq'.
[22, 31]	Secondary structure labels with the sequence of 'L', 'B', 'E', 'G', 'I', 'H', 'S', 'T', 'NoSeq'.
[31, 33]	C-terminal and N-terminal.
[33, 35]	Thermal stability.
[35, 57]	Sequence profile.

Table 1. Dataset Field Description

CHAPTER SIX: CNN AND RNN MODELS

6.1 CNN model

In this model, we will use the amino acid sequence along with other properties to train our model to predict the protein sequence. These properties are labels/classes in the model. Let's discuss the different CNN models with additional features in the training phase. Below figure illustrates the CNN model with only protein sequence. There are various existing research which uses protein sequence to predict the secondary structure and model accuracy varies for it. In the CNN, we utilize the amino acid sequences and two additional protein properties as features in training where we will use the convolutional, dropout, dense, batch normalization to train the model. In the below figure 12 we will use only protein sequence for our CNN model.

```
Layer (type)                 Output Shape              Param #
=================================================================
conv1d (Conv1D)              (None, 17, 256)           27136

batch_normalization (BatchN  (None, 17, 256)           1024
ormalization)

dropout (Dropout)            (None, 17, 256)           0

conv1d_1 (Conv1D)            (None, 17, 128)           98432

batch_normalization_1 (Batc  (None, 17, 128)           512
hNormalization)

dropout_1 (Dropout)          (None, 17, 128)           0

conv1d_2 (Conv1D)            (None, 17, 64)            24640

batch_normalization_2 (Batc  (None, 17, 64)            256
hNormalization)

dropout_2 (Dropout)          (None, 17, 64)            0

conv1d_3 (Conv1D)            (None, 17, 32)            6176

batch_normalization_3 (Batc  (None, 17, 32)            128
hNormalization)

dropout_3 (Dropout)          (None, 17, 32)            0

flatten (Flatten)            (None, 544)               0

dense (Dense)                (None, 128)               69760

dense_1 (Dense)              (None, 32)                4128

dense_2 (Dense)              (None, 8)                 264

=================================================================
Total params: 232,456
Trainable params: 231,496
Non-trainable params: 960
```

Figure 12. CNN architecture with sequence only

6.2.1 CNN with Charge Property

In this model, we have used the charge property as extra feature with N-terminal and C-terminal which represent negative and positive charge. Figure 13 shows the CNN model with charge property.

```
Layer (type)                 Output Shape              Param #
=================================================================
conv1d (Conv1D)              (None, 17, 256)           27136

batch_normalization (BatchN  (None, 17, 256)           1024
ormalization)

dropout (Dropout)            (None, 17, 256)           0

conv1d_1 (Conv1D)            (None, 17, 128)           98432

conv1d_2 (Conv1D)            (None, 17, 128)           49280

batch_normalization_1 (Batc  (None, 17, 128)           512
hNormalization)

dropout_1 (Dropout)          (None, 17, 128)           0

conv1d_3 (Conv1D)            (None, 17, 64)            24640

batch_normalization_2 (Batc  (None, 17, 64)            256
hNormalization)

dropout_2 (Dropout)          (None, 17, 64)            0

conv1d_4 (Conv1D)            (None, 17, 32)            6176

batch_normalization_3 (Batc  (None, 17, 32)            128
hNormalization)

dropout_3 (Dropout)          (None, 17, 32)            0

flatten (Flatten)            (None, 544)               0

dense (Dense)                (None, 128)               69760

dense_1 (Dense)              (None, 32)                4128

dense_2 (Dense)              (None, 8)                 264

=================================================================
Total params: 281,736
Trainable params: 280,776
Non-trainable params: 960
```

Figure 13. CNN architecture uses charge property.

6.2.2 CNN with Thermal Stability

In this model, we have used another property which thermal stability. Figure 14 represent the structure of the CNN model along with thermal stability.

```
Layer (type)                 Output Shape              Param #
=================================================================
conv1d (Conv1D)              (None, 17, 256)           27136

batch_normalization (BatchN  (None, 17, 256)           1024
ormalization)

dropout (Dropout)            (None, 17, 256)           0

conv1d_1 (Conv1D)            (None, 17, 128)           98432

conv1d_2 (Conv1D)            (None, 17, 128)           49280

batch_normalization_1 (Batc  (None, 17, 128)           512
hNormalization)

dropout_1 (Dropout)          (None, 17, 128)           0

conv1d_3 (Conv1D)            (None, 17, 64)            24640

batch_normalization_2 (Batc  (None, 17, 64)            256
hNormalization)

dropout_2 (Dropout)          (None, 17, 64)            0

conv1d_4 (Conv1D)            (None, 17, 32)            6176

batch_normalization_3 (Batc  (None, 17, 32)            128
hNormalization)

dropout_3 (Dropout)          (None, 17, 32)            0

flatten (Flatten)            (None, 544)               0

dense (Dense)                (None, 128)               69760

dense_1 (Dense)              (None, 32)                4128

dense_2 (Dense)              (None, 8)                 264

=================================================================
Total params: 281,736
Trainable params: 280,979
Non-trainable params: 757
```

Figure 14 CNN architecture uses thermal stability

6.2.3 CNN with both properties

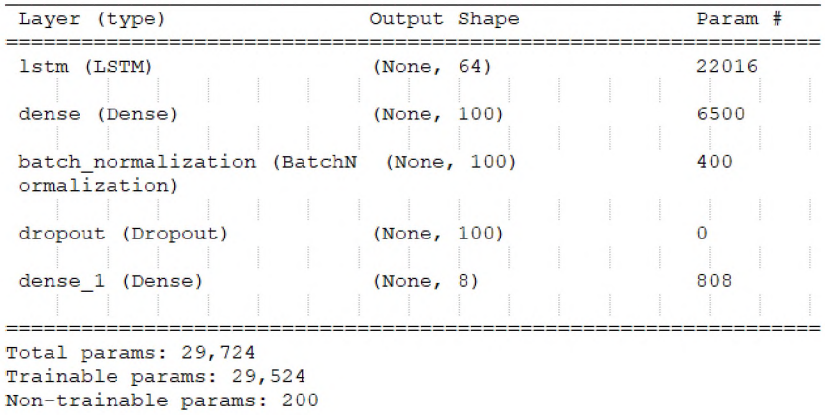
In this model, We combine both the properties mentioned as above into another CNN model. Figure 15 represent the CNN model structure.

```
Layer (type)                 Output Shape              Param #
=================================================================
conv1d (Conv1D)              (None, 17, 256)           27136

batch_normalization (BatchN  (None, 17, 256)           1024
ormalization)

dropout (Dropout)            (None, 17, 256)           0

conv1d_1 (Conv1D)            (None, 17, 128)           98432

conv1d_2 (Conv1D)            (None, 17, 128)           49280

batch_normalization_1 (Batc  (None, 17, 128)           512
hNormalization)

dropout_1 (Dropout)          (None, 17, 128)           0

conv1d_3 (Conv1D)            (None, 17, 64)            24640

batch_normalization_2 (Batc  (None, 17, 64)            256
hNormalization)

dropout_2 (Dropout)          (None, 17, 64)            0

conv1d_4 (Conv1D)            (None, 17, 32)            6176

batch_normalization_3 (Batc  (None, 17, 32)            128
hNormalization)

dropout_3 (Dropout)          (None, 17, 32)            0

flatten (Flatten)            (None, 544)               0

dense (Dense)                (None, 128)               69760

dense_1 (Dense)              (None, 32)                4128

dense_2 (Dense)              (None, 8)                 264

=================================================================
Total params: 281,736
Trainable params: 281,719
Non-trainable params: 17
```

Figure 15. CNN architecture uses two properties together

6.3 RNN model

Like CNN model, We will apply protein properties, charge property and thermal stability and build RNN model. Figure 16 represent RNN model with out any protein property.



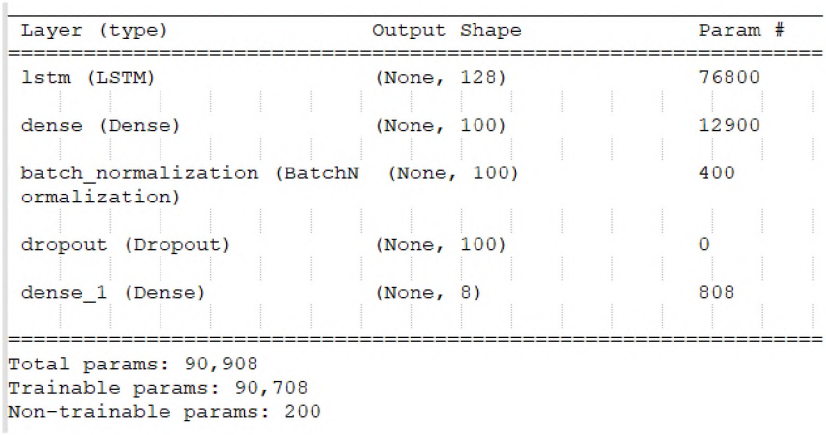
Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 64)	22016
dense (Dense)	(None, 100)	6500
batch_normalization (BatchNormalization)	(None, 100)	400
dropout (Dropout)	(None, 100)	0
dense_1 (Dense)	(None, 8)	808

Total params: 29,724
Trainable params: 29,524
Non-trainable params: 200

Figure 16 RNN architecture with sequence only

6.3.1 RNN with Charge Property

Charge property will be used as an additional feature in the RNN model. Figure 17 represent the RNN model.



Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 128)	76800
dense (Dense)	(None, 100)	12900
batch_normalization (BatchNormalization)	(None, 100)	400
dropout (Dropout)	(None, 100)	0
dense_1 (Dense)	(None, 8)	808

Total params: 90,908
Trainable params: 90,708
Non-trainable params: 200

Figure 17 RNN architecture uses charge property

6.3.2 RNN with Thermal Stability

Now, We will use another feature known as thermal stability to build RNN model. Figure 18 represent the RNN model with thermal stability.

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 128)	76800
dense (Dense)	(None, 100)	12900
batch_normalization (BatchNormalization)	(None, 100)	400
dropout (Dropout)	(None, 100)	0
dense_1 (Dense)	(None, 8)	908

Total params: 90,908
Trainable params: 90,828
Non-trainable params: 80

Figure 18. RNN architecture uses thermal stability

6.3.3 RNN with both properties

In this model, we combine both the properties. Figure 19 shows the RNN model.

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 256)	284672
dense (Dense)	(None, 100)	25700
dense_1 (Dense)	(None, 128)	12928
dense_2 (Dense)	(None, 32)	4128
batch_normalization (BatchNormalization)	(None, 32)	128
dropout (Dropout)	(None, 32)	0
dense_3 (Dense)	(None, 8)	264

Total params: 327,820
Trainable params: 327,756
Non-trainable params: 64

Figure 19. RNN architecture uses two properties together

CHAPTER SEVEN: REQUIREMENT AND TOOLS

7.1 Software Requirements

There are various programming languages to build machine learning model. R and Python are the most popular languages. We build our model using python programming language because it is easy to implement and lot of inbuild libraries and frameworks. List of software and hardware requirements:

- Keras-2.11.0
- Tensorflow-2.11.0
- Numpy-1.24.2
- Scikit_learn-1.2.1
- Matplotlib-3.7.0
- Scipy-1.10.0

7.2 Hardware Requirements

- Operating system: Windows 11
- Processor: 2.42 GHz Intel-core Intel i5
- Memory: 12 GB
- System type: 64-bit operating system, x64-bases processor
- Hard disk: 500 GB SSD

7.3 CNN model parameters

Below are the parameters we used to build CNN model:

- Sequence length: 700
- Loss function: categorical_crossentropy
- Number of features: 21
- Learning rate: 0.0008
- Pooling: max_pooling
- Dropout: 0.45
- Batch dimension: 128
- Epochs: 30
- Activation function: Relu

7.4 RNN model parameters

Below are the parameters we used to build RNN model:

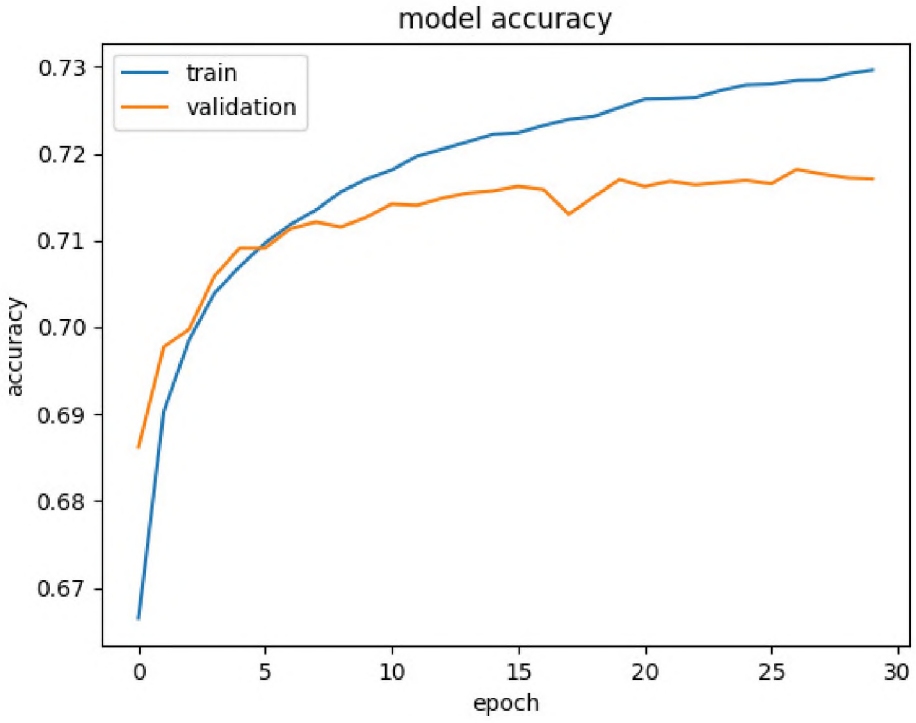
- Sequence length: 700
- Loss function: categorical_crossentropy
- Number of features: 21
- Optimizer: adam
- Learning rate: 0.0008
- Epochs: 30
- Batch dimension: 128
- Activation function: softmax
- Metrics: accuracy

CHAPTER EIGHT: RESULTS AND DISCUSSION

8.1 CNN model results:

In CNN model, during training Each epoch takes around 11 minutes approximately, for 30 epochs it takes around 6 hours to train the one CNN model.

Figure 20 shows the performance of CNN model accuracy, model loss and model mean absolute error for the training and evaluation.



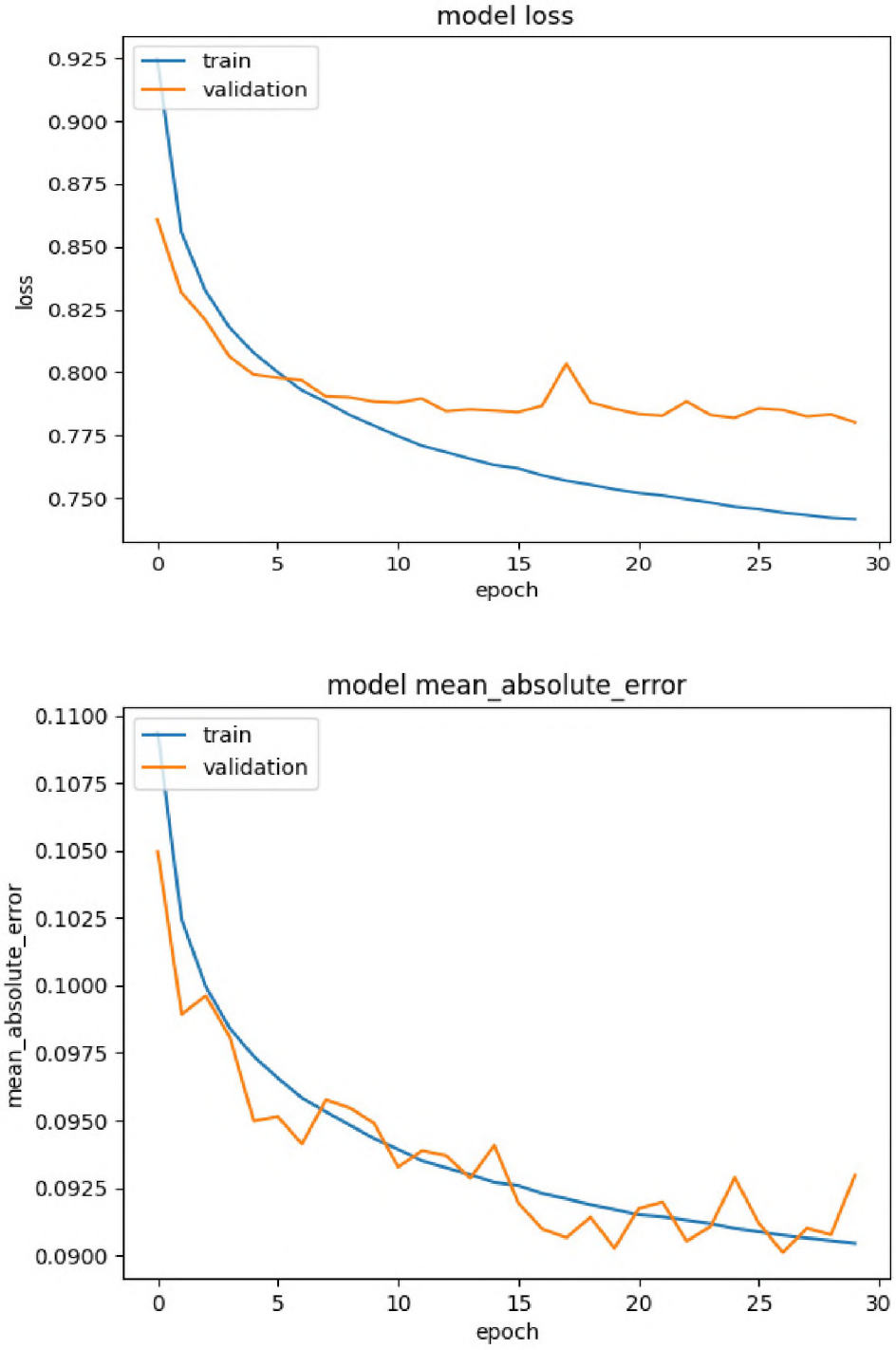
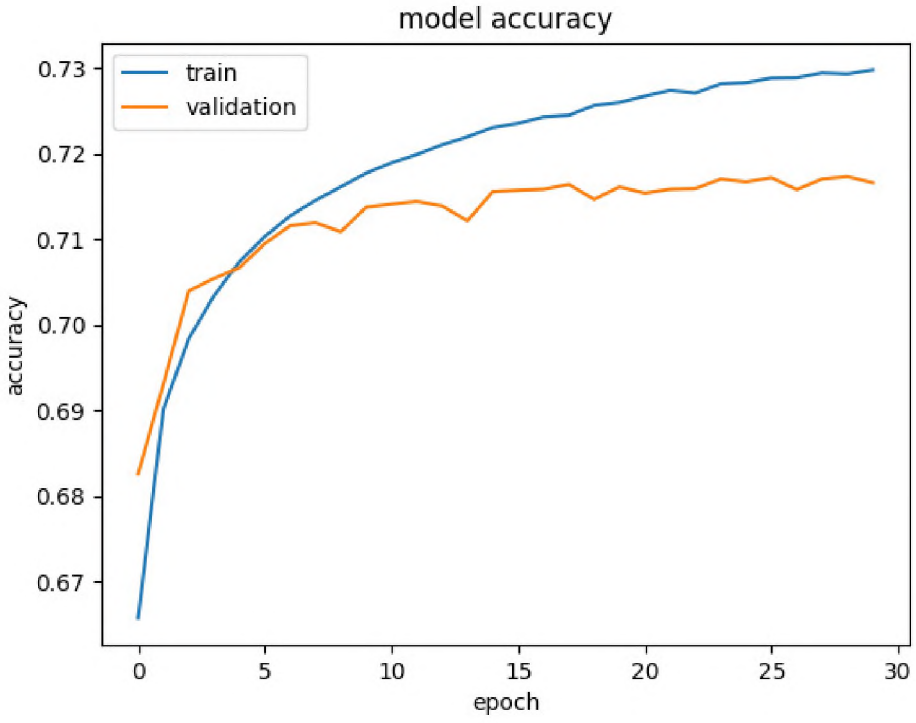
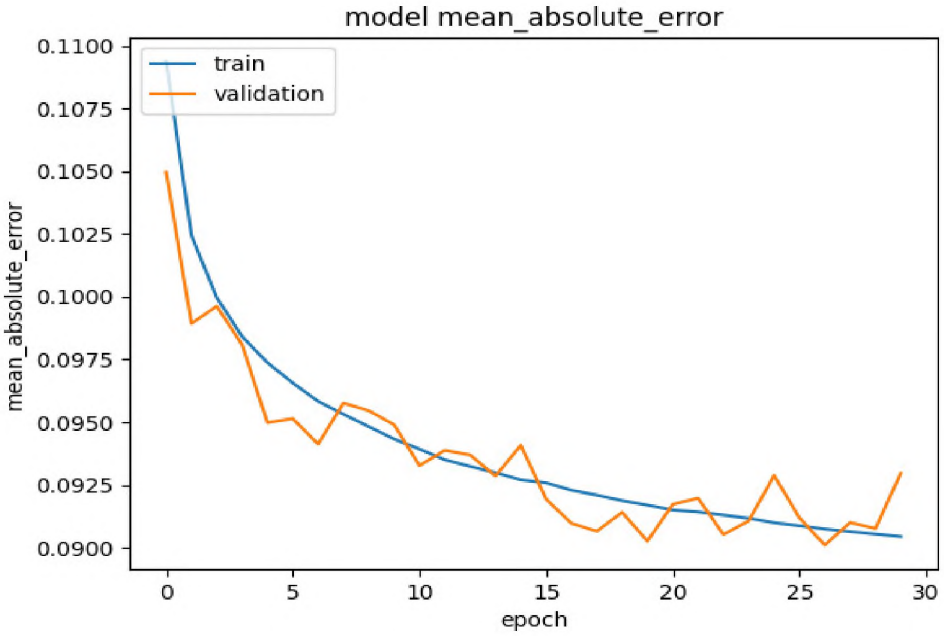


Figure 20. CNN model training results without using any properties

8.1.1 Using Charge Property

After adding the two charge properties i.e., C-terminal and N-terminal, the number of feature change to 23. Figure 21 shows the performance of CNN model accuracy, model loss and model mean absolute error.





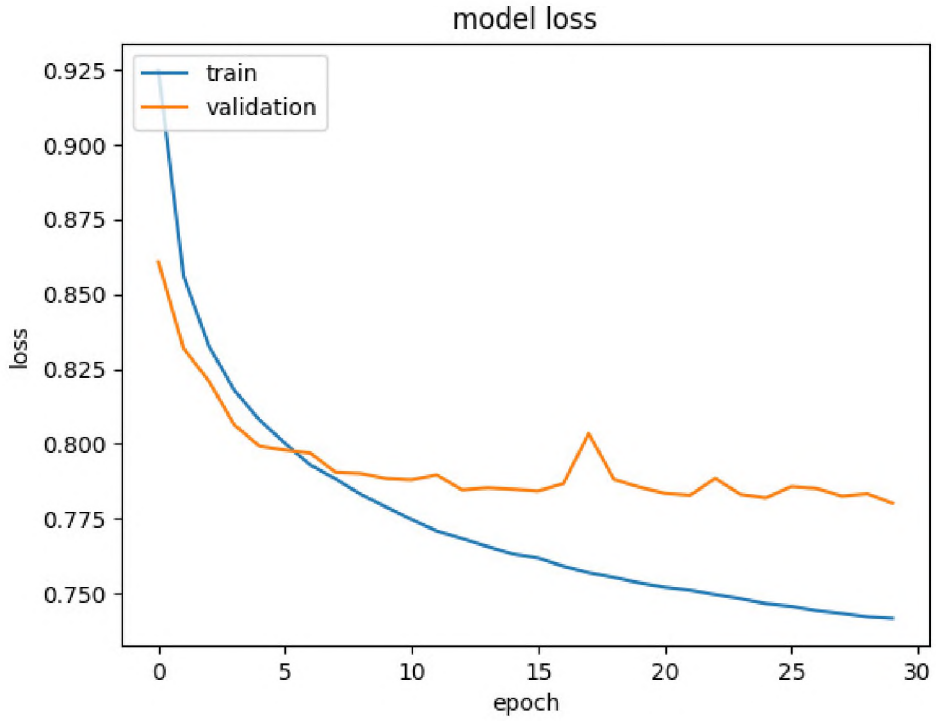
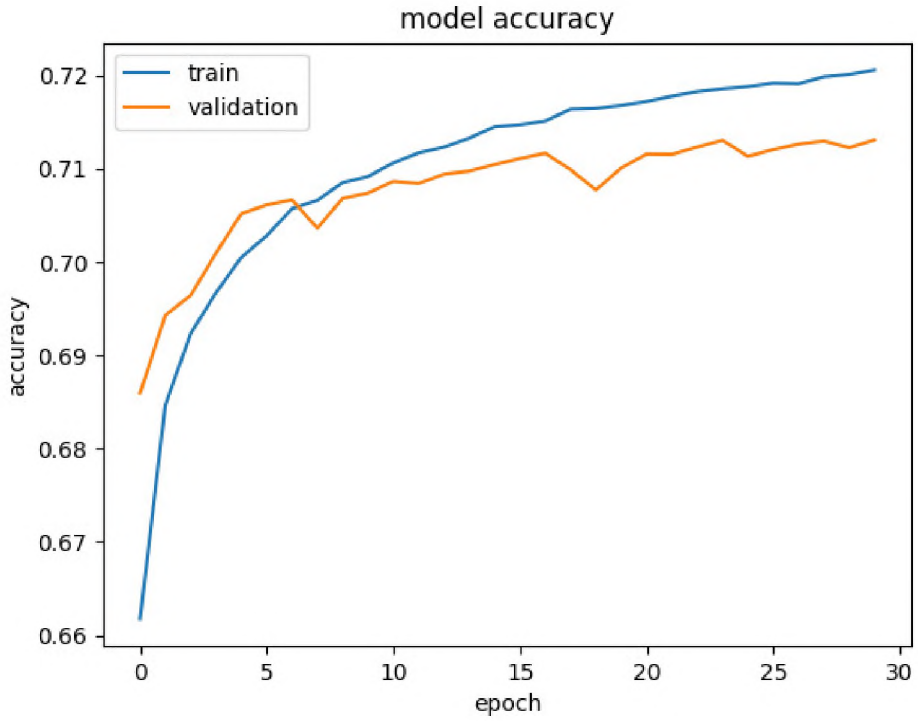


Figure 21. CNN model training results with using electrostatic charge properties.

8.1.2 Using thermal stability

After adding thermal stability to the amino acid sequence then the number of feature is 22 . Figure 22 shows the performance of CNN model accuracy, model loss and model mean absolute error.



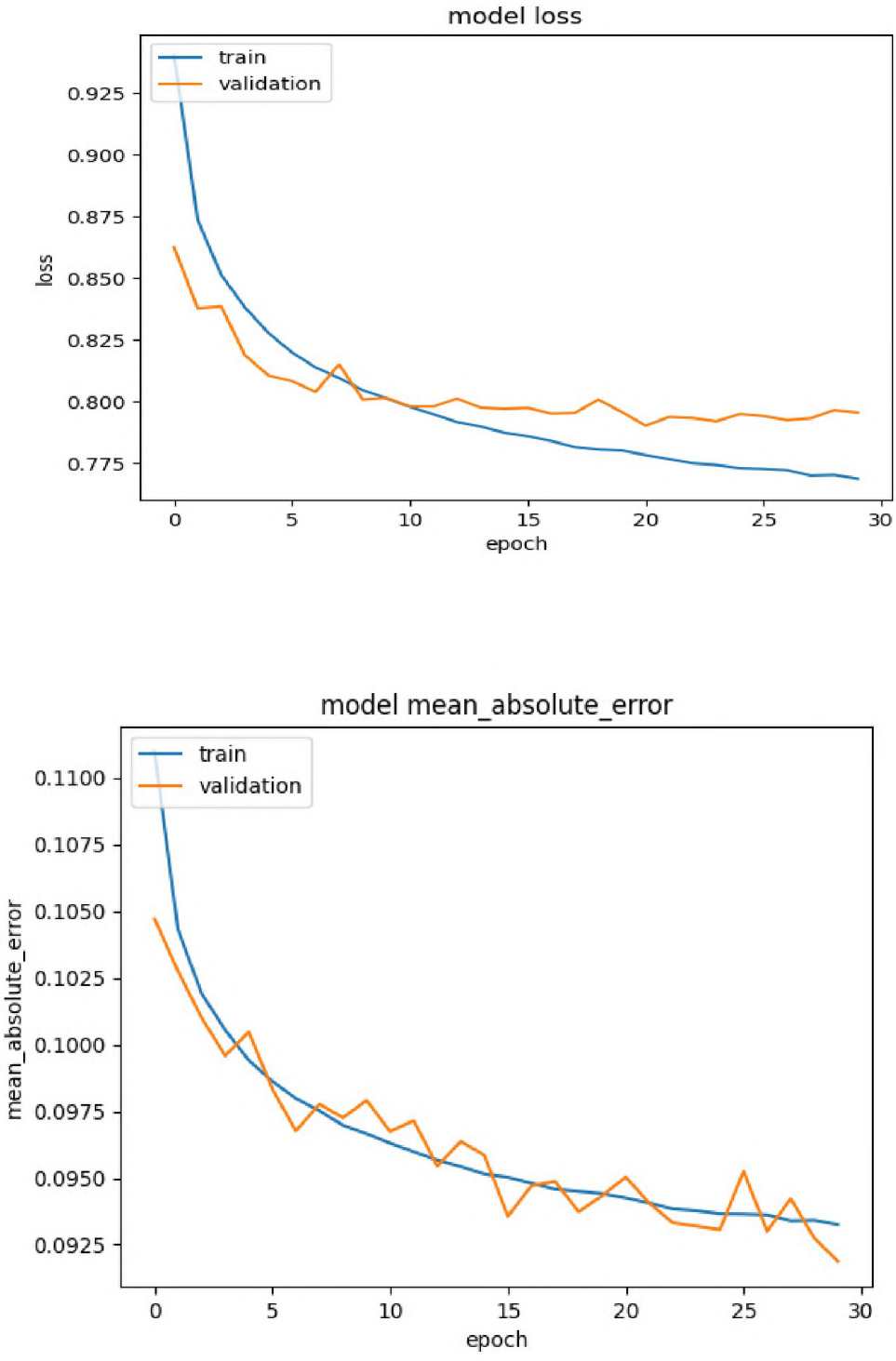
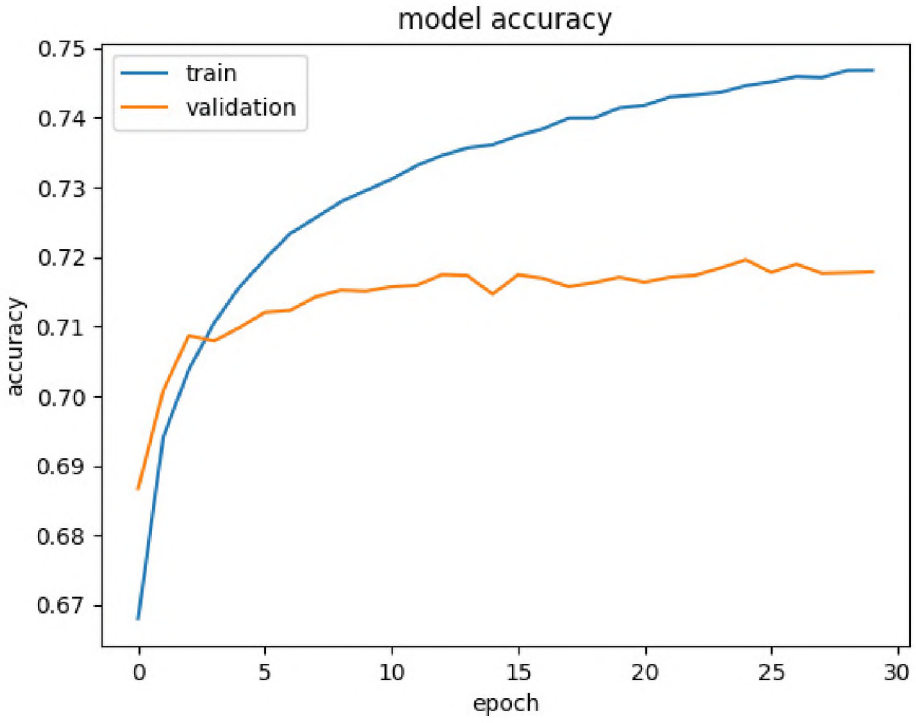
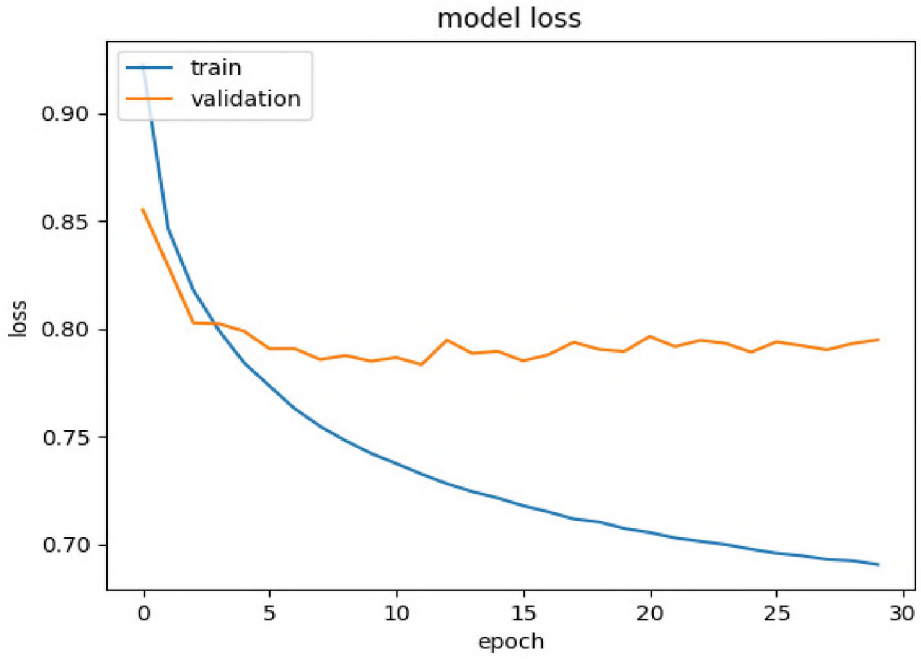


Figure 22. CNN model training results with using thermal stability

8.1.3 Using both Properties

Combining both the properties then the number of feature is 24 with charge properties and thermal stability. Figure 23 shows the performance of CNN model accuracy, model loss and model mean absolute error.





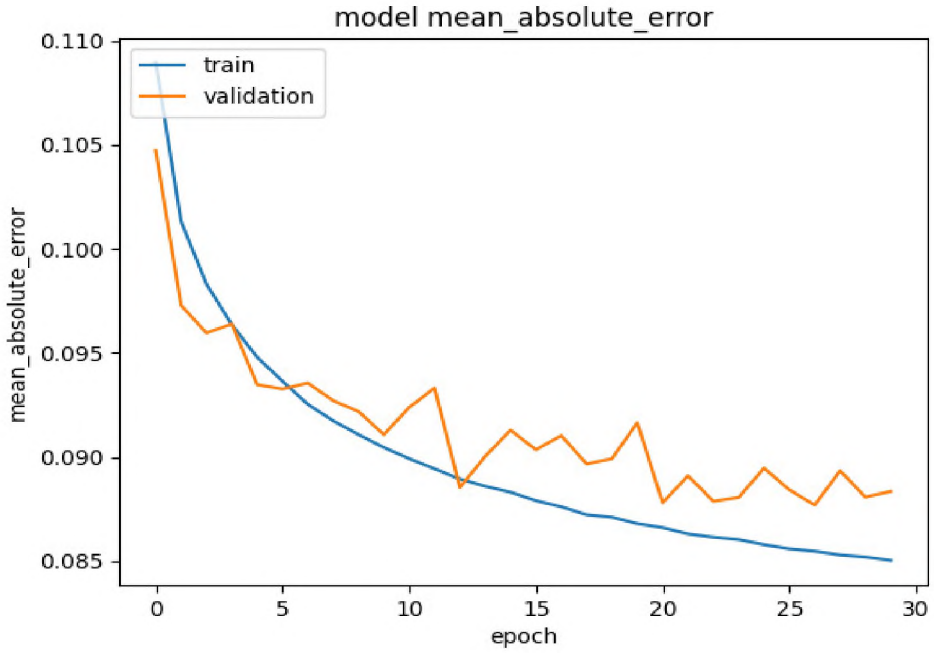
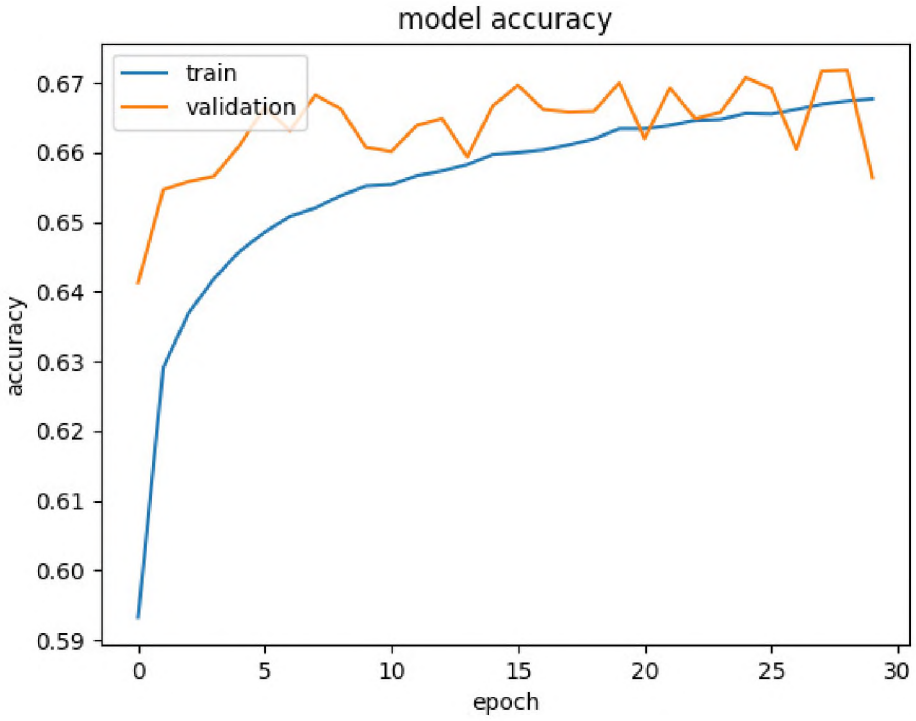


Figure 23. CNN model training results with both properties

8.2 RNN model results

In RNN model, during training Each epoch takes around 10 minutes approximately, for 30 epochs it takes around 5 hours to train the one RNN model.

Figure 24 shows the performance of RNN model accuracy, model loss and model mean absolute error for the training and evaluation.



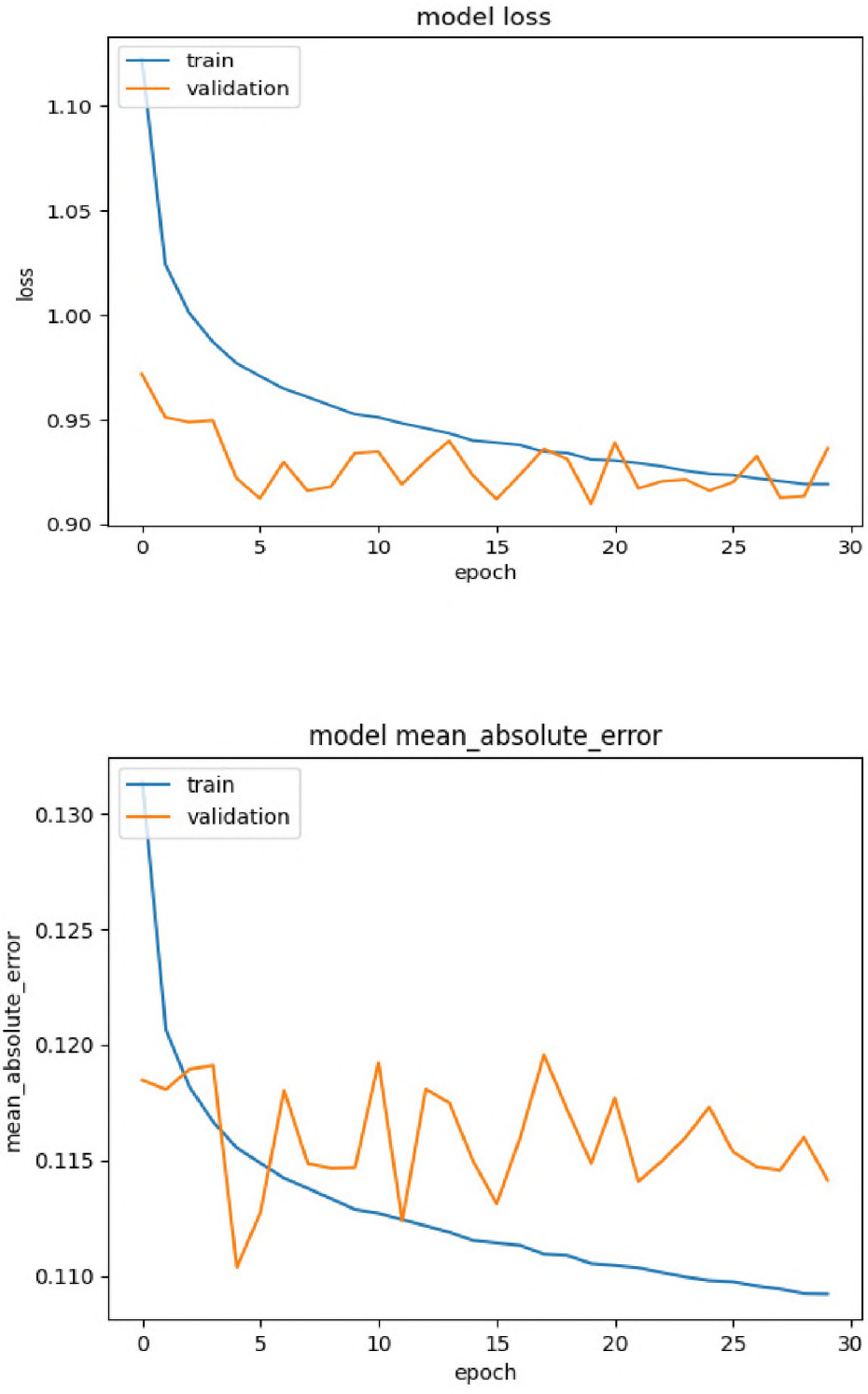
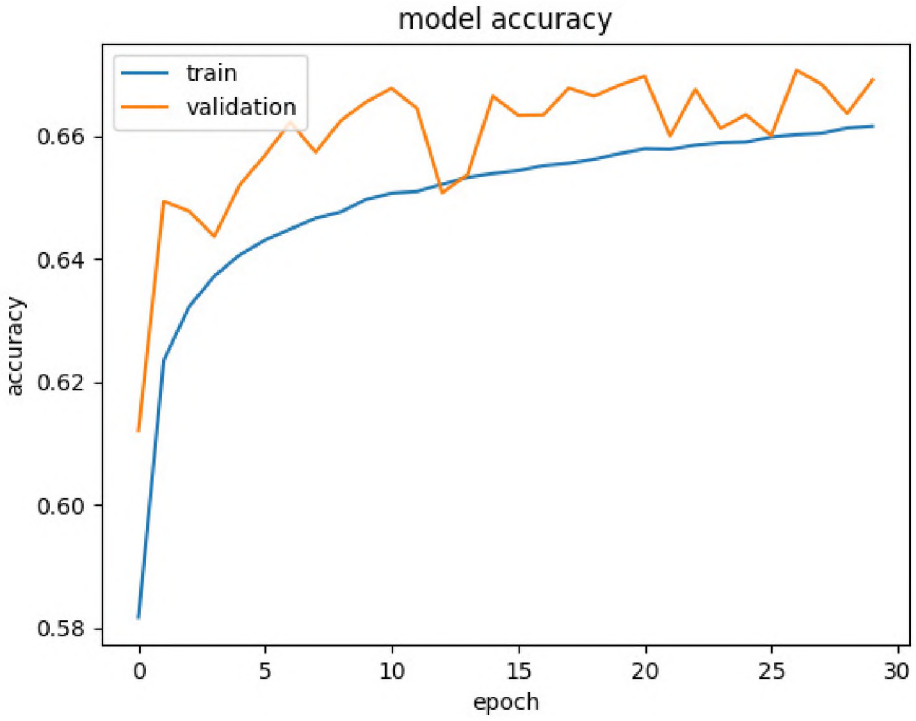
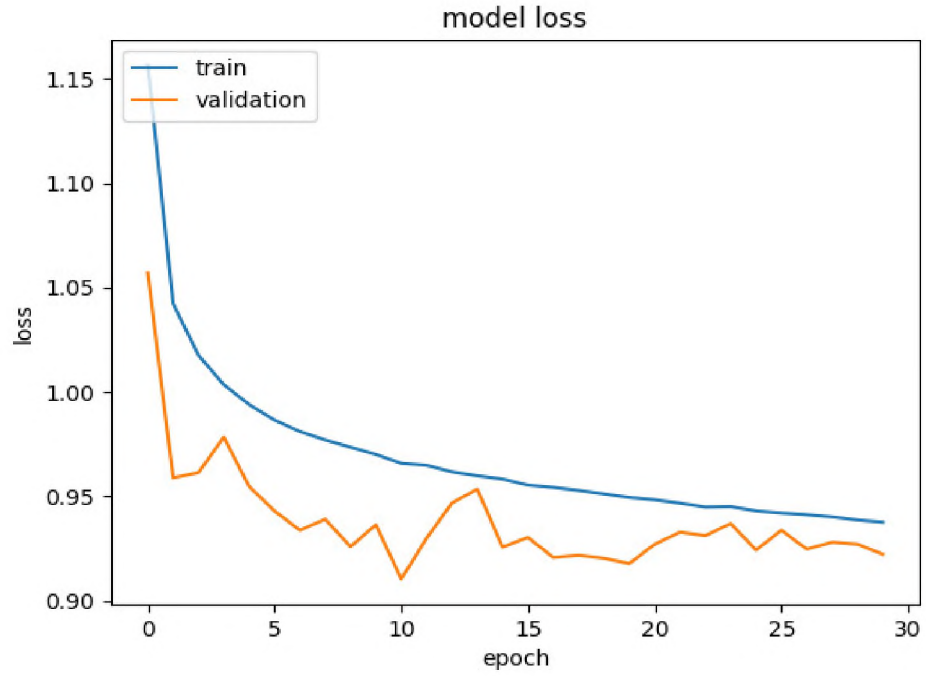


Figure 24. RNN model training results without using properties

8.2.1 Using Charge property

After adding the two charge properties i.e., C-terminal and N-terminal, the number of feature change to 23. Figure 25 shows the performance of RNN model accuracy, model loss and model mean absolute error.





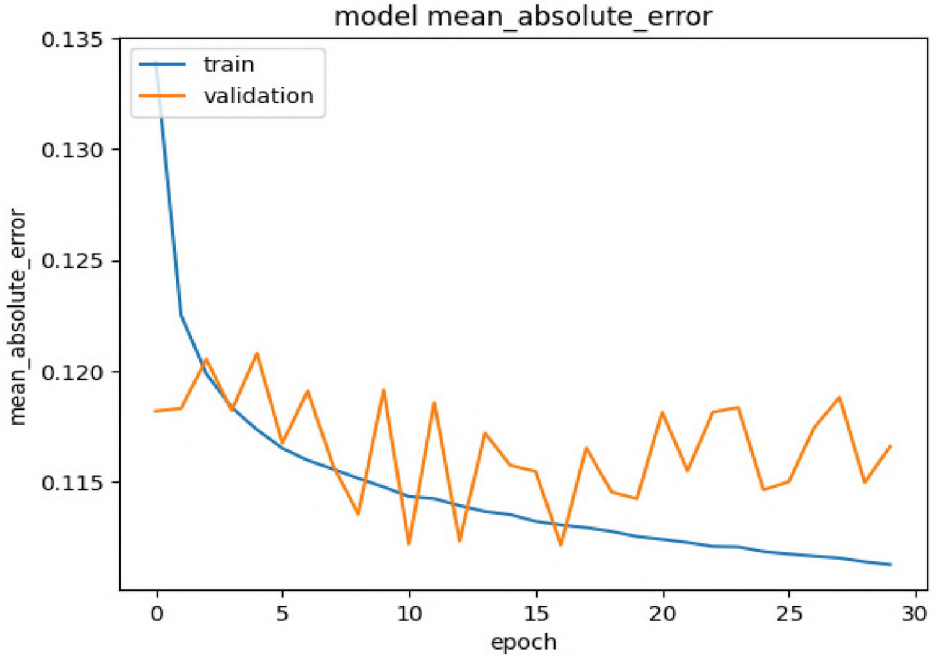
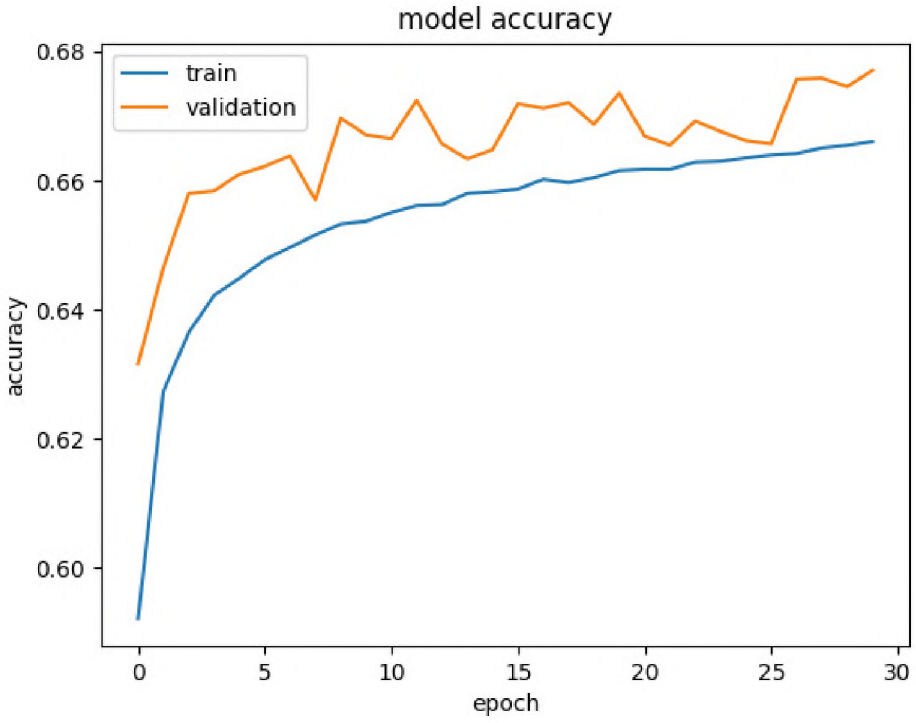


Figure 25. RNN model training results with charge properties

8.2.2 Using thermal stability

After adding thermal stability to the amino acid sequence then the number of feature is 22 . Figure 26 shows the performance of CNN model accuracy, model loss and model mean absolute error.



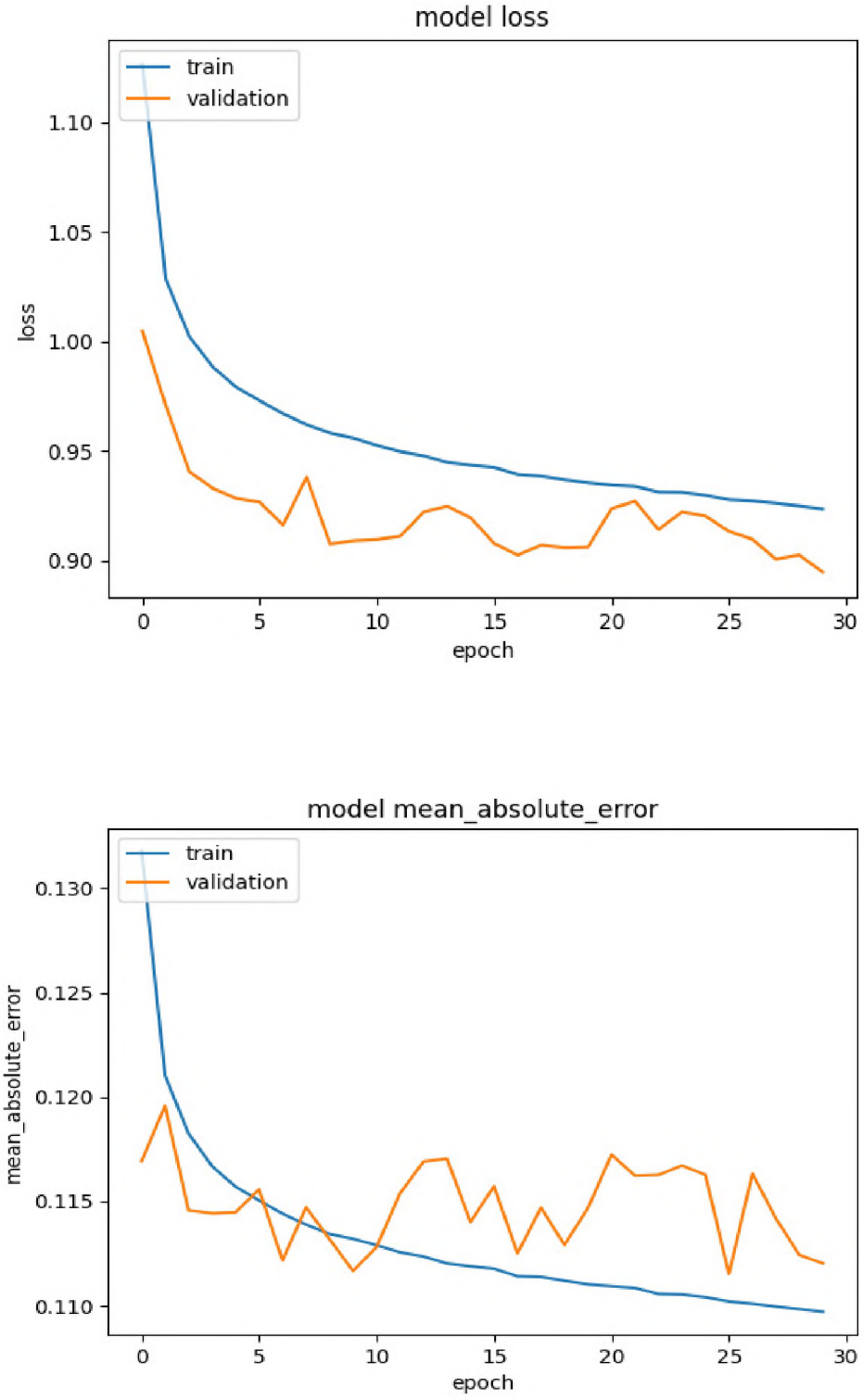
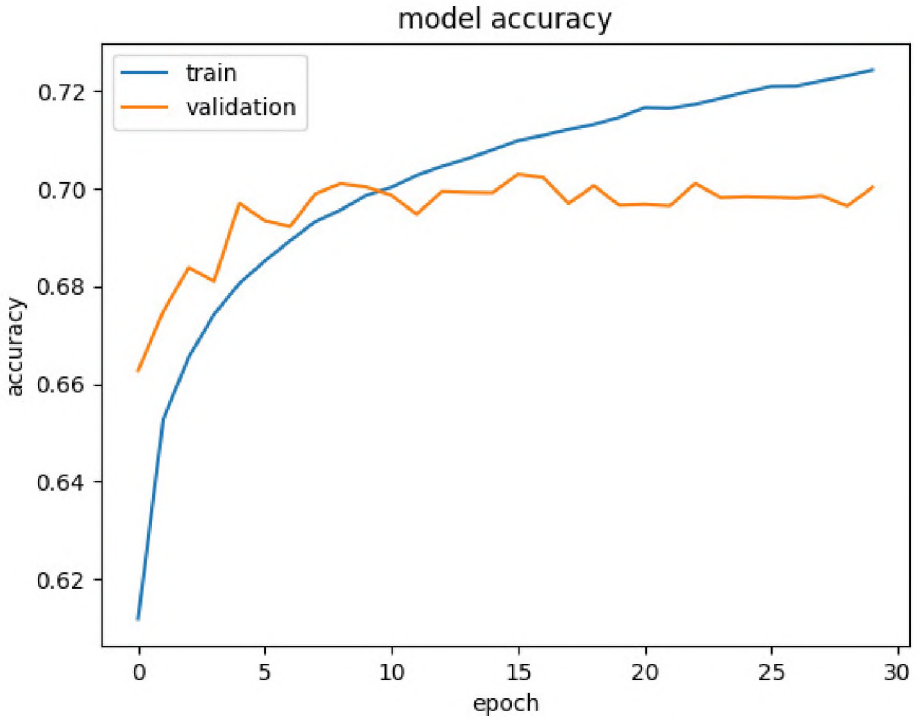
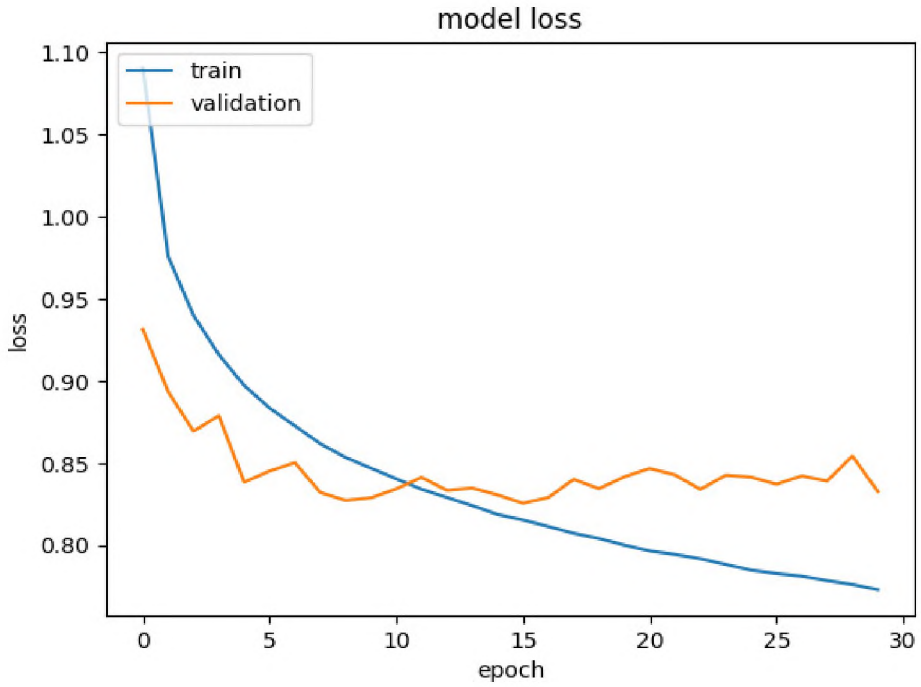


Figure 26. RNN model training results with using thermal stability property

8.2.3 Using both Properties

Combining both the properties then the number of feature is 24 with charge properties and thermal stability. Figure 27 shows the performance of RNN model accuracy, model loss and model mean absolute error.





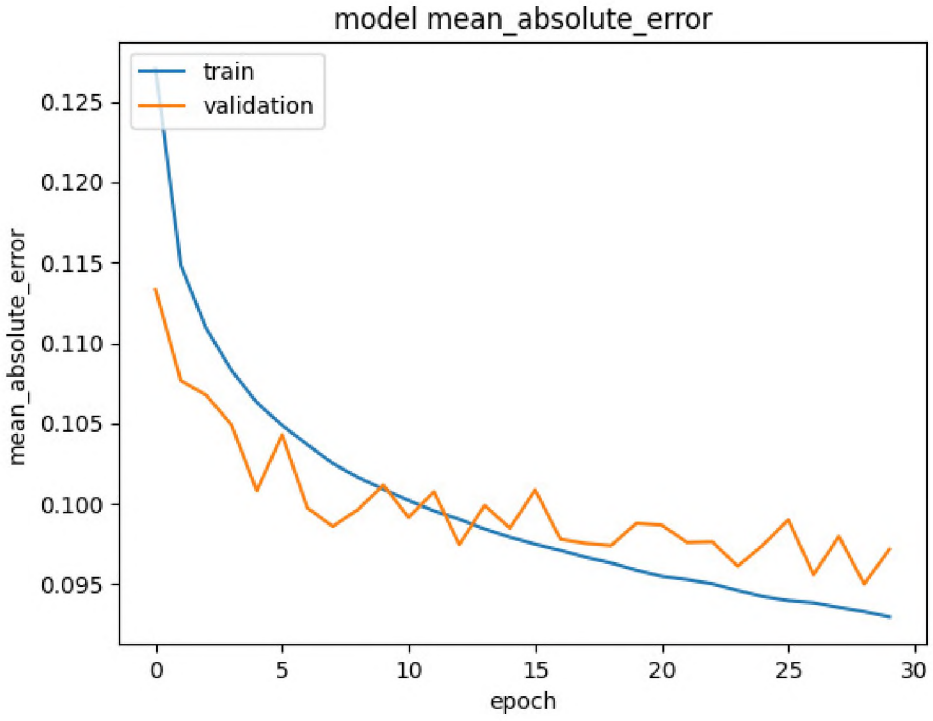


Figure 27. RNN model training results with both properties

Now, we can see the accuracy achieved by both CNN and RNN with CB513 test data is shown in Table 4.

PROPERTIES	CNN	RNN
Without any properties	0.72	0.65
With Charge property and 21 amino acid sequence	0.73	0.66
With Thermal stability and 21 amino acid sequence	0.72	0.67
With Charge property, Thermal stability and 21 amino acid sequence	**0.75**	**0.72**

Table 2. Comparison of the models

Upon comparison we have achieved by both models CNN and RNN, CNN used more time than RNN, but CNN models accuracy is better than any RNN model.

CHAPTER 9: SUMMARY AND FUTURE WORK

9.1 Summary

Protein is required elements for all living things. There are many limitation for finding the structure of protein and learning it helps in many industries like healthcare, etc. According to the records in the gene database, for example, GenBank has more than 2 billion sequences. However, there are currently over 160K protein structures that have been deposited into the RCSB. The biggest challenges in learning protein structures from experiments are cost and time. Usually, solving a new and unique structure costs around $100,000. After discovering a new protein sequence, finding an experimental structure with a lower cost will be challenging. Therefore, accurately predicting the 3D dimension of a protein is a popular research area in computational biology. Protein structure can be divided into four levels: primary, secondary, tertiary, and quaternary. The primary structure of a protein refers to the linear sequence of amino acid residues that make up the protein chain. The secondary structure in protein structure refers to the specific spatial configuration of the polypeptide backbone. The tertiary structure of a protein may have a substantial impact on 3D structure. Current research primarily focus on protein structure prediction by using amino acids and its properties. In our project, we will predict the protein structure from its properties and amino acid sequences by using deep learning technique to build the models. We used two other protein properties in training, the thermal stability and water solvent accessibility. Our architecture produced various results with different.

We have developed RNN and CNN model has got around same accuracy. It is notable that using extra features combined with amino acid sequence has increased the accuracy rather than just amino acid sequence. With comparison of CNN and RNN performance, all CNN models have more comparable accuracy and less comparable loss. CNN uses more time than RNN model and produces higher accuracy. The winner in our project is CNN model by applying two protein properties for training model.

9.2 Future work

Based on our results, the performance of RNN and CNN model on the test data can be improved by different methods. By reducing the learning rate, increasing the number of epochs, different loss function. Moreover, collecting more data will have great impact on learning in deep learning model.

REFERENCES

Aftabuddin, M., & Kundu, S. (2007). Hydrophobic, hydrophilic, and charged amino acid networks within protein. *Biophysical Journal*, *93*(1), 225–231. https://doi.org/10.1529/biophysj.106.098004

Dor, O., & Zhou, Y. (2006). Achieving 80% ten-fold cross-validated accuracy for secondary structure prediction by large-scale training. *Proteins: Structure, Function, and Bioinformatics*, *66*(4), 838–845. https://doi.org/10.1002/prot.21298

Fischer, D. (1999). Hybrid fold recognition: Combining sequence derived properties with evolutionary information. *Biocomputing 2000*. https://doi.org/10.1142/9789814447331_0012

nes, D. T. (1999). Protein secondary structure prediction based on position-specific scoring matrices 1 1edited by G. Von Heijne. *Journal of Molecular Biology*, *292*(2), 195–202. https://doi.org/10.1006/jmbi.1999.3091

Kabsch, W., & Sander, C. (1983). Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, *22*(12), 2577–2637. https://doi.org/10.1002/bip.360221211

Kumar, V., Ranjan, A., Cao, D., Krishnasamy, G., & Deepak, A. (2023). A sequence-motif based approach to protein function prediction via deep-CNN architecture. *Proceedings of the 15th International Conference on Agents and Artificial Intelligence*. https://doi.org/10.5220/0011647800003393

Lin, Z., Lanchantin, J., & Qi, Y. (2016). Must-CNN: A multilayer shift-and-stitch deep convolutional architecture for sequence-based protein structure prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, *30*(1). https://doi.org/10.1609/aaai.v30i1.10007

Pan, X., Rijnbeek, P., Yan, J., & Shen, H.-B. (2018). Prediction of RNA-protein sequence and structure binding preferences using deep convolutional and recurrent neural networks. *BMC Genomics*, *19*(1). https://doi.org/10.1186/s12864-018-4889-1

Al-Azzawi, A. (2017). Deep Learning Approach for secondary structure protein prediction based on first level features extraction using a latent CNN structure. *International Journal of Advanced Computer Science and Applications*, *8*(4). https://doi.org/10.14569/ijacsa.2017.080402

Cheung, M. S., Klimov, D., & Thirumalai, D. (2005). Molecular crowding enhances native state stability and refolding rates of globular proteins. *Proceedings of the National Academy of Sciences*, *102*(13), 4753–4758. https://doi.org/10.1073/pnas.0409630102

Nietz, V. (2015). A first order phase transition and self-organizing states in single-domain ferromagnet. *Physical Science International Journal*, *5*(4), 255–266. https://doi.org/10.9734/psij/2015/14982

Protein secondary structure prediction with hydrophobicity and hydrophobic moment. (2007). *Intelligent Engineering Systems Through Artificial Neural Networks, Volume 17*, 49–56. https://doi.org/10.1115/1.802655.paper8

Guo, Y., Wang, B., Li, W., & Yang, B. (2018). Protein secondary structure prediction improved by recurrent neural networks integrated with two-dimensional convolutional neural networks. *Journal of Bioinformatics and Computational Biology*, *16*(05), 1850021. https://doi.org/10.1142/s021972001850021x

Pearl, F. M. G., Lee, D., Bray, J. E., Buchan, D. W. A., Shepherd, A. J., & Orengo, C. A. (2009). The Cath extended protein-family database: Providing structural annotations for genome sequences. *Protein Science*, *11*(2), 233–244. https://doi.org/10.1110/ps.16802

Dawson, N. L., Sillitoe, I., Lees, J. G., Lam, S. D., & Orengo, C. A. (2017). Cath-Gene3D: Generation of the resource and its use in obtaining structural and functional annotations for protein sequences. *Protein Bioinformatics*, 79–110. https://doi.org/10.1007/978-1-4939-6783-4_4

Yang, Y., Heffernan, R., Paliwal, K., Lyons, J., Dehzangi, A., Sharma, A., Wang, J., Sattar, A., & Zhou, Y. (2016). Spider2: A package to predict secondary structure, accessible surface area, and main-chain torsional angles by deep neural networks. *Methods in Molecular Biology*, 55–63. https://doi.org/10.1007/978-1-4939-6406-2_6

Lee, B., & Richards, F. M. (1971). The interpretation of protein structures: Estimation of Static Accessibility. *Journal of Molecular Biology*, *55*(3). https://doi.org/10.1016/0022-2836(71)90324-x

Huang, Y., Weng, Y., Yu, S., & Chen, X. (2019). Diffusion convolutional recurrent neural network with rank influence learning for traffic forecasting. *2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. https://doi.org/10.1109/trustcom/bigdatase.2019.00096

Wang, G., & Dunbrack, R. L. (2003). Pisces: A protein sequence culling server. *Bioinformatics*, *19*(12), 1589–1591. https://doi.org/10.1093/bioinformatics/btg224

Yang, Y., Gao, J., Wang, J., Heffernan, R., Hanson, J., Paliwal, K., & Zhou, Y. (2016). Sixty-five years of the long march in protein secondary structure prediction: The final stretch? *Briefings in Bioinformatics*. https://doi.org/10.1093/bib/bbw129

Anfinsen, C. B. (1973). Principles that govern the folding of protein chains. *Science*, *181*(4096), 223–230. https://doi.org/10.1126/science.181.4096.223

APPENDICES

VITAE

Name of Author:	Sai Teja Kairamkonda
Address:	201 University Avenue, Troy, AL, 36081
Telephone Number:	(334) 492-2722

EDUCATION

Master of Science in Computer Science – Troy University, Troy, AL, 2023.

Major: Software Development

Bachelor of Technology in Information Technology – VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, India, 2016.

Major: Information Technology