Chapter 1
Introduction

1.1 Problem Summary

Predicting protein function is crucial for understanding how cells, tissues, and organs work. Proteins play an essential role in biological activities, and determining their functions can offer valuable insights into the inner workings of living organisms. However, discerning protein function experimentally may be a tedious and costly undertaking. With the rise in accessibility of protein sequence data, computational methods have become an important approach to protein function prediction. Even while current approaches, such as feature- and sequence-based machine learning algorithms, have showed promise, they are not as effective with distant or orphan proteins. The goal of this research is to create a model that, using the amino acid sequences of proteins, can reliably predict their function. Researchers would benefit greatly from such a model as it may illuminate biological functions and possibly contribute to the development of novel medications and treatments.

1.2 Aim and Objectives

Through the development of models trained on amino acid sequences and supplementary data, the initiative aims to enhance the prediction of protein function. The goal is to further our knowledge of protein activities, which is essential for deciphering the workings of cells and promoting developments in the fields of medicine and agriculture.

The goal is to use feature-based extraction to estimate the likelihood that a protein sequence will be connected to a given collection of GO keywords. The project aims to overcome difficulties in precisely allocating biological roles to proteins by utilizing data science and deep learning techniques. This would ultimately promote advancements in the treatment of diseases, the discovery of new drugs, and general health and wellbeing.

1.3 Problem Specifications

The biological role of genes and gene products is described at various degrees of abstraction by the Gene Ontology (GO), a concept hierarchy .It is a useful model for explaining how protein activity is multifaceted.

GO takes the form of Directed Acyclic Graph. Functional descriptors, such as words or classes, are the nodes in this network, which are linked together by relational links such as is_a, part_of, etc. Based on the root nodes of each subgraph, this graph may be divided into three subgraphs, or subontologies: Biological Process, Molecular Function, and Cellular Component. In terms of biology, each subgraph represents a distinct facet of the protein's role: its molecular functions (MF), the biological processes in which it takes part (BP), and its location inside the cell (CC). This means that a subset of one or more of the subontologies represents the function of the protein. For each protein, the model's training data will be labeled with term-protein assignments that have been found through experimentation. The likelihood that a protein sequence would be connected to a certain collection of GO keywords should be precisely predicted by the model.

1.4 Background

1.4.1 What are Proteins?

The intricacies of life can be broken down into a countable number of smaller components. Proteins are among these biological building blocks, huge macromolecules that are responsible for thousands of diverse biological and chemical functions necessary for life's creation and maintenance. More precisely, proteins govern the vast majority of the labor that occurs within an organism's cells, resulting in the formation, function, and maintenance of tissues and organs in the body. Proteins are large biological molecules with complex dynamics, and understanding how they operate and what functions they may perform may be extremely difficult without first considering the constituent components from which they are created.

However, proteins are composed of several lengthy chains of small organic substances known as amino acids. Every amino acid contains an amine group, a carboxyl group, and what is known as a side chain defining each amino acid's attributes. There are twenty potential side chains, and as a result, there are twenty distinct naturally occurring amino acids . As a result of these characteristics, long chains of amino acids interact in complex ways to form interesting and often novel structures within the proteins which ultimately determine their functions .
Strategically analyzing the sequence of amino acids in a protein can provide insights into its structure, function, and interactions with other proteins. This approach is becoming increasingly prominent in computational biology for investigating the functions of various proteins in a cost-effective manner.

1.4.2 Gene Ontology

We humans have over 20,000 distinct proteins that help run the activities inside our cells. Identifying what each of these proteins performs is a collosal task. First, we need a means to categorize them. This is where the concept of an ontology enters . It's like developing a map of what we know about protein activities and how they interact with one another.

Francis Crick proposed the Central Dogma of Molecular Biology. It all reduces to this: DNA produces RNA, which then produces proteins via transcription and translation processes. Essentially, our DNA encodes a unique set of instructions for each protein. This theory eventually leads to the conclusion that each protein is encoded by a distinct gene in our DNA. [1]

Michael Ashburnerl recognized the importance of an ontology and developed the notion of Gene Ontology (GO) [3]. Gene ontology seeks to standardize the gene representation and gene product characteristics across all species in order to provide and maintain annotations of gene products—and thus protein functions—as well as to facilitate the simple functional interpretation of experimental data.

.

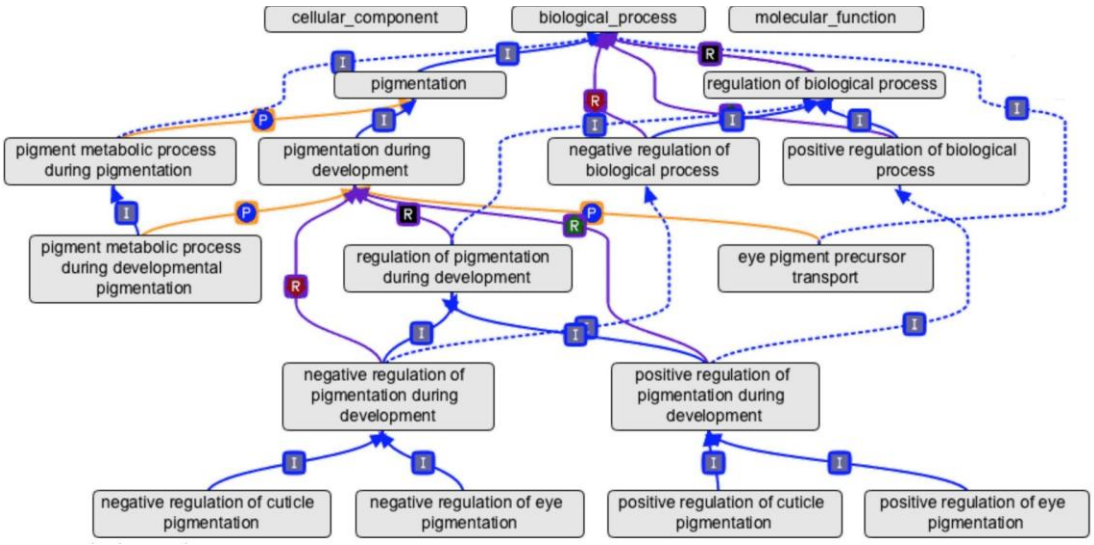


Figure 1 A Gene Ontology DAG graph displaying a few of the tree's more advanced nodes [http://www.geneontology.org/page/ontology-structure]

Like a family tree, but for proteins, is how the Gene Ontology is organized. The cellular component, molecular function, and biological process are its three primary branches. These branches are akin to broad groups that encompass a wide variety of proteins. Things become more precise as you descend the tree. Every branch encompasses a more limited subset of proteins, with each protein being characterized by certain features. These characteristics consist of the protein's name, unique ID, function, and place in the larger biological context. It can be compared to an intricate map of the world of proteins.

For instance, a typical GO node looks

```
id: GO:0000016
name: lactase activity
namespace: molecular_function
def: "Catalysis of the reaction: lactose + H2O=D-glucose
+ D-galactose."
synonym: "lactase-phlorizin hydrolase activity" BROAD
[EC:3.2.1.108]
synonym: "lactose galactohydrolase activity" EXACT [EC:3.2.1.108]
xref: EC:3.2.1.108
xref: MetaCyc:LACTASE-RXN
xref: Reactome:20536
is_a: GO:0004553 ! hydrolase activity, hydrolyzing O-glycosyl
compounds
```

1.4.3 The Critical Assessment of Protein Function Annotation Algorithms (CAFA)

Protein function annotation algorithms are developing quickly, so it's critical to have a method for testing them in an impartial environment to see how well these tools perform in real-world scenarios. A timed task called the Critical Assessment of Protein Function Annotation Algorithms (CAFA) is intended to gauge the comprehensive evaluation of computer techniques devoted to protein function prediction [3]. Heres, how CAFA operates:

- A significant proportion of protein sequences that have not yet undergone experimental annotation are made public by CAFA organizers.
- Rivals use their models to forecast these proteins' functions by mapping them to GO keywords or Human Phenotype Ontology.
- A few months after a prediction deadline, some proteins whose roles were unclear through experimentation during the competition have been confirmed through experimentation

- These proteins serve as a standard for testing and comparing the various approaches.

1.5 Literature Review

One of the most significant and active areas of bioinformatics study is protein function prediction. Protein sequences have been steadily rising in databases like Swiss-Prot [4] and PDB [5] because of the development of efficient sequencing techniques. One of the many rival sequencing businesses, Illumina, manufactures a range of sequencers that can sequence the human genome up to 30 times in less than 30 hours [6]. The graphic in Figure 2 [7] shows how this exponential rise has resulted in a disparity between the number of sequences available and sequences annotated, with more than 80 percent of sequences in UniProtKB version 2015 01 missing assignments for at least one of the three GO sub-ontologies.

Conventional experimental techniques such as targeted mutations, gene knockdown, and expression inhibition are labor-intensive and time-consuming. These techniques require a significant investment of time and labor to analyze a single protein. This strategy is no longer practical on a broad scale due to the constantly expanding collection of protein sequences that are available. For this reason, developing computational methods for assessing and forecasting protein functions is crucial, if not ideal. By bridging the gap between the shortcomings of previous methods, these solutions assist reduce the effort required to effectively annotate protein sequences and make the process more manageable.

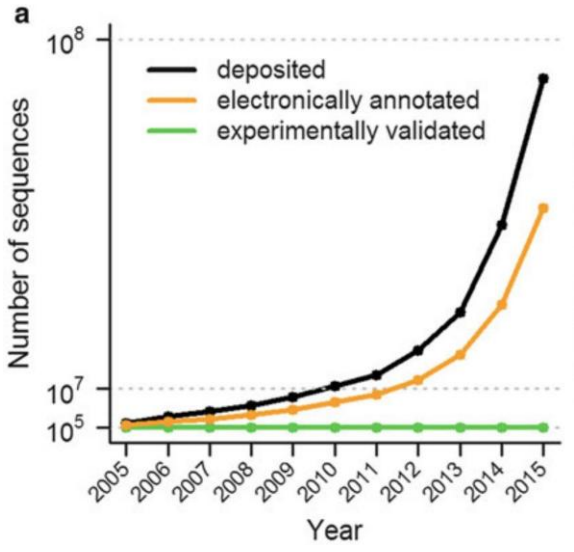


Figure 2 This graphic illustrates how the number of amino acid sequences skyrocketed exponentially (black) while the number of experimentally validated supported GO term assignments grew steadily and linearly (green). The increasing trend in the number is depicted by the orange line.

There are three primary classes of sequence-based function prediction techniques:

- Homology-based techniques are predicated on the idea that proteins with comparable sequences frequently perform comparable tasks. By comparing a target protein to others in sizable databases, these techniques use similarities to deduce the target protein's function.
- Sequence motif-based techniques involve Identifying particular domains within a query sequence and offering support for possible functionalities based on these motifs
- Feature-based methods involve Protein sequences with certain motifs or other attributes, such as length, the makeup of amino acids, or secondary structure, are used in feature-based approaches. Using these characteristics, a function approximator that can infer protein functions is constructed.

Sequence-homology based techniques such as FASTA [9] and BLAST [10] were among the first effective attempts to predict protein function. These systems compare protein sequences in databases using approximation sequence alignment techniques in an attempt to find homologous proteins with sequence structures comparable to the one under study. The underlying notion is that proteins with almost identical sequences evolved from a single common ancestor and may hence play similar roles [11]. However, these procedures are not flawless, as a protein's copy may perform a distinct function as a consequence of divergent evolution [12]. It has been suggested that sequence-based approaches should use protein structure to overcome this difficulty. This is owing to the fact that a protein's folding pattern frequently persists even in cases where evolutionary divergence renders sequence similarity undetectable [13].

Subsequences within a protein can often be powerful markers of the precise tasks that the protein performs. These subsequences, known as motifs, are potentially functional locations inside proteins because they are conserved among proteins in the same family. Pfam, a database of protein families that comprises a sizable collection of families represented by numerous sequence alignments and hidden

Markov models, is a useful tool for locating domains within a given sequence [14]. A sequence's motifs can contain DNA binding sites, ligand binding sites, and locations that promote interactions with other proteins, all of which provide important hints regarding the function of the protein [15, 16].

Hannenhalli and Russell were the first to apply these techniques, especially on protein sets with few near homologs. They found that their accuracy was 20% higher than that of sequence-based similarity techniques like as BLAST, with an accuracy of 96% as opposed to 74% for BLAST [17].

Using motif-based approaches, which examine the raw sequence of amino acids to identify relevant areas, potential protein functions can be inferred. Alternatively, this amino acid sequence can first be changed into biologically significant properties, thereby developing a functional presentation of the protein that might be more effective for categorizing functions. These techniques, referred to as feature-based methods, usually predict a protein's function based on its sequence by using machine learning algorithms like SVMs and neural networks. Similar to motif-based approaches, feature-based approaches perform well in situations where there are no homologues for the protein sequence.

Jensen et al.'s neural network-based function predictor, which focuses on cellular roles and enzymatic functions classified by the EC system, was one of the groundbreaking achievements [18]. They used sequence-derived characteristics such as protein sorting signals, predicted post-translational modifications, and various physical and chemical properties calculated from amino acid composition. With a neural network that included one hidden layer with 10–50 neurons that were suited to several functional categories, they were able to accurately estimate the roles of orphan proteins. They then used an analogous process to create a model that included a number of physiologically and pharmaceutically significant GO categorization system categories. With a 70% recall rate for hormone and receptor function classes, this model can be used for assay selection and gene identification.[19]

FFPred is a feature-based function predictor for vertebrate proteomes that uses a variety of Support Vector Machines (SVMs) [20]. FFPred trained SVMs for predicting activities related with over 300 GO annotations by utilizing a variety of protein attributes, such as amino acid composition, structural disorder, secondary structure, and other biologically significant factors. Five SVMs with a Radial Basis Function kernel were used to address each GO term. This method enabled accurate function prediction for a wide variety of GO terms in the MF and BP subontologies, including remote or orphan proteins. Following developments, FFPred 2.0 was developed, which included bigger protein sequence and annotation datasets, updated feature predictors, and modified training processes to improve prediction accuracy [21]. FFPred 2.0 maintained its SVM-based technique and predicted 442 GO annotations inside the MF and BP sub-ontologies, winning them the first CAFA competition

Continuing their search for reliable protein function prediction methods, Domenico Cozzetto et al. improved the FFPred model to include the cellular component GO sub- ontology, yielding FFPred 3 [22]. FFPred 3, which uses an extended SVM library, emerged as the cutting-edge model for predicting the activities of distant or orphan proteins, with 597 GO annotations. Compared to BLAST and naïve approaches across all three GO sub- ontologies, FFPred 3 predictions displayed improved precision values for high recall rates, and they beat rivals in term-centric assessments during the CAFA2 experiment [23].

As machine learning increasingly incorporates deep neural approaches, important successes have arisen in a variety of fields, including facial recognition [24], language translation [25], and reinforcement learning using learning agents [26]. Consequently, interest in applying these models to bioinformatics has increased. Recent studies have used deep convolutional neural networks and long short-term memory recurrent neural networks to predict protein secondary structure [27, 28], protein disorder [29], protein subcellular localization [30], and protein contact [31]. These initiatives constantly produce major advances, frequently setting new standards in their respective professions.

Similarly, deep learning has made considerable breakthroughs in predicting protein functions. Kulmanov et al. extracted characteristics from protein sequences using a convolutional neural network with a representation embedding layer based on trigrams. They integrated this with data on protein structure & protein-protein interaction networks, using a hierarchy of deep fully-connected layers to fine-tune characteristics for each specific component of the GO categorization. This architecture outperforms baseline techniques like BLAST, particularly for predicting cellular positions [32]. Fa et al. examined the usefulness of MTDNN, in which every GO term share a hidden-layer representations that branches into concurrent stacks of distinct hidden layers for each GO term being categorized. Interestingly, our approach outperforms baseline techniques for proteins lacking near homologs by using similarities and differences between prediction tasks [33].

However, because to the vast number of factors involved, these approaches often require extensive datasets, making training computationally expensive and time-consuming, frequently taking weeks. Compounding this difficulty is the necessity for specialized datasets engaged in protein function prediction, which necessitates specific changes to reduce the similarity of sequence between the training and test sets, resulting in smaller data sizes, as mentioned in the FFPred work.

1.6 Plan of the work

Phase	Description	Timeline (Weeks)
1. Project Initiation		4
1.1	Literature Review	2
1.2	Plan Development & Research	1
1.3	Board Approval	1
2. System Design & Development		6
2.1	Decide Architecture	1
2.2	Data Preprocessing & Model Development	1
2.3	Model Evaluation & Refinement	2
2.4	System Integration & Deployment Planning	2
3. System Implementation & Testing		6
3.1	Start Implementing System	3
3.2	Testing & Deployment	2
3.3	Documentation	1

Table 1 Plan of Work

1.7 Material/Tools Required

Software requirements

- Development environment: Jupyter notebook environment
- Deep Learning Framework: TensorFlow or PyTorch
- Sequence Analysis: Biopython
- Numerical Computing: NumPy
- Visualization: Matplotlib, Seaborn
- Git for tracking code changes and collaborating with team members.

Hardware requirements

- A stable internet connection and networking infrastructure
- CPU : Multi-core processor with good clock speed
- GPU: (Nvidia Tesla P100 , 16 GB)
- RAM: At least 16GB, 32GB+ preferred

Chapter 2 Observation, Methodology, and System Design

2.1 Observations

- **Data Outpaces Traditional Methods:**

  The exponential growth of protein sequence data (e.g., Swiss-Prot, PDB [4, 5]) surpasses the capabilities of traditional, time-consuming experimental function prediction methods (e.g., gene knockout) [8]. This necessitates the development of robust computational techniques to bridge this gap [7].

- **Limitations of Homology-Based Methods:**

  While initially successful [9, 10], homology-based methods suffer from inaccuracies due to divergent protein evolution [11, 12]. Incorporating protein structure can help address this issue, as structure can be conserved even with low sequence similarity [13].

- **Sequence Motifs Offer Functional Clues**:

  Sequence motif-based methods analyze the protein sequence for functional regions, providing clues to protein function [14, 15, 16]. These methods outperform homology-based methods for proteins lacking close homologs [17].

- **Machine Learning Success with Feature-Based Methods**:

  Machine learning approaches, particularly feature-based methods utilizing SVMs and neural networks, have demonstrated success in protein function prediction [18, 19, 20]. These methods are especially valuable when sequence homology is unavailable [18, 19]. FFPred is a noteworthy example of a successful SVM-based function predictor [20, 21, 22, 23].

- **Deep Learning Shows Great Promise**:

  Deep learning methods employing CNNs and RNNs have lately emerged as powerful tools in protein function prediction [27, 28, 29, 30, 31, 32, 33]. These methods achieve SOTA results in various protein function-related tasks, including protein-protein interactions and cellular localization [32, 33].

Observation	Description	Reference
Data Outpaces	Exponential growth of protein sequence data	[4, 5, 8]
Traditional Methods	outpaces traditional experimental function prediction methods	
Limitations of Homology-Based Methods	Homology-based methods can be inaccurate due to protein evolution; incorporating protein structure can improve accuracy	[9, 10, 11, 12, 13]
Sequence Motifs Offer Functional Clues	Sequence motif-based methods analyze protein sequences for functional regions, providing clues to protein function	[14, 15, 16, 17]
Machine Learning Success with Feature-Based Methods	Machine learning approaches, particularly feature-based methods using SVMs and neural networks, are successful in protein function prediction, especially when sequence homology is unavailable (e.g., FFPred)	[18, 19, 20, 21, 22, 23]
Deep Learning Shows Great Promise	Deep learning methods using CNNs and RNNs show great promise in protein function prediction, achieving SOTA results in protein-protein interactions as well as cellular localization	[27, 28, 29, 30, 31, 32, 33]

Table 2 Literature Review Summarised

2.2 Idea

The proposed idea is to train a Deep Neural Network for multi-label prediction of GO Terms based on sequential biophysical features extracted from protein sequences, the goal for the network is to decipher functional representations that depend on the sequence.. Most appropriate choice for Deep Learning Architecture

would be either LSTM or 1D-CNN, given their ability to extract high-level feature-representations and dependencies in Sequential Data.

2.3 Methodology

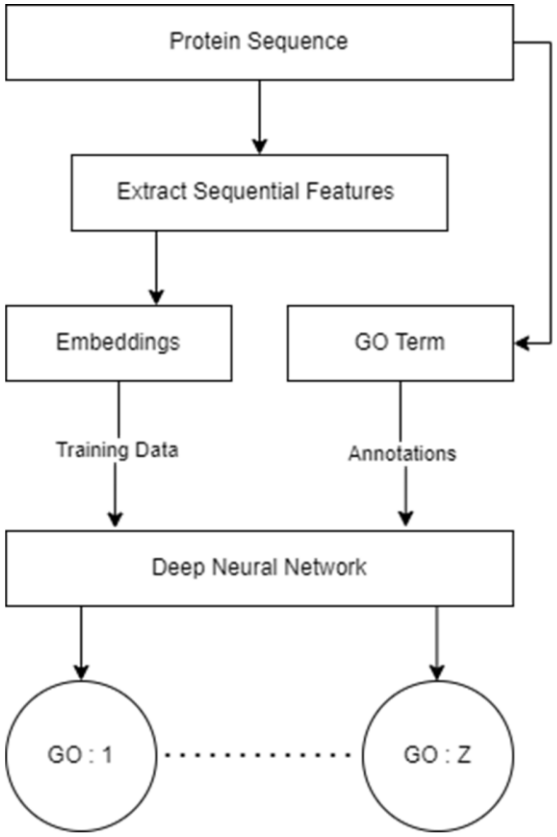


Figure 3 Pipeline for the model

This process outlines the steps involved in using a deep neural network to predict the protein function via its amino acid sequence:

1. Protein Sequence Input:

The process starts with the protein sequence as input. The sequence is composed of series of amino acids that constitute the protein The order and specific types of amino acids determine a protein's structure and ultimately its function.

2. Feature Extraction/Embedding:

In this step, the protein sequence is transformed into a series of numbers that the neural network can use. Every amino acid or brief chain of an amino acid has a number value stored into a vector. These vectors, which go by the name of embeddings, encapsulate the crucial aspects of the biological protein sequence that are important for predicting function.

3. Deep Neural Network Processing:

Deep learning is utilized to create neural networks that are provided with regular protein sequences that have been encoded, or embeddings. In turn, the network is composed of numerous layers of connected neurons, which are learning units capable of understanding complex connections between protein function and protein sequence properties. Moreover, as data moves through the network, neurons conduct computations and transformations, with the output ultimately being linked to protein function.

4. GO Term Prediction:

In general, the neural network output is converted to terms in the Gene Ontology . With the assembled sequence knowledge, the network can forecast the probability that a protein will fall into a given GO category.

5. Interpretation and Analysis:

The GO terms this protein will likely be associated with are predicted to be the role of the protein within the cell. This, combined with other protein data, gives insights into the processes in biology and relationships between proteins.

2.4 Dataset Details

GO is a hierarchical structure that describes protein functions at various levels. Proteins can be annotated with GO terms from three main subontologies: MF, BP, and CC. This dataset uses experimentally validated GO annotations as ground truth labels for prediction of protein function.

- **Training set**:

Contains proteins with experimentally validated GO annotations from UniProtKB. Most protein sequences are taken from the Swiss-Prot database, but a subset of proteins that are not represented in Swiss-Prot were extracted from the TrEMBL database. In both cases, the sequences come from the 2022_05 release from 14-Dec-2022. Protein sequences are provided in FASTA format. Corresponding GO terms and subontology information are provided in a separate file.

- **Test sets:**
  - **Test superset**: Contains protein sequences for prediction, but their GO annotations are unknown.
  - **Test set**: A subset of the test superset that accumulates experimental annotations. Used for final evaluation.

  The GO ontology structure is provided in OBO format. Taxonomy information for proteins is also available.

- Information accretion (IA) weights for GO terms are included for weighted evaluation metrics.

2.5 System Design

As mentioned in the problem specification, the aim is to predict the probability of protein sequence belonging to a set of GO terms, through feature-based extraction which can done using 1D-CNN or LSTM

2.5.1 1D-CNN (One Dimensional Convolutional Neural Network)

1D-CNN is a deep learning architecture that is tailored for effectively handling sequential data.

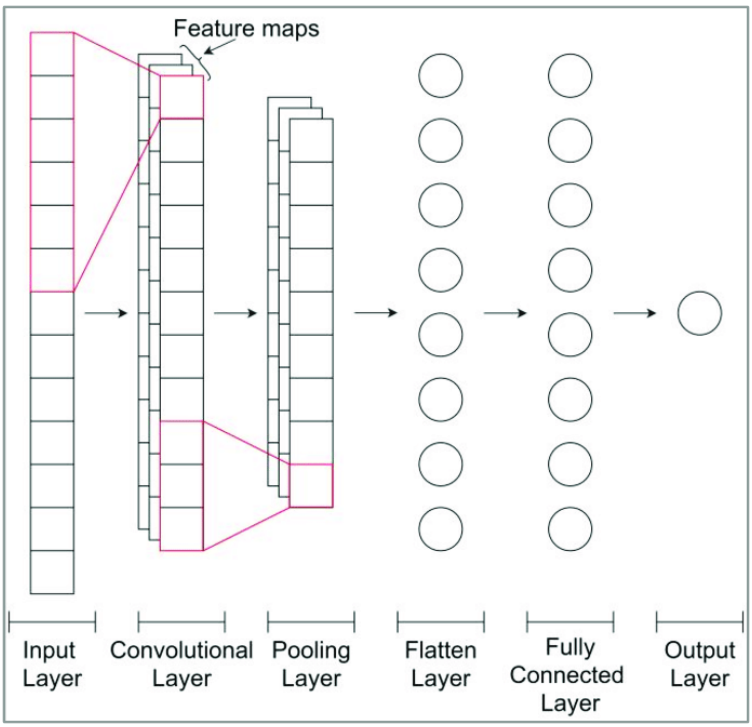


Figure 4 1D-CNN Architecture [https://www.researchgate.net/profile/Elsa-Chaerun-Nisa/publication/348502722/figure/fig2/AS:982335168212994@1611218363330/One-dimensional-convolutional-neural-network-1D-CNN-architecture-for-the-timeseries.png]

1D-CNN architecture consists of the following key components:

- **Input Representation**
  - The input to a 1D-CNN for protein function prediction is typically the protein's amino acid sequence.
  - The sequence is represented as a one-dimensional array or vector, where each element corresponds to a specific amino acid.
  - To represent the amino acids, a standard practice is to use a one-hot encoding scheme, in which every amino acid is encoded in a binary vector of length 20 (the number of standard amino acids).
  - For instance, the amino acid 'Alanine' is represented as [1, 0, 0, 0, ..., 0], 'Cysteine' as [0, 1, 0, 0, ..., 0], and so on.
  - Alternatively, the amino acids can be represented using numerical encodings, such as biochemical properties or learned embeddings.
- **1D-Convolutional Layers**

- The convolutional layers in a 1D-CNN are designed to extract local features and patterns from the input protein sequence.
- Each convolutional layer comprises a set of learnable filters (called kernels or feature detectors) that move along the input sequence, performing convolution operations.
- The filters are typically small in size, such as 3-5 amino acids, and are trained to detect specific motifs or patterns within the protein sequence.
- The convolution process entails multiplying each element of the input sequence by the corresponding element of the filter, and then adding these products to produce a single output value.
- This process is repeated for multiple filters, to create a feature map that captures different local patterns in the input sequence.

- **Pooling Layers**
  - After the convolutional layers, pooling layers are often used to reduce feature map's Spatial Dimensions and extract the significant features.
  - Max-Pooling and Average Pooling are standard pooling operations
  - Max-pooling selects the maximum value within a specific window, while average-pooling calculated the average value.
  - Pooling layers help to make the model more resilient to small shifts and distortions in the input sequence, and they also reduce the computational complexity of the network.
- **Fully Connected Layers**
  - The features extracted by the previous layers are then fed to one or more fully connected layers, which act as a classifier.
  - The fully connected layers learn complicated non-linear relationships between the extracted features and the target protein functions.
  - These layers typically consist of dense matrix multiplications, followed by a non-linear activation function, Rectified Linear Unit.
  - It calculates the expected probability or scores that indicate the protein's function or class.
- **Output Layer**

- The output of the 1D-CNN model is a vector of probabilities or scores, where each element corresponds to the likelihood of the input protein belonging to a specific GO term.

- **Model Training and Optimization:**
  - A properly labeled dataset containing protein sequences and their corresponding GO Terms is employed to train the one CNN model, for predicting protein functions.
  - Techniques such as descent and backpropagation are utilized to tune the model parameters, including weights of fully connected layers and convolutional filters.
  - During the training process the objective function typically involves a loss function that measures the difference between labels and predicted outputs, such, as entropy loss.
  - The regularization technique called dropout is implemented to improve the algorithms generalization performance by preventing overfitting.
  - The accuracy and performance of the trained model are then evaluated using a test set.

2.5.2 LSTM (Long Short-Term Memory)

LSTM is a variant of RNN architecture , designed by Sepp Hochreiter & Jürgen Schmidhuber in 1997 to handle long-term dependencies.

The key components of an LSTM unit include:

- **Forget Gate**: It controls whether information from the prior hidden state and present input should be forgotten or not.
- **Input Gate**: It determines whether information from the current input and prior concealed state should be added to the cell state.
- **Output Gate**: It determines the new hidden state depending on the current input, previous hidden state, and current cell state.
- **Cell State**: Serves as a memory bank that selectively keeps or discards information, allowing the LSTM to detect long-term relationships within the

input sequence.

Apart from architectural differences, remaining steps remain the same for training the model for protein function prediction.

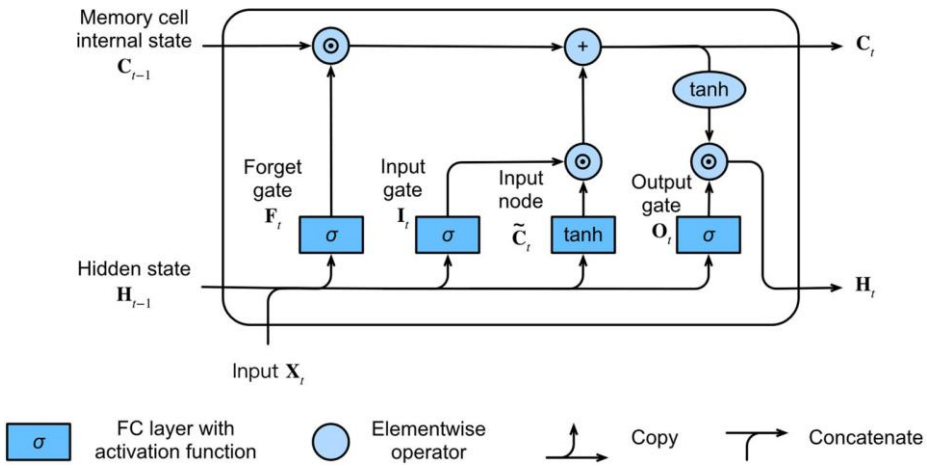


Figure 5 LSTM Memory Cell (Credits: https://d2l.ai/chapter_recurrent-modern/l

Chapter 3
Implementation and Results

3.1 Implemented Functionality

3.1.1 Working with Protein Sequences

Proteins are built from a set of 20 different amino acids, each of which has a unique side chain. The side chains of amino acids have different chemical structures, which give proteins their unique properties.

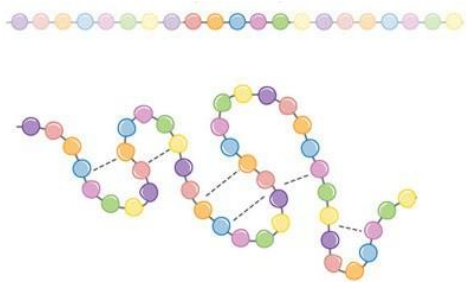


Figure 6 Protein Structure representation

Example of a Protein Sequence taken from train dataset (id = P01112):

MTEYKLVVVGAGGVGKSALTIQLIQNHFVDEYDPTIEDSYRKQVVIDGETC
LLDILDTAGQEEYSAMRDQYRTGEGFLCVFAINNTKSFEDIHQYREQIKRV
KDSDDVPMVLVGNKCDLAARTVESRQAQDLARSYGIPYIETSARQGVEDAF
YTLVREIR QHKLRKLNPPDESGPGCMSCKCVLS

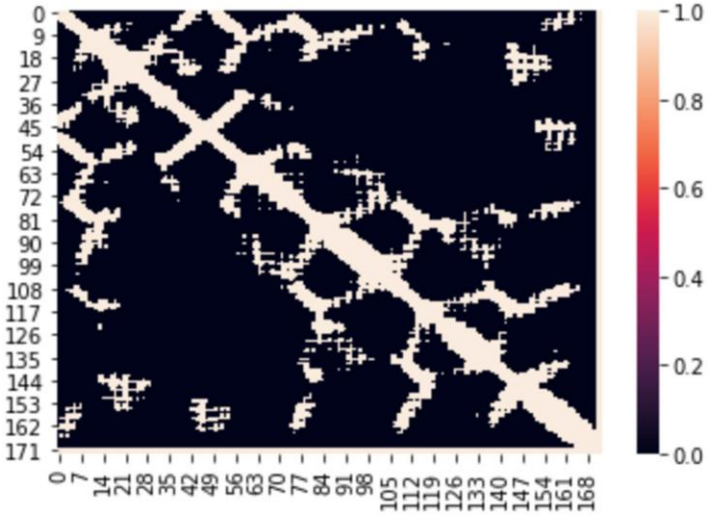


Figure 7 Chain A with 12 angstroms as distance threshold

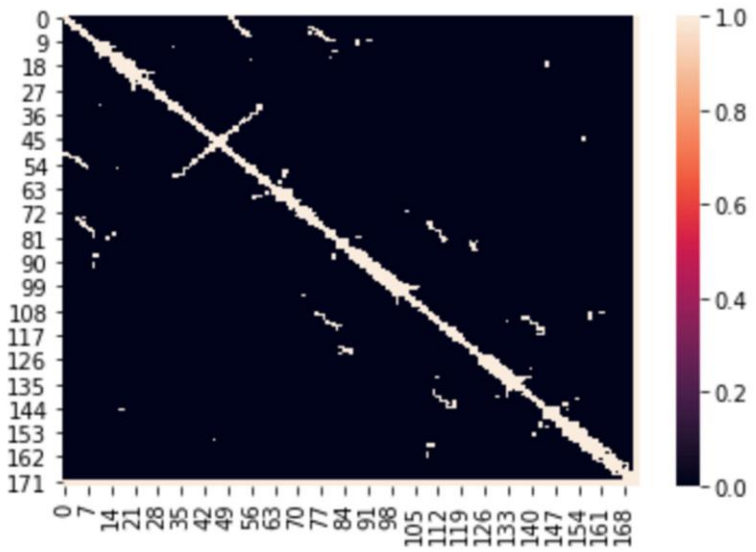


Figure 8 Chain A with 6 angstroms as distance threshold.

The above images show two heat maps, one with a distance threshold of 12 Armstrong and one with a distance threshold of 6 Armstrong. The heat maps show the distances between the amino acids in the protein chain. The red areas of the heat maps show the areas where the amino acids are close together, and the black areas of the heat maps show the areas where the amino acids are far apart

3.1.2 Exploratory Data Analysis

In EDA we delved into a dataset extracted from the 'train_term' file, which presents a total of 5,363,863 rows and 3 columns. Each row corresponds to a specific protein function instance, and the columns contain essential information regarding these functions. Specifically, the columns represent:

Protein Function ID
Protein Function Description
Subontology Root (BPO, CCO, MFO)

Our study mainly concentrated on comprehending the features and structure of the dataset, especially in relation to the prediction of protein function. The

Gene Ontology (GO) words, which classify proteins according to their functionalities as well as cellular locations, were also used to examine the underlying biological linkages.

Exploration Highlights:

To better comprehend the diversity and prevalence of various function types, we looked at the distribution of protein functions among the three subontology roots (BPO, CCO, and MFO).

Statistical Analysis:

Carried out statistical analysis on the dataset to find trends, patterns, and outliers. Assessing the pattern of distribution of protein activities and their properties was made easier by this analysis.

GO Term Analysis:

Investigated GO terms linked to the functionalities of proteins in order to understand their biological relevance. Important details like title, the namespace, the definition, associations, etc. are provided by each GO term.

Network Visualization of GO Terms:

The GO term "GO: 0052892," which represents a particular example, was used to display it utilizing a network diagram. The GO term's title, the namespace, the definition, relationships, along with other pertinent information are all summarized in this visualization. The network diagram offers a thorough summary of the

relationships and biological context of the protein's function that the GO term represents.

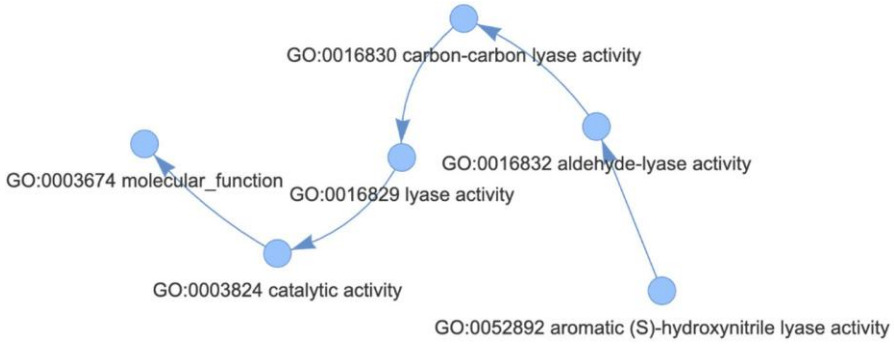


Figure 9 Geno Ontology Graph for GO:0052892

3.1.3 Data Preprocessing

Step 1: **Filter out proteins from a large amount of data**

In this stage, all of the proteins in the dataset that do not satisfy certain requirements are filtered out. The following requirements are met:

- The protein sequences can only have a maximum of 500 amino acids.

By using these standards, we can make sure that the data used for training is of the highest caliber as well as whether the model will have the ability to able to draw useful conclusions from it.

Step 2: Generating a list of candidates GO terms

The GO terms that are three or fewer distances away from each GO term which has been identified in a minimum of 100 entries in the training set are the ones that are chosen to create the collection of applicants terms in the example. This guarantees that a collection of GO terms pertinent for the protein in the data set will be employed for training the model.

```
'GO:0051641': ['Q9CQV8', 'P62259', 'P61982', 'O70456', 'P68254'],
'GO:0097708': ['Q9CQV8', 'P62259', 'P63101'],
'GO:0030234': ['Q9CQV8', 'Q6PD03', 'Q6PD28', 'Q61151'],
'GO:0048523': ['Q9CQV8', 'P68510', 'Q6PD03', 'Q6PD28'],
'GO:0019538': ['Q9CQV8', 'P62259'],
'GO:0045184': ['Q9CQV8', 'P62259', 'P61982', 'O70456', 'P68254', 'P63101'],
'GO:0010563': ['Q9CQV8', 'P62259'],
'GO:0010646': ['Q9CQV8', 'P68510', 'P61982'],
'GO:0010467': ['Q9CQV8', 'P68510', 'P68254'],
'GO:0051649': ['Q9CQV8', 'P62259', 'P61982', 'O70456', 'P68254'],
'GO:0051246': ['Q9CQV8', 'P62259', 'Q6PD03'],
'GO:0140678': ['Q9CQV8'],
'GO:0048583': ['Q9CQV8', 'P63101'],
'GO:0006796': ['Q9CQV8', 'P62259'],
'GO:0010558': ['Q9CQV8', 'P68254'],
'GO:0060255': ['Q9CQV8', 'P68510', 'P68254', 'Q6PD28'],
```

Figure 10 Candidates Proteins for Specific GO Term

Step 3: Annotation of GO Terms

To assign importance to terms for model building, this entails finding pertinent GO terms, sorting them according to the information they provide, and assigning scores to each term.

a. **List frequently used GO words**

   We start by identifying GO terms that are present in the training as well as the test groups of proteins. It guarantees that the phrases selected are pertinent to the proteins under study.

b. **Use the content of information as a filter.**

   The content of the data of the discovered terms is used to filter them. The degree of informativeness a term possesses in respect to its wider category is determined by its information content. Retained terms are those that have data whose content is greater than a predetermined level.

c. **Determine the score and protein count**

   The total of the number of proteins present in the test set connected to each remaining term is tallied. In addition, a score is determined by multiplying the term's information content by the number of proteins. This score takes into account the term's particular correlation with

protein content in the set that was tested as well as its overall significance.

d. Prioritize terms

The terms are arranged in order of score. When creating a protein's function predictions model, phrases that are both instructive and pertinent to the particular proteins under study are given a higher priority according to this ranking.

3.1.4 Model training for a single GO period

Using Gene Ontology (GO) terminology, the following procedures are followed in order to train the model

Step 1: Choosing a GO term

Choose the GO keyword that scored the highest in the analysis. Given its high informational content and protein count, this phrase is probably the most relevant and common in the collection.

	go_term	ia	num_proteins	score
28010	GO:0000407	9.175414	11921	109380.109699
28298	GO:0005793	8.166037	12609	102965.554769
28792	GO:0020023	5.231151	16184	84660.945708
28438	GO:0008180	5.615446	14814	83187.212975
28297	GO:0005791	5.236855	14798	77494.973859
...	...	...	...	...
12458	GO:0045893	0.000343	2097	0.718521
13509	GO:0048364	0.002516	229	0.576072
2192	GO:0006355	0.000499	922	0.460238
2601	GO:0006874	0.003343	113	0.377809
34441	GO:0015267	0.000844	219	0.184820

824 rows × 4 columns

Figure 11 GO Term Analysis: IA, Proteins, and Scoring

Step 2: Preparing the training data for that GO term

- **Filter proteins:** Exclude proteins that already have the chosen GO term assigned. This ensures the model learns from proteins

that lack the term and need prediction.

- **One-hot Encoding:** Encode the amino acid sequences of both positive (with the term) and negative example proteins into a one-hot vector format suitable for the machine learning model. Ensure all sequences have the same length by padding shorter ones with empty vectors.

Step 3: Feeding the data in the model

- The prepared data is now ready to be used for Model training.
- Architecture of 1D-CNN model:

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d (Conv1D)              (None, 500, 64)           19264

dropout (Dropout)            (None, 500, 64)           0

max_pooling1d (MaxPooling1D  (None, 166, 64)           0
)

conv1d_1 (Conv1D)            (None, 166, 128)          90240

dropout_1 (Dropout)          (None, 166, 128)          0

max_pooling1d_1 (MaxPooling  (None, 55, 128)           0
1D)

conv1d_2 (Conv1D)            (None, 55, 256)           229632

dropout_2 (Dropout)          (None, 55, 256)           0

max_pooling1d_2 (MaxPooling  (None, 18, 256)           0
1D)

conv1d_3 (Conv1D)            (None, 18, 512)           655872

dropout_3 (Dropout)          (None, 18, 512)           0

max_pooling1d_3 (MaxPooling  (None, 6, 512)            0
1D)

global_average_pooling1d (G  (None, 512)               0
lobalAveragePooling1D)

dense (Dense)                (None, 512)               262656

dropout_4 (Dropout)          (None, 512)               0

batch_normalization (BatchN  (None, 512)               2048
ormalization)

dense_1 (Dense)              (None, 256)               131328

dropout_5 (Dropout)          (None, 256)               0

batch_normalization_1 (Batc  (None, 256)               1024
hNormalization)

dense_2 (Dense)              (None, 2)                 514

=================================================================
Total params: 1,392,578
Trainable params: 1,391,042
Non-trainable params: 1,536
_________________________________________________________________
```

Figure 12 Architecture of 1D-CNN model

- Architecture of LSTM model:

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 64)                21760

 dense (Dense)               (None, 128)               8320

 dense_1 (Dense)             (None, 2)                 258

=================================================================
Total params: 30,338
Trainable params: 30,338
Non-trainable params: 0
_________________________________________________________________
```

Figure 13 Architecture of LSTM Model

3.2 Results and Reports

3.2.1 EDA Results

Statistical Analysis

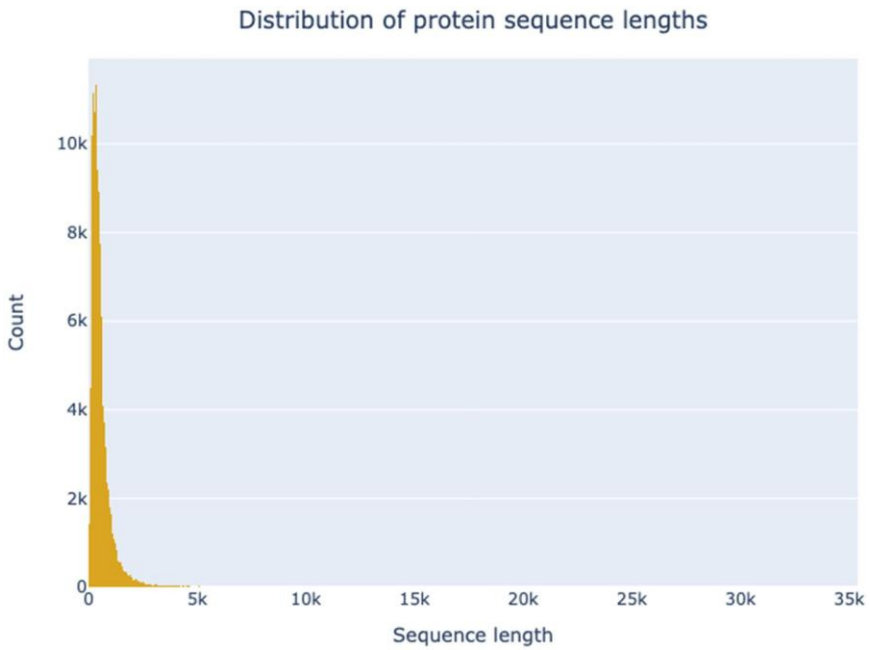


Figure 14 Distribution of protein sequence lengths. The x-axis represents protein sequence length in amino acids. The y-axis represents the number of protein sequences.

Figure 14 explores a protein function prediction dataset derived from the train_term file. The data comprises 5,363,863 protein-GO term associations represented in a three-dimensional matrix. Each dimension corresponds to a Gene Ontology (GO) category: Biological Process (BPO), Cellular Component (CCO), and Molecular Function (MFO). Network analysis techniques will be employed to unveil key functional relationships between Gene Ontology terms.

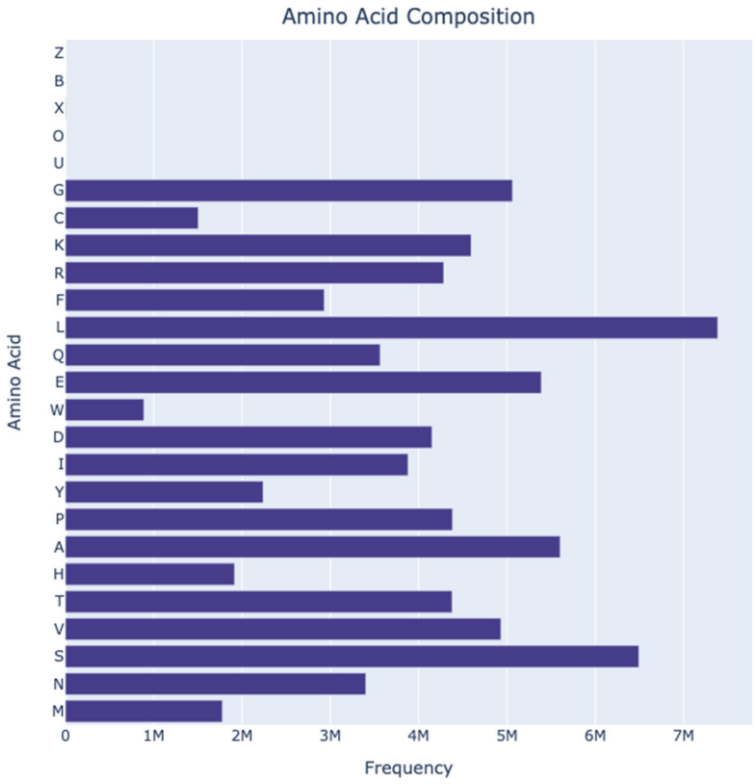


Figure 15 Amino Acid Composition

Figure 15 shows the distribution of the amino acids found in proteins. Every amino acid is represented along the x-axis using single-letter abbreviations, while the y-axis likely represents their absolute frequencies within the protein sequence. Taller bars indicate higher frequencies of occurrence for the corresponding amino acids. This visualization allows for comparative analysis of amino acid compositions across proteins, offering insights into their functional characteristics

Exploration Highlights

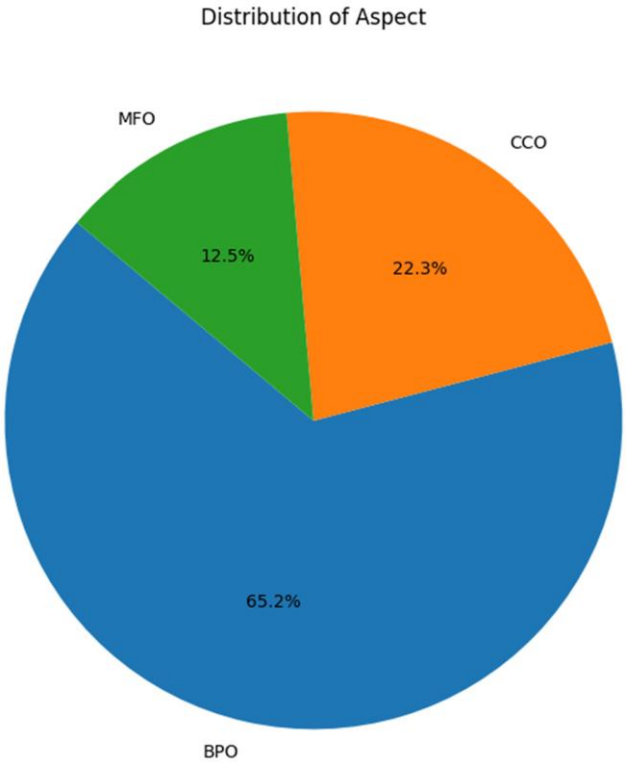


Figure 16 Distribution of Aspect

The Figure 16 illustrating the proportion of data across four categories: MFO, CCO, BPD, and an additional category that appears to start with "TH." Each slice in the pie chart represents a category, with the size of the slice indicating the proportion of data it represents.

Here's a breakdown of the information conveyed:

- MFO: Represents 12.55% of the data.
- CCO: Represents 22.3% of the data.
- BPD: Dominates the distribution with 65.2% of the data.
- THX: Represents a smaller percentage of the data that couldn't be precisely determined from the image.

While pie charts are effective for illustrating proportions within a dataset, they can be less suitable for comparing multiple categories, especially when there are many categories. In this instance, the pie chart quickly

communicates that BPD is the predominant category, followed by CCO and MFO. However, the small size of the THX category makes it challenging to discern its proportion relative to the others.

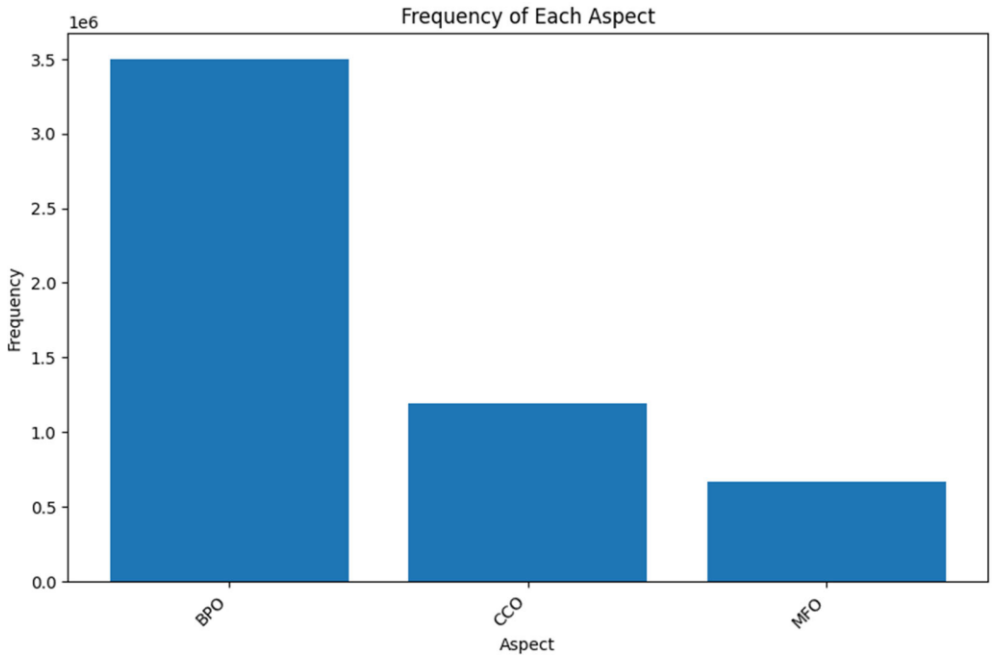


Figure 17 Frequency of Aspects

Figure 17 is a bar chart illustrating the occurrence frequency of various aspects within a dataset. This visualization aids in identifying prevalent aspects within the dataset, crucial for tasks like sentiment analysis or topic modeling.

- X-axis ("Aspect"): Lists the different aspects identified within the dataset, such as topics, products, or features.
- Y-axis ("Frequency"): Quantifies the frequency of each aspect's occurrence in the dataset. A taller bar indicates a higher frequency of occurrence for the corresponding aspect.

GO Term Analysis

```
Min GO terms per protein ID: 2
Max GO terms per protein ID: 815
Median GO terms per protein ID: 24.0
Average GO terms per protein ID: 37.70835735275509


Min unique protein ID per GO term: 1
Max unique protein ID per GO term: 92912
Median unique protein ID per GO term: 8.0
Average unique protein ID per GO term: 170.46535943558126


Min unique protein ID per GO term per aspect:
BPO 1
CCO 1
MFO 1

Max unique protein ID per GO term per aspect:
BPO 92210
CCO 92912
MFO 78637

Median unique protein ID per GO term per aspect:
BPO 10.0
CCO 9.0
MFO 5.0

Average unique protein ID per GO term per aspect:
BPO 164.3284942447733
CCO 404.46973283733513
MFO 92.76218161683278
```

Figure 18 GO Terms Analysis

From Figure 18 we can draw the following conclusions:

- **Variation in protein-GO term associations:**

  The wide range in the minimum and maximum number of unique protein IDs per GO term (from 1 to over 90,000) suggests a high degree of variation in how proteins are associated with different GO terms across the three aspects (BPO, CCO, MFO).

- **Differences in protein-GO term complexity across aspects:**

The median number of unique protein IDs per GO term is highest for BPO (10.0), followed by CCO (9.0) and MFO (5.0), indicating that on average, BPO GO terms are associated with a larger number of unique protein IDs compared to CCO and MFO.

- **Skewed distribution of protein-GO term associations**:
  The large differences between the median and average values for unique protein IDs per GO term, especially for CCO and MFO, suggest a skewed distribution where a relatively few GO terms are associated with a disproportionately high number of unique protein IDs.
- **Potential functional differences between aspects:**
  The observed differences in the protein-GO term associations across the three aspects (BPO, CCO, MFO) may reflect the underlying functional differences between the BP , CC , and MF represented by these aspects.

3.2.2 Model Benchmarks

For Benchmarking our model we've used ROC curve and Area under the ROC Curve (AUC) as the metric. The curve shows the trade-off between true positive rate (correctly predicted positive instances) and false positive rate (incorrectly predicted negative instances). AUC value is between 0 to 1. A model with all incorrect predictions has an AUC of 0.0, whereas one with all right predictions has an AUC of 1.0.

Thes are the results obtained after training the model on GPU Nvidia GTX 1060 for 18 hrs

1) 1D-CNN Model:

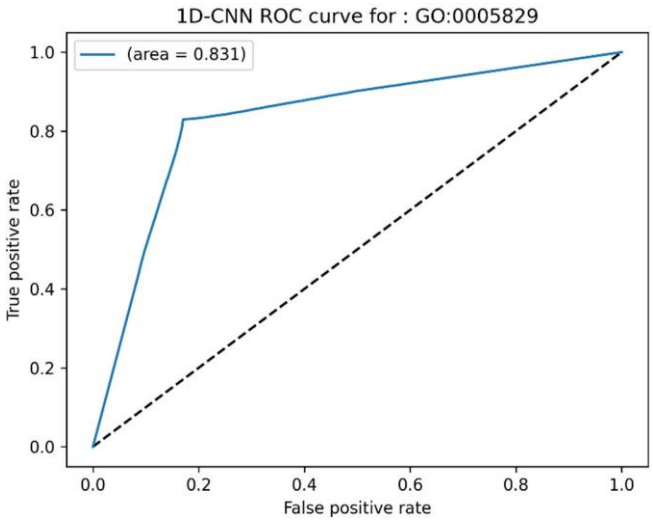


Figure 19 ROC-AUC Curve for 1D-CNN

The AUC for 1D-CNN is 0.831, which indicates the model has reasonably good discriminative power.

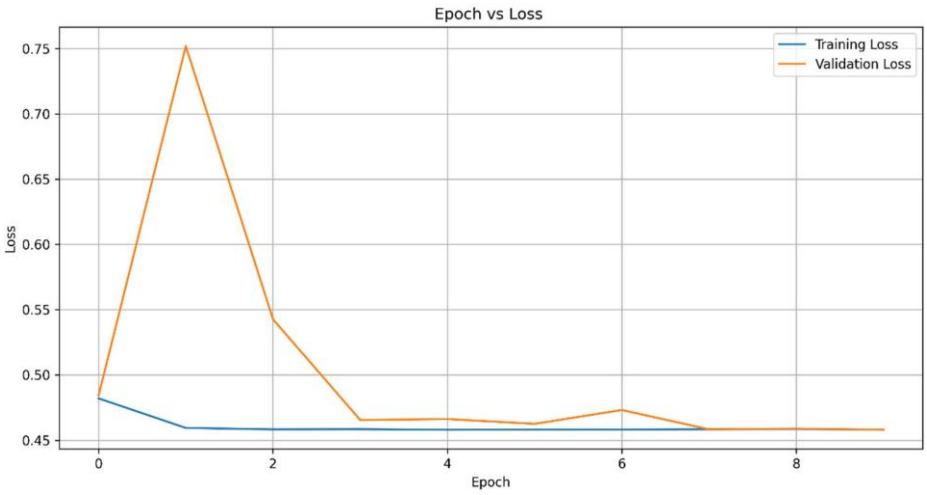


Figure 20 Training and Validation Losses for 1D-CNN

The lower validation loss compared to the training loss suggests that the 1D-CNN model can generalize well to new, unseen data.

2) LSTM Model

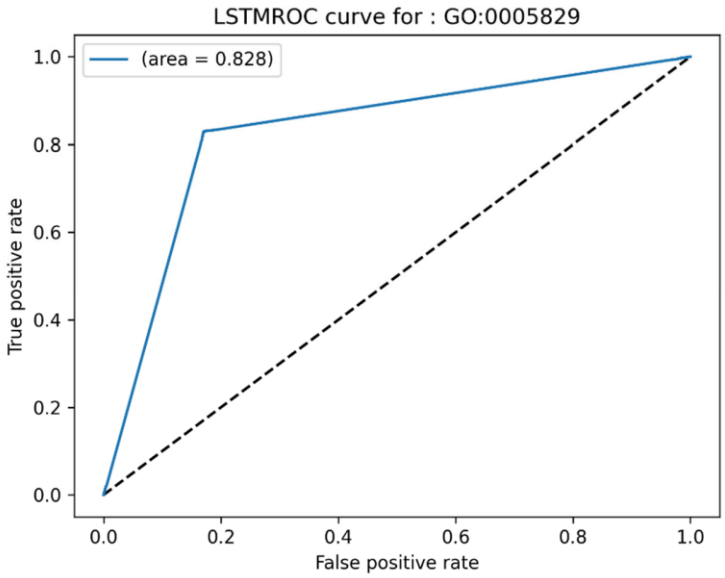
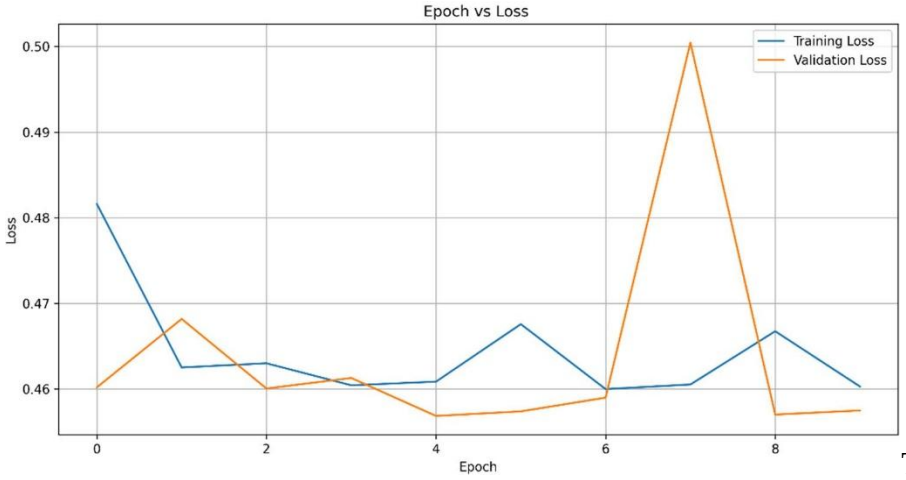


Figure 21 ROC-AUC Curve for LSTM

AUC for the LSTM model is 0.828, which is slightly lower than the 1D- CNN model's AUC.



The

Figure 22 Training and Validation Losses for LSTM Model

Validation loss is higher compared to the 1D-CNN model, indicating that the LSTM model may not generalize as well to new data.

Final Predictions

	Protein ID	GO:Term	Probability
0	Q9H3V2	GO:0005793	0.999
1	Q04744	GO:0005793	0.999
2	P61146	GO:0005793	0.999
3	P0AE06	GO:0005793	0.999
4	Q9BYD3	GO:0005793	0.999
...	...	...	...
31027	P31663	GO:0007155	0.435
31028	Q00788	GO:0007155	0.422
31029	P10908	GO:0007155	0.419
31030	Q6CDV7	GO:0007155	0.413
31031	P35284	GO:0007155	0.406

31032 rows × 3 columns

Figure 23 Prognostications showing the likelihood of falling under a specific GO word

Chapter 4
Conclusion

4.1 Summary of the results

All things considered, the 1D-CNN model turned out to be a superior option than the LSTM model. The 1D-CNN model fared better with respect of accuracy for validation than the LSTM-based model, while having equal AUC values, as seen by the smaller validation loss. For the 1D-CNN model to be used in real-world scenarios, it is crucial to take into account the possibility that the model can extract more reliable and broadly applicable characteristics based on the composition of proteins data, as shown by the reduced validation loss.

Since the model must correctly anticipate the activities of novel, previously undiscovered protein sequences, generalization performance is crucial in the field of protein function prediction. since of its advantage in validation loss, the 1D-CNN model is a superior fit for this particular task since it can identify underlying trends in the protein's sequences information as well as make accurate predictions on freshly instances.

4.2 Advantages of work

Enhanced comprehension of biological processes: Researchers can gain an understanding of how biological processes work by using protein function prediction to unravel the complexities, within cells and animals. This knowledge plays a role in shedding light on aspects of biology, such as gene regulation, metabolism and signaling pathways.

Drug research and discovery: In the field of drug research and discovery predicting protein functions accelerates the identification of drug targets. By grasping the roles of proteins scientists can develop medications that specifically target these proteins leading to treatment options.

The functional annotation of genomes: Protein function prediction also simplifies the task of assigning functions to discovered genes or genes, with roles. This process aids in enhancing our understanding of genomes through comprehensive functional annotations.

Research attempt establishing priorities: Researchers can identify proteins with high medicinal or industrial value by using protein function prediction. The most intriguing possibilities are the focus of this prioritizing, which simplifies research efforts.

Creating new enzymes and biocatalysts: By using their knowledge about how proteins function, and scientists may create enzymes with specialized roles for a range of industrial uses, including the synthesis of biofuel or the remediation of biological sites.

4.3 Scope of future work.

There are a ton of fascinating opportunities for protein function prediction in the future. Advances in sequence-based prediction involving a lower computing load are anticipated. More intricate connections between sequencing and functionalities may be captured by utilizing protein-protein interaction data and deeper neural network topologies. Furthermore, by using unlabeled protein sequences, one can enhance generalization and reveal obscure trends.

The creation of compact designs and the application of hardware-accelerated approaches are essential for achieving high accuracy and quicker outcomes. Transfer learning using large-scale datasets including models that have been trained can enhance precision and effectiveness even further. The interpretability of findings will be improved by user- friendly tools such as interactive APIs for displaying structures of proteins and Gene Ontology terms.

Ultimately, investigating cutting-edge model architectures such as generative models and graph neural networks opens the door to a better comprehension of protein function and even the production of proteins with certain functions in mind. These developments will transform the prediction of protein function and speed up drug discovery, biological research, and the creation of innovative biotechnologies.

The future of protein function prediction is focused on:

- Enhanced sequence-based prediction with less computation
- High accuracy and faster results
- Improved user experience through visualization tools
- Exploring entirely new model architectures

4.4 Unique Features (UDP)

Individualized Embeddings for Every GO Term: This novel strategy departs from the accepted practice by providing a number of benefits, including:

Personalized Education: By creating unique embeddings for every GO word, our models can focus on capturing the particular characteristics linked to each function. This targeted learning approach enables a deeper understanding of the nuances within each area of protein function.

Models Pre-Trained for Effective Prediction: Our method makes use of pre-trained models, specifically LSTM and the one-dimensional CNN architectures, and produces notable speed and accuracy gains

Efficiency: We use pre-trained models customized to each GO word instead than developing a single, universal model for all GO terms. This significantly reduces the computational load associated with protein function prediction, improving our methodology's effectiveness.

Accuracy: Making use of pre-trained models makes the most of the knowledge acquired throughout the training process. We can potentially get higher accuracy by using such models for prediction as opposed to developing a single model from start for every GO word.

In protein function prediction, this novel combination of pre-trained LSTM and a 1D CNN algorithms with focused learning has shown encouraging results. Let's investigate its possible advantages in more detail and prospective directions for growth.

References

[1] Francis Crick. Central dogma of molecular biology. Nature, 227(5258):561, 1970.

[2] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. Gene ontology: tool for the unification of biology. Nature genetics, 25(1):25, 2000.

[3] Predrag Radivojac, Wyatt T Clark, Tal Ronnen Oron, Alexandra M Schnoes, Tobias Wittkop, Artem Sokolov, Kiley Graim, Christopher Funk, Karin Verspoor, Asa Ben-Hur, et al. A large-scale evaluation of computational protein function prediction. Nature methods, 10(3):221, 2013.

[4] Brigitte Boeckmann, Amos Bairoch, Rolf Apweiler, Marie-Claude Blatter, Anne Estreicher, Elisabeth Gasteiger, Maria J Martin, Karine Michoud, Claire O'donovan, Isabelle Phan, et al. The swiss-prot protein knowledgebase and its supplement trembl in 2003. Nucleic acids research, 31(1):365–370, 2003.

[5] Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank, 1999–. In International Tables for Crystallography Volume F: Crystal- lography of biological macromolecules, pages 675–684. Springer, 2006.

[6] Jason A Reuter, Damek V Spacek, and Michael P Snyder. High-throughput sequencing technologies. Molecular cell, 58(4):586–597, 2015.

[7] Christophe Dessimoz and Nives Springer, 2017.S˘kunca. The Gene Ontology Handbook.

[8] Robert F. Weaver. Molecular biology. McGraw-Hill, 2002.

[9] William R Pearson. [5] rapid and sensitive sequence comparison with fastp and fasta. 1990.

[10] Stephen F Altschul, Thomas L Madden, Alejandro A Scha¨ffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. Nucleic acids research, 25(17):3389–3402, 1997.

[11] Leonardo de Oliveira Martins and David Posada. Proving universal common

ancestry with similar sequences. Trends in evolutionary biology, 4(1), 2012.

[12] John A Gerlt and Patricia C Babbitt. Can sequence determine function? Genome Biology, 1(5):reviews0005–1, 2000.

[13] James C Whisstock and Arthur M Lesk. Prediction of protein function from protein sequence and structure. Quarterly reviews of biophysics, 36(3):307–340, 2003.

[14] Alex Bateman, Lachlan Coin, Richard Durbin, Robert D Finn, Volker Hollich, Sam Griffiths-Jones, Ajay Khanna, Mhairi Marshall, Simon Moxon, Erik LL Sonnhammer, et al. The pfam protein families database. Nucleic acids re- search, 32(suppl 1):D138–D141, 2004.

[15] Peer Bork and Eugene V Koonin. Protein sequence motifs. Current opinion in structural biology, 6(3):366–376, 1996.

[16] Jimmy Y Huang and Douglas L Brutlag. The emotif database. Nucleic acids research, 29(1):202–204, 2001.

[17] Sridhar S Hannenhalli and Robert B Russell. Analysis and prediction of functional sub-types from protein sequence alignments1. Journal of molecular biology, 303(1):61–76, 2000.

[18] L Juhl Jensen, Ramneek Gupta, Nikolaj Blom, D Devos, J Tamames, Can Kesmir, Henrik Nielsen, Hans Henrik Stærfeldt, Krzysztof Rapacki, Christopher Workman, et al. Prediction of human protein function from post-translational modifications and localization features. Journal of molecular biology, 319(5):1257– 1265, 2002.

[19] Lars Juhl Jensen, Ramneek Gupta, H-H Staerfeldt, and Søren Brunak. Prediction of human protein function according to gene ontology categories. Bioinformatics, 19(5):635–642, 2003.

[20] Anna E Lobley, Timothy Nugent, Christine A Orengo, and David T Jones. Ffpred: an integrated feature-based function prediction server for vertebrate proteomes. Nucleic acids research, 36(suppl 2):W297–W302, 2008.

[21] Federico Minneci, Damiano Piovesan, Domenico Cozzetto, and David T Jones. Ffpred 2.0: improved homology-independent prediction of gene ontology terms for eukaryotic protein sequences. PLoS One, 8(5):e63754, 2013.

[22] Domenico Cozzetto, Federico Minneci, Hannah Currant, and David T Jones. Ffpred 3: feature-based function prediction for all gene ontology domains. Scientific reports, 6:31865, 2016.

[23] Yuxiang Jiang, Tal Ronnen Oron, Wyatt T Clark, Asma R Bankapur, Daniel DAndrea, Rosalba Lepore, Christopher S Funk, Indika Kahanda, Karin M

Ver- spoor, Asa Ben-Hur, et al. An expanded evaluation of protein function predic- tion methods shows an improvement in accuracy. Genome biology, 17(1):184, 2016.

[24] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, et al. Deep face recog- nition. In BMVC, volume 1, page 6, 2015.

[25] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Advances in neural information processing systems, pages 3104–3112, 2014.

[26] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602, 2013.

[27] Søren Kaae Sønderby and Ole Winther. Protein secondary structure prediction with long short term memory networks. arXiv preprint arXiv:1412.7828, 2014.

[28] Sheng Wang, Jian Peng, Jianzhu Ma, and Jinbo Xu. Protein secondary structure prediction using deep convolutional neural fields. Scientific reports, 6:18962, 2016.

[29] Jack Hanson, Yuedong Yang, Kuldip Paliwal, and Yaoqi Zhou. Improving protein disorder prediction by deep bidirectional long short-term memory recurrent neural networks. Bioinformatics, 33(5):685–692, 2016.

[30] Jose Juan Almagro Armenteros, Casper Kaae Sønderby, Søren Kaae Sønderby, Henrik Nielsen, and Ole Winther. Deeploc: prediction of protein subcellular localization using deep learning. Bioinformatics, 33(21):3387–3395, 2017.

[31] David T Jones and Shaun M Kandathil. High precision in protein contact prediction using fully convolutional neural networks and minimal sequence features. Bioinformatics, 1:8, 2018.

[32] Maxat Kulmanov, Mohammed Asif Khan, and Robert Hoehndorf. Deepgo: predicting protein functions from sequence and interactions using a deep ontology- aware classifier. Bioinformatics, 34(4):660–668, 2017.

[33] Rui Fa, Domenico Cozzetto, Cen Wan, and David T Jones. Predicting human protein function with multi-task deep neural networks. PloS one, 13(6):e0198216, 2018.