ARISTOTLE UNIVERSITY OF THESSALONIKI

Intrinsically disordered protein prediction for genomes and metagenomes

by

Ilias Papastratis

Student ID 66

A thesis submitted in partial fulfillment for the
Master degree

in the
Digital Media and Computational Intelligence
School of Informatics

Supervising Professor: Christos Ouzounis

February 2022

Abstract

Proteins are biomolecules that define a cell's function, architecture, sensitivity to environmental changes and lifetime. They are made up of one or more long chains of organic chemicals known as amino acids. In general, proteins have discrete and stable three-dimensional structures determined by relatively consistent atom positions. However, intrinsically disordered proteins is another class of functional proteins and protein sections, which comprises highly dynamic regions or proteins that lack an entire ordered structure. The biophysical and functional properties of intrinsically disordered proteins are represented by the composition and sequence complexity properties and have been used to develop algorithms that can predict intrinsic disorder in a protein sequence with high accuracy. Accurate prediction of disordered proteins and regions is critical for both basic research, such as protein structure and function prediction as well as practical applications, such as drug development and personalised, precision medicine.

Furthermore, the study of the genome sequences of a collection of species living in the same environment is known as metagenomics and allows scientists to examine the genomes of uncultured bacteria. Moreover, sequencing technologies are expanding and revolutionizing our knowledge of the microbial world and they can sequence uncultured microorganisms obtained directly from their surroundings. Overall, metagenomics sequencing studies have greatly expanded our understanding of the protein universe, providing over half of all currently known protein sequences.

In this thesis, we investigate the problem of intrinsic disorder prediction using machine and deep learning networks and employ prediction methods on metagenomic data. We adopt a Transformer network for intrinsic disorder prediction, which uses only protein sequences as input and creates representations with biological features to predict intrinsic disorder. In addition, transfer learning from large scale protein databases is adopted in order to increase prediction performance. The second contribution of this research is to conduct a complete comparison of several intrinsic disorder predictors in order to assess their agreement of their predicted disordered regions and explain their performance. Finally, we test intrinsic disorder predictors on assembled proteins from

simulated metagenome data as a forward-looking perspective towards this new direction.

Περίληψη

Οι πρωτεΐνες είναι μόρια που καθορίζουν τη λειτουργία, την αρχιτεκτονική, την ευαισθησία στις περιβαλλοντικές αλλαγές και τη διάρκεια ζωής ενός κυττάρου και αποτελούνται από μία ή περισσότερες μακριές αλυσίδες οργανικών χημικών ουσιών γνωστών ως αμινοξέα. Γενικά, οι πρωτεΐνες έχουν διακριτές και σταθερές τρισδιάστατες δομές που καθορίζονται από σχετικά σταθερές θέσεις ατόμων. Ωστόσο, οι εγγενώς διαταραγμένες πρωτεΐνες είναι μια άλλη κατηγορία λειτουργικών πρωτεϊνών και πρωτεϊνικών τμημάτων, η οποία περιλαμβάνει μικρότερες ή μεγαλύτερες εξαιρετικά δυναμικές περιοχές ή πρωτεΐνες που στερούνται μιας ολόκληρης διατεταγμένης δομής. Οι βιοφυσικές και λειτουργικές ιδιότητες των εγγενώς διαταραγμένων πρωτεϊνών αντιπροσωπεύονται από ιδιότητες, όπως η πολυπλοκότητας της σύνθεσης και της αλληλουχίας και έχουν χρησιμοποιηθεί για την ανάπτυξη αλγορίθμων που μπορούν να προβλέψουν την εγγενή διαταραχή σε μια αλληλουχία πρωτεΐνης με υψηλή ακρίβεια. Η ακριβής πρόβλεψη διαταραγμένων πρωτεϊνών και περιοχών είναι κρίσιμη τόσο για τη βασική έρευνα, όπως η πρόβλεψη της δομής και της λειτουργίας των πρωτεϊνών αλλά και για πρακτικές εφαρμογές όπως η ανάπτυξη φαρμάκων.

Η μελέτη των αλληλουχιών του γονιδιώματος μιας συλλογής ειδών που ζουν στο ίδιο περιβάλλον είναι γνωστή ως μεταγονιδιωματική και επιτρέπει στους επιστήμονες να εξετάσουν τα γονιδιώματα των μη καλλιεργημένων βακτηρίων. Επιπλέον, οι τεχνολογίες προσδιορισμού της αλληλουχίας των νουκλεϊκών οξέων διευρύνουν και φέρνουν επανάσταση στη γνώση μας για τον κόσμο των μικροβίων και μπορούν να εξερευνήσουν μικροοργανισμούς που λαμβάνονται απευθείας από το περιβάλλον. Συνολικά, οι μελέτες αλληλουχίας μεταγονιδιωμάτων έχουν διευρύνει σημαντικά την κατανόησή μας για ένα μεγάλο εύρος των πρωτεϊνών, παρέχοντας ένα πολύ μεγάλο ποσοστό από τις πιο γνωστές αλληλουχίες πρωτεϊνών.

Σε αυτή τη διατριβή, διερευνούμε το πρόβλημα της πρόβλεψης ενδογενών διαταραχών χρησιμοποιώντας δίκτυα μηχανικής και βαθιάς μάθησης και εφαρμόζουμε αυτές τις μεθόδους πρόβλεψης σε μεταγονιδιωματικά δεδομένα. Αναπτύσσουμε ένα βαθυ νευρωνικό δίκτυο για την πρόβλεψη εγγενών διαταραχών, το οποίο χρησιμοποιεί μόνο αλληλουχίες πρωτεϊνών ως είσοδο και δημιουργεί αναπαραστάσεις λαμβάνοντας υπόψιν τα βιολογικά χαρακτηριστικά για την πρόβλεψη της εγγενούς διαταραχής.

Επιπλέον, υιοθετείται η μεταφορά γνώσης από μεγάλης κλίμακας βάσεις δεδομένων πρωτεϊνών για την βελτίωση της αποτελεσματικότητας της μεθόδου στην αναγνώριση πρωτεινών χαμηλής πολυπλοκότητας. Η δεύτερη συνεισφορά αυτής της έρευνας είναι η διεξαγωγή μιας πλήρους σύγκρισης αρκετών προγνωστικών εγγενών διαταραχών προκειμένου να αξιολογηθεί η συμφωνία τους με τις προβλεπόμενες διαταραγμένες περιοχές τους και να εξηγηθεί η απόδοσή τους. Τέλος, δοκιμάζουμε διάφορους παράμέτρους για την παραγωγή δεδομένων προσομοίωσης μεταγονιδιώματων και χρησιμοποιούμε μεθόδους πρόβλεψης πρωτεϊνών χαμηλής πολυπλοκότητας στα παραγόμενα δεδομένα.

Acknowledgements

First and foremost, I would like to thank my supervisor professor Christos Ouzounis, for his unwavering support, advice and comments throughout this project. Under his supervision, I learnt a lot of technical and non-technical skills. I want to express my gratitude to him for putting up with my academic curiosity and entrusting me with our research cooperation, as well as for encouraging my intellectual growth and critical thinking.

In addition, I would like to express my gratitude to my adviser, Dr. Anastasia Chasapi, for sharing her scientific knowledge and making useful suggestions during. She gave me sensible advice with her solid expertise to solve my scientific and technical problems.

Also, thanks to the IDP consortium, an open community for the systematic exploration of Intrinsically Disordered Proteins.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Proteins are biomolecules that are made up of one or more long chains of organic compounds named amino acids. The proteins that are present in a cell define its function, architecture, response to environmental changes and its lifespan. Proteins differ largely in their amino acid sequence, which is governed by their genes' nucleotide sequence and generally defines that the protein folds into a specific 3D structure determining its function. In the last few years, experimental methods have improved rapidly and provide deep knowledge of the structure and function of individual proteins. In addition, sequencing techniques have created large-scale databases with whole proteomes from all living organisms. The overall opinion is that functional proteins or protein domains have distinct and stable three-dimensional structures defined by generally constant atom locations [10]. However, there is another class of functional proteins and protein sections that contains smaller or larger highly dynamic portions, while some proteins are even characterized by a total or nearly complete lacking of organized structure under physiological settings. This characteristics appear to be a key feature of their function. These proteins are referred as Intrinsically Disordered Proteins (IDPs) and their regions as intrinsically disordered regions (IDRs), respectively. Despite the fact that all IDPs lack an essential stable structure, their biophysical features range from one extreme to the other [11]. Composition, sequence complexity and conservation reflect the biophysical and functional features of IDPs. These sequence characteristics have been used to create algorithms that can accurately predict intrinsic disorder from a protein sequence. For both fundamental research, such as protein structure and function prediction and

practical applications, such as drug development, accurate prediction of disordered proteins and regions is crucial. Moreover, many techniques have been presented during the last few decades, considerably facilitating the growth of this essential topic. The first methods were computational and used manually extracted features. However, they were time-consuming and expensive and in last years research has focused on developing computational predictors using essential properties of IDRs. More recently, with the technological advances in artificial intelligence and the construction of large-scale databases of proteins, protein modelling is becoming a popular research area. Several machine and deep learning methods, such as Support Vector Machines (SVMs) [12], Deep Neural Networks (DNNs) [4, 13] and Recurrent Neural Networks (RNNs) [5, 14], have been applied for protein representation learning and have achieved outstanding performance on several tasks, such as secondary structure prediction [15–17], contact prediction [16, 17] and intrinsic disorder prediction. In the last years, deep learning methods, such as transformers have achieved outstanding performance on Natural Language Processing (NLP) tasks [18–20]. Similarly to natural languages, proteins can be thought as the language of life and language models have been successfully generalized to model protein sequences [21, 22] with amino-acids being analogous to words. To this end, several works have applied language models to model proteins [16, 17] and capture their structural properties. In addition, over the last years, new sequencing technologies have resulted in a massive increase in the size of protein databases. Furthermore, these large unlabeled datasets of protein sequences are likely to include significant biological information due to billions of years of evolution sampling the regions of protein sequence space that are important to life. However, annotation of this datasets is expensive and time-consuming and scientists have focused towards self-supervised and unsupervised techniques extract useful biological information [17, 23].

In this work, we adopt a Transformer network for intrinsic disorder prediction. The Transformer will take as input only sequence data use protein and generate representations that contain biological characteristics. This is accomplished with self-supervised training on large-scale protein datasets. Then, in the classification step, these representations are used for intrinsic disorder predictions. The second contribution of this work is to do a comprehensive comparison between different IDP predictors to test the agreement between their predictions and explain their performance. Finally, we employ IDP methods on simulated metagenomic datasets and analyze the results regarding to the

simulation parameters and investigate their effect on the disorder content of proteins. The latter effort provides a novel perspective, as these methods have not bee applied to metagenomes systematically, opening up new avenues for original research.

The rest of this work is summarized as follows.

In Chapter 2, we describe related work and stat-of-the-art methods about compositional bias detection, intrinsic disorder prediction and protein language models. We describe computational methods at first and then we focus on machine-learning and deep-learning methods.

In Chapter 3, we describe in more detail the algorithms from the bibliography that we use during this dissertation in our proposed frameworks that we analyze in Chapter 4.

In Chapter 4, we introduce a novel approach for intrinsic disorder prediction based on Transformer networks. Our approach is able to learn useful protein representations and achieves a good performance at identifying intrinsic disordered regions from input protein sequences. Then, we describe the framework that we implement to compare IDP predictors. Finally, we propose a new framework to simulate metagenomes with different simulation parameters and assemble proteins from the simulated datasets. Then, we run several IDP methods on the assembled protein sequences to find relations between the simulation parameters and the ratio of the disordered proteins.

In Chapter 5, we discuss the experimental setup that we use in this work. We describe the datasets, the metrics and the optimization steps that we used in our framework. Then, we compare our proposed method for IDP with several state-of-the-art methods and analyze the results. Furthermore, we present and discuss the results of our second framework that compares IDP predictors. Finally, we conduct experiments on simulated metagenomic data.

In Chapter 6, we conclude our work and discuss future work ideas.

Chapter 2

Related Work

Proteins are very complex molecules found in all living things. They have a high nutritional value and they play a direct role in the chemical reactions that keep life going, while their significant importance was discovered in the early nineteenth century. A protein molecule has a large molecular weight and it is made up of several amino acids linked together in long chains, much like beads on a string. Proteins are made up of 20 distinct amino acids that exist naturally. In the next paragraphs, we discuss the main algorithms that have been applied on modelling essential characteristics of proteins such as biological properties and intrinsic disorder regions. Then, we will discuss related method on the metagenomics area and the sequencing technologies, which are responsible for discovering massive amounts of protein sequences in the last years.

2.1 Protein Language Models

The meaning of natural language, which is made up of letters, such as the alphabet, is deduced and formed using grammar and semantics. Language models have been widely adopted in a variety of research areas such as speech recognition and natural language processing to effectively model the linguistic information and knowledge inherent in natural languages. Biological sequences can be thought of as sentences with distinct letters and characteristics like as structure or function can be defined using biophysical and biochemical principles. Moreover, biological sequences are similar to natural

language, since they use characters to define their meaning and the meaning is dependent on the meaning of the sequences around them. A Protein Language Model (PLM) is a successful machine-learning tool for extracting information from large protein sequence databases. These models find evolutionary, structural and functional organization across protein space using just readily available sequence data. We can encode amino-acid sequences into distributed vector representations that reflect their structural and functional features using language models and we can also assess the evolutionary fitness of sequence variants using language models.

Bepler *et al.* in [24], provide a framework that converts every protein sequence into a series of structurally encoded vector embeddings (one for each amino acid position). A feedback process is adopted in order to train bidirectional long short-term memory (LSTM) models on protein sequences, using input information from pairwise residue contact maps for individual proteins as well as global structural similarity across proteins. Moreover, a similarity measure across arbitrary length sequences of vector embeddings is computed using soft symmetric alignment (SSA). Finally, without knowledge of the position level connections between sequences, their method is able to model relevant position-wise embeddings. In [21], the authors adopt the deep bidirectional model ELMo [25], that has been successfully deployed in NLP tasks and train it on a large scale protein dataset to transform protein sequences as continuous vectors. This is just one example of a vast number of computational models for protein sequences, developed over the past half century, since the early 1970s.

2.1.1 Self-supervised/Unsupervised protein language models

Self-supervised training of deep learning methods has enabled breakthroughs on representation learning and statistics generation in the field of artificial intelligence. Recently, artificial intelligence (AI) researchers have made significant progress in constructing AI systems that can learn from large volumes of labelled data. Self-supervised approaches, unlike supervised learning, which involves manual annotation of each datapoint, leverage from large-scale volumes of unlabelled data. There are also several tasks for which there are insufficient labelled data, such as natural language processing (NLP) tasks, translation systems for low-resource languages and medical applications. In addition, self-supervised learning on large amount of data helps the models to learn

meaningful representations and perform well on downstream tasks. In NLP research, self-supervised learning has been applied usually in the form of predicting the next token, masked token prediction as well as classifying the next sentence.

More recently, self-supervised learning was applied to computational biology and bioinformatics tasks. Similarly to NLP tasks, it has been confirmed that unlabelled protein sequences include important structural and functional information and can be modelled in the same manner by describing proteins as discrete token sequences. In [23], Rao *et al.* pretrain several models with unsupervised techniques and conclude that deep-learning models benefit significantly and achieve better performance on a variety of tasks with biological sequences. Elnaggar *et al.* in [17], train deep Transformer models, such as BERT and Transformer-XL on large scale datasets to extract embeddings without the need for Multiple Sequence Alignments (MSA) or evolutionary information. The authors used large datasets of protein sequences for self-supervised training such as UniRef50 [26]. To evaluate the output embeddings of the Transformer models, they used them for several downstream tasks, such as secondary structure prediction and outperformed state-of-the-art methods. Similarly, in [16], Rives *et al.* used unsupervised learning to train very deep language models on million of protein sequences with evolutionary information. More specifically, the masked language modelling is used to train the models, by masking a fraction of the amino acids for each input sequence. It is shown that these models benefit from the unsupervised optimisation and learn powerful representation as well as fundamental physical properties of proteins. Then, they are employed on several downstream tasks, such as long-range residue–residue contacts prediction, remote homology detection and prediction of secondary structure and manage to achieve competitive results.

2.2 Detection of intrinsically disordered proteins

Up to the 20th century, it was assumed that proteins should have a solid structure. According to that assumption, the amino acid sequences adopt a robust and solid three-dimensional structure, which characterises the function of the protein. The essential principle of this paradigm is that 3D structure is assumed to be a prerequisite for protein function, hence natural protein structure is equivalent to organised 3D structure.

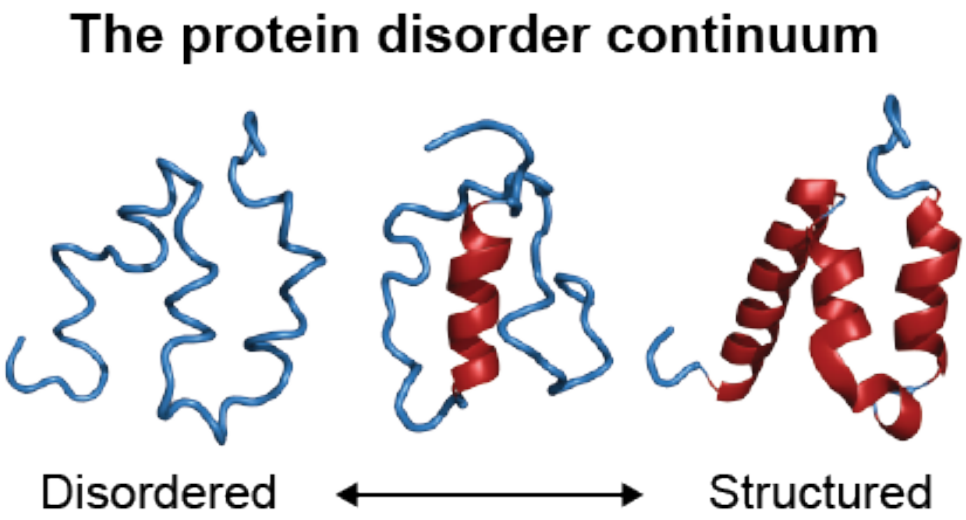

FIGURE 2.1: An example of ordered and disordered proteins [1]. Disordered proteins lack the fixed structure compared to ordered ones.

However, it was later discovered that this principle is not fundamental and there are also other non-structured forms of proteins. The ordered state, the molten globule and the random coil are three different forms that proteins can take. Molten globules are proteins with partially folded regions and lack the solid structure of fully ordered proteins. Random coil refers to proteins that have no secondary structure, while the only fixed connections are those between adjacent amino acids. However, any of the three states (not only the ordered state) can correspond to natural protein structure and have an important role in protein function. In structural biology, intrinsically disordered proteins and regions are proteins that do not assume a stable, three-dimensional fold under physiological conditions. These, protein sequences feature a surprising number of low-complexity local areas, while different types of residue clusters, some of which contain homo-polymer or brief period repeats. The length of IDRs is one of the most widely utilized properties [27] and is used to split IDRs into two categories, long disordered regions (LDR) and short disordered regions (SDR). LDRs typically contain more than 30 amino acid residues, while SDRs typically have 30 amino acid residues. Because of their role in illnesses including Alzheimer, Parkinson and cancer they are also attractive for generating therapeutic treatments. Several methods for studying the links between amino acid sequence and order or disorder have been established and we may predict intrinsic disorder from amino acid sequence using this knowledge. The structure connections reveal that disorder is encoded and the predictions strongly suggest that proteins in nature have substantially more inherent disorder than those in the Protein Data Bank. In

the following section, we will describe the methods for detecting disordered regions in proteins.

2.2.1 Compositional bias detection

Wooton *et al.* [28] proposed SEG, one of the first approaches for segmenting low-complexity regions. SEG is based on the similarity scores of self-aligned sequences at various offsets, as well as several metrics like the log likelihood function of complexity based on information theory informational entropy and multinomial probabilities of the observed composition. SEG has been tested on the well-known SwissProt database. Promponas *et al.* [29], proposed the CAST algorithm, which compares the query sequence against 20 homopolymers with unlimited gap penalties using a multiple-pass Smith–Waterman comparison. The technique produces a masked output sequence that is used for additional analysis, such as searching against databases, as well as detecting low-complexity areas. Low-complexity region identification is extremely selective for single residue types and this method is proved to be adequate for concealing database query sequences while avoiding false positives. The method's key benefit over comparable techniques is its ability to selectively mask particular residue types without harming other potentially relevant locations. In [30], the authors created a database server named LPS-annotate to annotate such low complexity regions and analyze the disorder in protein sequences. Through a comprehensive search for low probability subsequences (LPSs), the method determines the least likely sequence regions in terms of composition. Input proteins or nucleotide sequences of interest can be annotated by detecting such areas. More specifically, the algorithm scans the input sequence with different window lengths and calculates the lowest probability for each windows. Then, they are used as query input to a database of over one million pre-calculated low complexity regions to acquire further functional properties and information about the protein's disorder. The main advantage of this method is that LPS sections of different amino acid residue types may be allocated entirely and accurately, with clearly defined borders. Harrison *et al.* in [31], proposed the fLPS method that is based on the LPS algorithm but incorporates a number of novel steps to improve efficiency, by reducing notably the computation of probabilities unless it is absolutely required. It also features a new feature that allows you to change user-defined settings. Subsequently, it analyzes compositionally biased

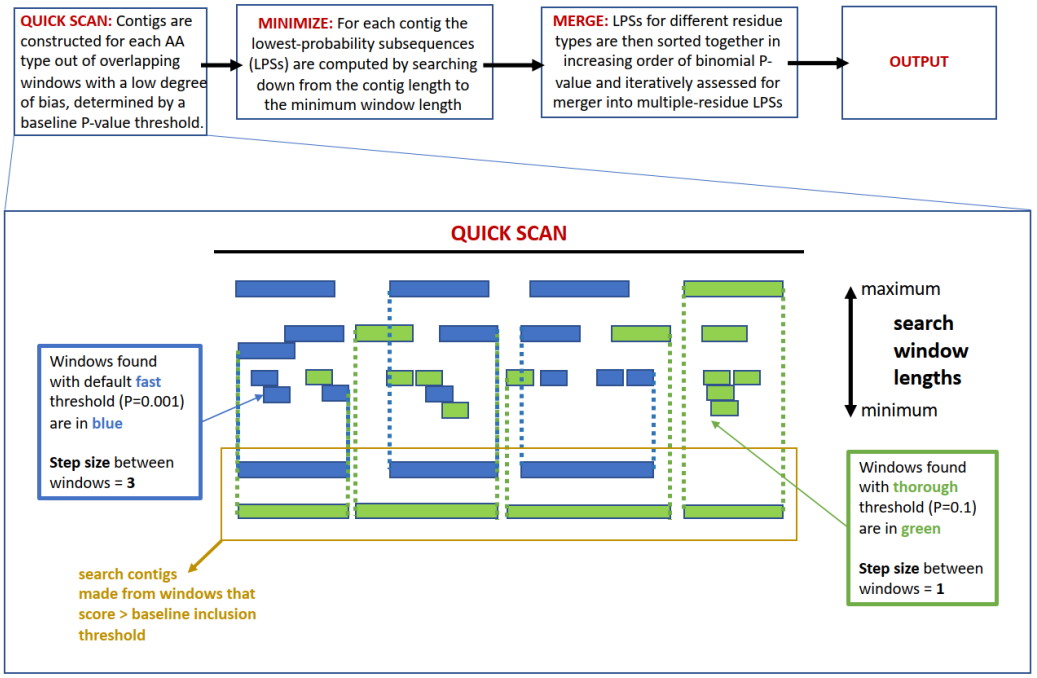

FIGURE 2.2: Overview of fLPS algorithm [2].

sections faster than other current algorithms and can detect extremely modest biases over large segments of sequence as well as severe biases over short spans. The limits of low complexity areas are determined by examining the quantity of each particular amino-acid type one at a time. Finally, fLPS generates lists of compositionally biased areas that are labeled based on their amino-acid composition as well as takes into account the different background frequencies of different residue kinds. In the latest version of the algorithm (fLPS 2.0) [2], several options have been added in order to set the precision of the method, while subsets of residue types and existing domain annotations can be set in order to find discontinuous biased and filter sequences.In addition, the algorithm's baseline precision can be tweaked to find more moderately biased areas with biological importance. The workflow of fLPS 2.0 method is shown in Figure 2.2.

2.2.2 Intrinsic disorder prediction

Early methods for intrinsic disorder prediction used manual extraction of characteristics including nuclear magnetic resonance (NMR), X-ray crystallography, circular dichroism (CD) spectroscopy, small-angle X-ray scattering (SAXS) and single-molecule fluorescence resonance energy transfer (smFRET) [27]. Dosztányi *et al.* [32] proposed IUPred, which is based on the physical explanation of the disordered characteristics of proteins. More specifically, globular proteins create a lot of interresidue contacts with non-covalent bonds (such as salt bridges, hydrogen bonds or van der Waals interactions) that provide enough stabilizing energy to counteract entropy loss during folding. Since intrinsically unstructured/disordered proteins contain unique sequences that are unable to create sufficient interresidue connections, IUPred aims to identify these characteristics. Nevertheless, the IUPred model predicts such regions from amino acid sequences by calculating their total pair-wise interaction energy between residues. In addition, IUPred is designed to predict short or and long disordered sections and structured domains are available as an option to the prediction. ANCHOR [33, 34] is a similar method to IUPred that uses energy estimates to represent the essential biophysical features of disordered regions and can distinguish them from organized proteins. ANCHOR focuses on specific and important characteristic of IDPs, which function differently in isolation than they do when attached to their companion protein. IUPred2A [3] is a refined method that combines IUPred2 and ANCHOR2 to give energy estimation-based predictions for ordered and disordered residues and disordered binding areas. IUPred2 is an updated version of IUPred with minor bug fixes, while with a new architecture and parameters modified on multiple datasets, the next version of ANCHOR is substantially improved. In addition, redox-sensitive portions may now be highlighted utilizing a unique experimental feature.

DisEMBL [4] was one of the first attempts to predict disordered regions using artificial neural networks. This method calculates the likelihood of disordered portions in a protein sequence with three predictors. The first network is trained to identify regions without regular secondary structure and can recognize almost half of the negative samples while discarding almost none of the positive ones. A second network is used to detect regions with high B factors i.e., hot loops, while a third network is combined with the second to analyze missing coordinates. Extracting long-range dependencies

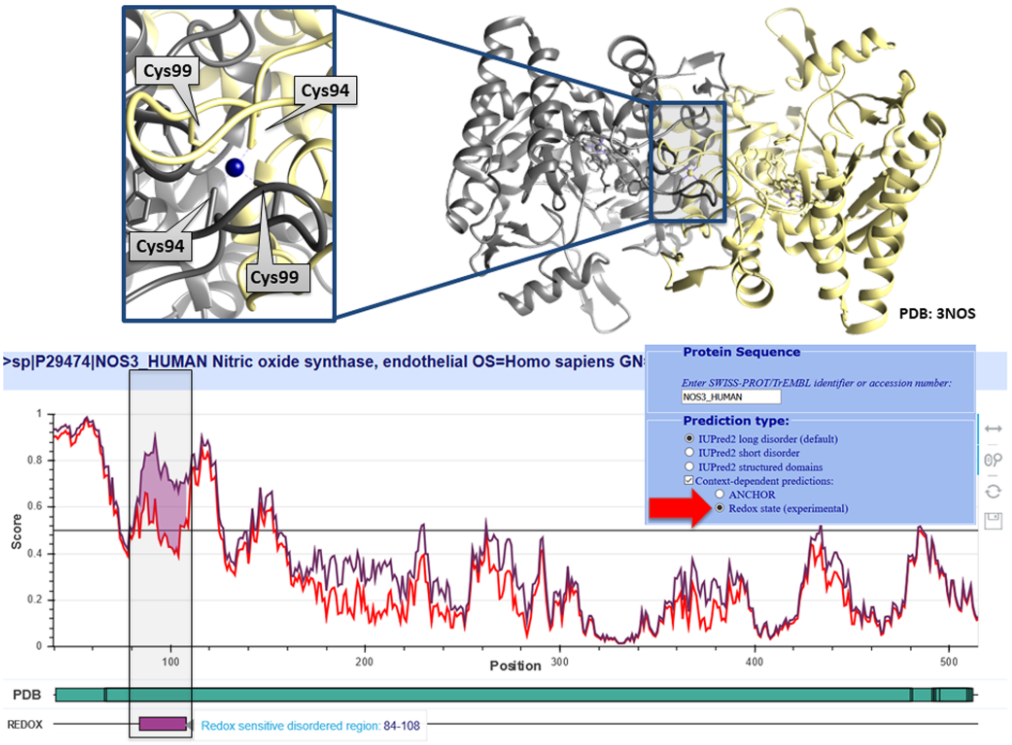

FIGURE 2.3: Example of IUPred predictions [3]. The threshold for classifying a disordered protein is the black line (score¿=0.5).

between protein sequences has been a very challenging task in bioinformatics. Recent advances in deep neural networks and especially recurrent neural networks (RNN) aim to capture prior information in sequential data. More specifically, long-short term memory (LSTM) networks are able to learn long sequential information and deal with the vanishing gradient problem of vanilla RNNs.

Walsh *et al.* , in [35], developed ESpritz a method based on bidirectional recurrent networks that used only the amino acid sequence as input. For input, Espritz employs the five Atchley sequence metrics as numerical sequence characteristics that represent polarity, molecular volume, secondary structure and codon. Normalization is also performed due to the different range of these features as well as evolutionary information from multiple sequence alignments is used to improve overall performance. In addition, this method is developed to produce high-throughput predictions without sacrificing accuracy. In [36], Marco *et al.* implemented MobiDB-lite, which is a fusion method from eight distinct predictors. This method was designed to make very precise predictions of long disordered regions. The outputs of the eight predictors were processed

FIGURE 2.4: Predictions of DisEMBL [4], where red color corresponds to disordered regions and blue to ordered regions.

in order to remove false predictions and provide the best possible annotation. It has been experimentally shown that MobiDB-lite can combine efficiently the different architectures and improve specificity. Zhang *et al.* , in [37], developed a single neural network to predict short and long disordered regions, named Sequence based Prediction with Integrated NEural network for Disordered residues (SPINE-D). Projected torsion angle variations, predicted secondary structure and solvent accessibility are used as input features. The method was simultaneously trained with long and short IDRs and is insensitive to the adopted training dataset. Wanf *et al.* in [38], proposed AUCPred that combines deep convolutional neural networks (DCNN) with conditional random fields (CRF). This method was able to simulate sequence–structure relationships and associations between adjacent residues in a hierarchical fashion. To deal with the class-imbalance problem, the method was trained in order to maximize the area under curve metric (AUC). Hanson *et al.* , in [14], developed a new method, namely SPOT-Disorder that is based on deep bidirectional LSTMs in order to capture long-range interactions and decide whether a protein will fold or not into a distinct three dimensional structure.

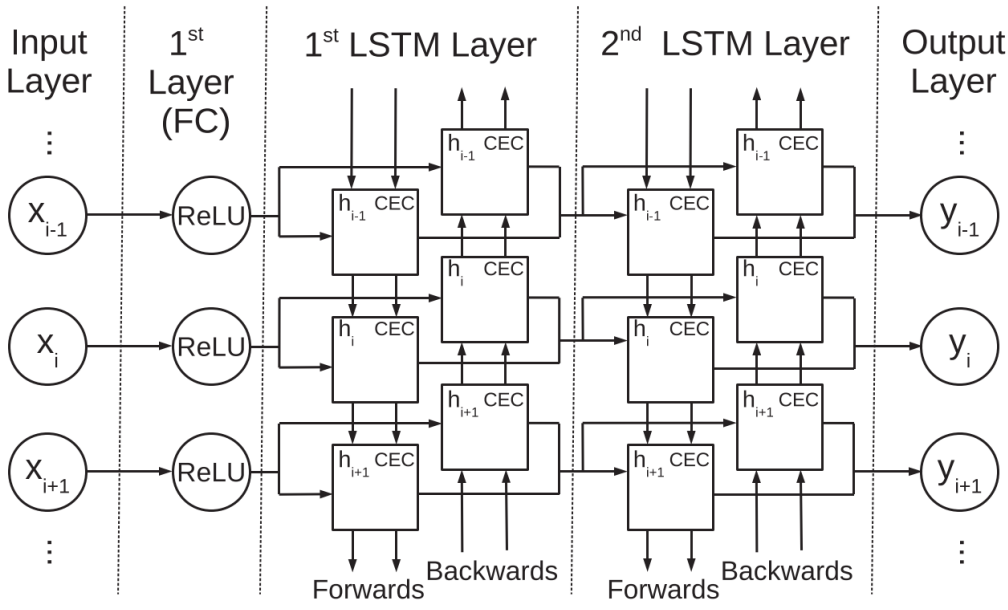

FIGURE 2.5: Architecture of Spot-Disorder2 [5] based on bidirectional LSTMs and fully connected (FC) layers.

In more detail, the bidirectional LSTM (BLSTM) is found to predict very precisely short and long regions by using both forward and backward information from sequences. In a later work [5], the authors proposed an improved method SPOT-Disorder2, which outperforms the previous architecture based only on LSTM networks in terms of accuracy and consistency. SPOT-Disorder2 adopts residual convolutional layers to learn short-distance interactions between amino acids, solve the vanishing gradient problem and adopt a deeper architecture to learn more powerful representations of IDPs. Moreover, this method utilizes Squeeze-and-Excitation networks that compress output features into an excitation signal. Similarly, in [39], Tang *et al.* implemented an recurrent neural network based on the sequential learning approach (Seq2Seq) that is commonly adopted for Natural Language Processing (NLP). Several feature such as the Position-Specific Scoring Matrix (PSSM) and Residue–Residue Contacts (CCMs) were employed by IDP-Seq2Seq to collect evolutionary data and map the protein sequence to a latent space and model the structural patterns. In addition, an attention mechanism was applied to extract global relationships between protein residues. Three different predictors were trained to predict short, long and all intrinsically disordered regions, respectively.

Hu *et al.* in [7], proposed FlDPnn method that uses several tools to process the input

sequence and provide important putative structural and functional information such as DNA and protein bindings. These were used to create the input features, which are aggregated at three levels, residue, window and protein level, respectively. Finally, these characteristics were then fed into a machine-learning model that was trained to generate disorder predictions and functions.

In this section, we presented very sophisticated algorithms that have been developed for protein structure and disorder prediction. Similarly, the field in sequence analysis has advanced greatly to the extent that we can now identify the source of genetic material in extremely complex and segmented mixtures such as Metagenomics of Built Environments (MBE) datasets.

2.3 Metagenomics

The study of microorganisms that exist in every region of the world has faced several challenges over time, including the discovery of new species and a better knowledge of how they interact with their surroundings [40]. In early years, microbial communities were studied mostly in laboratories using microscopes. More specifically, microorganisms were studied based on their shape, growth and the selection of biochemical profiles. These approaches gave us the first knowledge for the microbial world. Later on, ribosomal RNA genes were used to categorize these communities. In addition, several methods were proposed such as the polymerase chain reaction (PCR), rRNA gene cloning and sequencing, fluorescent *in situ* hybridization (FISH), denaturing gradient gel electrophoresis (DGGE and TGGE), restriction-fragment length polymorphism and terminal restriction-fragment length polymorphism (T-RFLP), to describe microbial diversity [41]. Despite these advancements, many additional microbiological discoveries, such as those relating to the metabolic and ecological functions of microbes, remained unexplained. In contrast, for some species, only after cloning a gene from total DNA and associating it with a certain metabolic function, we were able to characterise certain activities resulting in the development of genomics. Genomics is the research field that aims to identify whole genomes present on organisms by high-throughput sequencing of the base pairs of its DNA. Metagenomics, on the other hand, is the study of the genome sequences of a group of species living in the same environment and allows scientists to

analyse the genomes of uncultured microbes. Moreover, metagenomics has also been described more generally as any sort of examination of DNA taken directly from the environment i.e., screening DNA for a specific enzyme activity. However, the data in metagenomics comes from diverse microbial communities, which can comprise more than 10,000 species and the sequencing data is noisy and incomplete. Our understanding of microbial diversity has been biased by this culture bottleneck and our appreciation of the microbial world has been constrained. Finally, metagenomics gives a generally impartial assessment of a community's metabolic capacity as well as the structure of the community. To this end, sequencing technologies are increasing and revolutionising our understanding of the microbial world, while they have the ability to sequence uncultured microorganisms collected straight from their environments. The genomic data in cultivated bacteria comes from a single clone, making sequence assembly and interpretation simple. Despite the fact that metagenomics is a relatively new subject, computational approaches for metagenomic research have exploded in recent years, as well as a plethora of metagenome shotgun sequence datasets has been recorded.

The typical steps of a sequence-based metagenome project are the following: sampling, sequencing technology, assembly, binning, annotation, experimental design, statistical analysis and data storage and sharing. These steps will be explained in the following subsections.

2.3.1 Sampling

In each metagenomics study, sampling is the first and most important phase [42, 43]. DNA extracted from the sample should be representative of all cells present, while adequate quantities of high-quality nucleic acids must be acquired. Collection of high-quality DNA is a major bottleneck in genomic studies because there is no specific extraction approach that suits all environmental samples. Filtering is also an important part in order to get as much information of the target cells as possible and leave out non-target material to avoid contamination of the sample. Physical separation of the required cells from samples may also be necessary in order to guarantee a representative DNA extraction. Moreover, there are cases where only a little quantity of DNA is found in some types of samples. In these cases, amplification of the starting material is necessary, because most sequencing methods need nanograms or micrograms of DNA.

The final step is to keep records of the metadata of the samples such as the location of the samples, the sampling conditions, the date, pH, the temperature, the depth in marine and altitude in terrestrial samples. This information might be important in order to find correlations between species and environments that might lead to important breakthroughs.

2.3.2 Sequencing technologies

DNA sequencing is a technique for determining the exact sequence of bases (cytosine (C), guanine (G), adenine (A) or thymine (T)) in a DNA molecule in a laboratory setting. This information is important for a cell as it contains the instructions to build proteins and RNA molecules that are included in the DNA base sequence, as well as to annotate the functioning of genes. The Sanger sequencing technology [44] was one of the first sequencing technologies and is still considered the main baseline. The first step is to shear the DNA content into random fragments commonly known as ”shotgun”. After that, the fragments are cloned onto plasmid vectors and cultured in monoclonal libraries to provide enough genetic material for sequencing. Then, dye-termination procedures are used to sequence the DNA. This method is repeated numerous times to guarantee that all regions of the genome under study are sequenced. The genetic segments are subsequently assembled into the whole genome using assembly tools. However, large genomes cannot be fully recovered while it is a time-consuming cloning method and usually biased against genes that are toxic with respect to the cloning host. Nonetheless, because of its low error rate and high read length, Sanger sequencing is still used nowadays [42].

The current standard for the collection of genomic data is next-generation sequencing (NGS) technologies, which produce large volumes of data quickly and at a small cost. With these approaches we are able to rapidly sequence DNA and RNA and collect large amounts of genomic data. NGS technologies offer quicker sequencing duration and lower costs than traditional Sanger sequencing due to their higher degree of parallelism and smaller reaction volumes, but at the trade-off of increasing error rates and shorter read lengths. One of this technologies that is widely used in metagenomics, is pyrosequencing, which detects pyrophosphate (PPi) release from the DNA polymerization and generates light on nucleotide incorporation. The system detects light emission from a

plate with millions of microwells holding a specific DNA fragment and converts it to nucleotide sequences. This approach provides a greater yield than Sanger sequencing at a cheaper cost but with shorter read lengths, while it can have error on long homopolymeric regions [43]. Although NGS certainly accelerates the acquisition of massive data sets, downstream processing remains a significant barrier.

2.3.3 Assembly

Assembly is the process of merging sequence reads into continuous sections of DNA called contigs and is based on sequence overlaps between reads. This is necessary because DNA sequencing technologies cannot detect whole genomes in a single pass, but rather reads little chunks (reads) at a time. The assembled contig is either constructed based on the highest-quality nucleotide in each given read at each location or on the majority voting [45]. For the construction of full genomes two main techniques are adopted: reference-based assembly (co-assembly) and *de novo* assembly.

Reference-based assembly software work well in cases, where the data contain genomes similar to the available reference genomes. These software packages offer rapid and memory efficient algorithms that may typically be completed in a few of hours on laptop computers. However, if the genuine genome of the sample differs from the reference genome, the assembly may be fragmented and ambiguous sections may not be recovered. On the other hand, without the use of a reference template, *de novo* assembly software attempts to assemble small reads into full-length sequences. *De novo* assemblies are orders of magnitude slower and more memory heavy than reference assemblies in terms of complexity and time needs. This is primarily due to the algorithm's requirement to compare each read with each other, resulting in a time complexity of $O(n^2)$. Different types of graph-based algorithms are adopted by current *de novo* genome assemblers, such de Bruijn Graphs (DBG) and greedy graph-based techniques. DBG-based methods, which are widely adopted, rely on K-mer graphs to represent related genomes and can handle large amounts of data. However, *de novo* assemblers require more computational power compared to reference-based ones due to the graph construction and the time-consuming pairwise comparisons.

2.3.4 Binning

Binning classification is a simple and convenient approach for predicting taxonomic composition from read data and can be performed using either reads or assembled genomes, with the most reliable binning classification using assemblies. There are two commonly adopted strategies:

- classification-based binning using sequence composition
- similarity-based binning using alignments of the sequence against references

Classification-based binning incorporates k-mer frequencies approaches, in which short words (k-mers) are used to represent the sequence and finally the similarities between all words in the query are calculated. The advantage of k-mer–based binning is that it does not require any reference sequences for the actual binning. As a result, k-mer is an effective technique for binning sequences with few or no homologs and unrecognized function. In contrast, similarity-based binning is a searching method for binning sequences, which looks for similarities against reference sequences and classify the gene using the highest similarity. This approach has proven to be beneficial, when the majority of the sequences in a sample are highly comparable to reference sequences from known genomes. In addition, these methods are very fast and reliable [46]. To employ the most suitable approach for binning we should check the type of the input data and the availability of reference databases to search for similar genomes. In general, classification-based binning is not suitable for short reads since they don't provide sufficient information. However, short reads can have similarity compared to reference genomes and be classified to specific genes.

2.3.5 Annotation

Two alternative methods can be used to annotate metagenomes [42]. First, if the goal of the study is to rebuild genomes and the assembly process has resulted in big contigs, it is better to employ existing pipelines. On the other hand, unassembled reads or short contigs can also be used to do annotation on the entire community. However, the

genome annotation tools are substantially less accurate in the latter case. In general, annotation is a two-step process, where, the key characteristics (genes) are found and then, the gene functions and the taxonomic neighbors (functional annotation) are assigned. The method of classifying sequences as genes or genomic elements is known as feature prediction. Algorithms that take into account di-codon frequency, preferential bias in codon usage, patterns in the use of start and stop codons and if possible, information about species-specific ribosome-binding sites patterns, ORF length and GC content of coding-sequences are more suitable for gene prediction. Functional annotation is a crucial computational task for metagenomic project. According to current estimations, only 20 to 50 percent of a metagenomic sequence can be annotated, leaving the urgent question of the remaining genes' relevance and function. It should be pointed out that annotation is not done from scratch, but rather by mapping existing information to gene or protein libraries.

2.3.6 Simulation tools

Computer simulations of genetic and genomic data are becoming more common as a means of evaluating and verifying biological models, as well as acquiring a better knowledge of specific data sets. Simulations can be used as a guide for developing new computational tools, troubleshooting and assessing software performance on its own. Computer simulations also help us create new ideas, aid in the design of sequencing projects and are critical for verifying various conclusions, such as the correctness of an assembly, the accuracy of gene prediction and the ability to rebuild accurate genotypes and haplotypes. In recent years, a number of computational methods for simulating NGS data have been created. MetaSim [47] is a simulation tool for genomics and metagenomics datasets, with a set of known genome sequences and an abundance profile are used as input. The relative abundance of each genome sequence in the dataset is determined by this profile, which decides which genome sequences are chosen for simulation. In addition, MetaSim includes a "induced tree view" of the NCBI taxonomy that may be used to adjust relative abundances of taxa and inner nodes of the taxonomy. Using a population simulator, the user may also simulate an evolved population with a single genetic sequence. This feature aims to simulate the typical real-world circumstance in which a lineage's many diverse but closely related strains live in the same environment.

Finally, MetaSim features a flexible read sequencing simulator for creating a realistic read data collection. FASTQSim [48] is a tool that combines the characterisation of NGS datasets with the production of metagenomic data. It works with any sequencing platform and can calculate distributions of read length, quality scores, indel rates, single point mutation rates and the indel size. In addition, FASTQSim can transform target sequences into *in silico* reads with precise error profiles obtained during the characterization process to build training or testing datasets. create standardized test scenarios for planning sequencing projects or assessing metagenomic tools by simulating individual read collections. The quality of the NGS datasets is displayed since read length, read quality, repetitive and non-repetitive indel profiles and single base pair replacements are all provided by the tool. CAMISIM [49] is a simulation tool that was created to generate the simulated metagenome data sets utilized in the initial CAMI challenge. Many aspects of the created communities and data sets may be customized using CAMISIM, including the total number of genomes (community complexity), strain diversity, community genome abundance distributions, sample sizes, number of replicates and sequencing technique. At first, CAMISIM begins with the ”community design” stage, which may be done from scratch (requires a taxon mapping file and reference genomes) or using a taxonomic profile. This stage generates a community genome and taxon profile, which is then utilized in one of four read simulators to simulate the metagenome (ART, wgsim, PBsim, NanoSim).

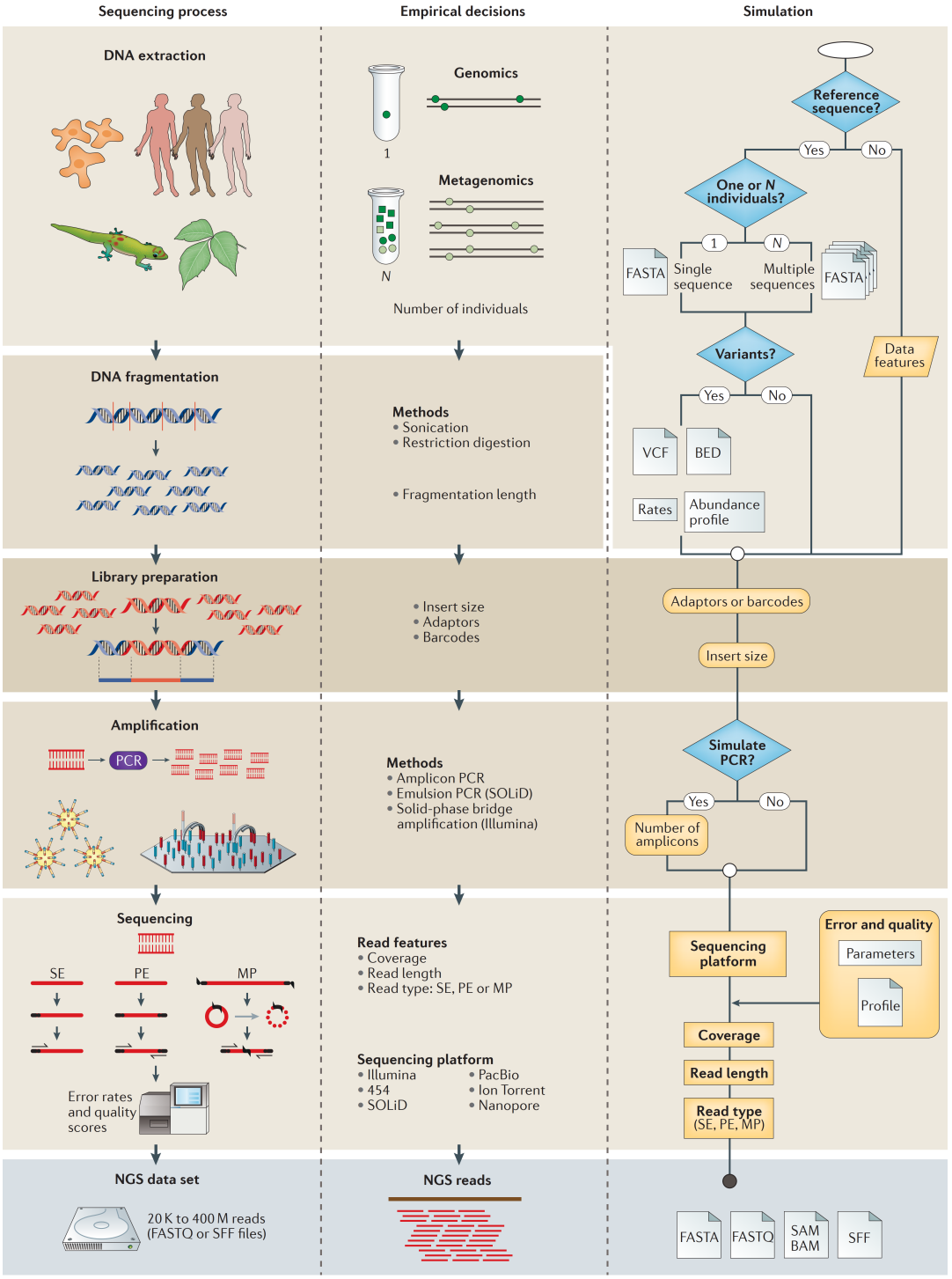

FIGURE 2.6: Overview of the metagenomic simulation process. [6]. The acronyms and the abbreviations of the figure are fully described in reference [6].

Chapter 3

Relevant Approaches and Software

In this section, we will present an overview of IDR methods that are applicable to any sequential structure, such as protein sequences and DNA sequences.

3.1 Adopted intrinsic disorder prediction methods

3.1.1 CAST

CAST is dynamic programming algorithm that identifies low complexity sequences based on alignment to homopeptides of the 20 common amino acids. The chance of finding a match between two residue categories a, b, of a search and a test sequence, is calculated from the joint probability (p_{ab}) as:

$$p_{ab} = f_a \, p_b, \tag{3.1}$$

where f_a, p_b are the fractions of a, b. This is based on the assumption that the fraction percentages of two residue types in a search and test sequence are statistically independent events, which equals the product of their probabilities. To calculate all possible matches of a, b a comparison matrix M is constructed. Then, the mean score over a region is:

```
>DP00084
S-rich region from 103 to 160 corrected with score 46
MSDNDDIEVESDEEQPRFQSAADKRAHHNALERKRRDHIKDSFHSLRDSVPSLQGEKASR
AQILDKATEYIQYMRRKNHTHQQDIDDLKRQNALLEQQVRALEKARXXAQLQTNYPXXDN
XLYTNAKGXTIXAFDGGXDXXXEXEPEEPQXRKKLRMEAX
>DP00182
MAPTKRKGSCPGAAPKKPKEPVQVPKLVIKGGIEVLGVKTGVDSFTEVECFLNPQMGNPD
EHQKGLSKSLAAEKQFTDDSPDKEQLPCYSVARIPLPNINEDLTCGNILMWEAVTVKTEV
IGVTAMLNLHSGTQKTHENGAGKPIQGSNFHFFAVGGEPLELQGVLANYRTKYPAQTVTP
KNATVDSQQMNTDHKAVLDKDNAYPVECWVPDPSKNENTRYFGTYTGGENVPPVLHITNT
ATTVLLDEQGVGPLCKADSLYVSAVDICGLFTNTSGTQQWKGLPRYFKITLRKRSVKNPY
PISFLLSDLINRRTQRVDGQPMIGMSSQVEEVRVYEDTEELPGDPDMIRYIDEFGQTTTR
MQ
>DP00206
T-rich region from 66 to 156 corrected with score 48
MKAAQKGFTLIELMIVVAIIGILAAIAIPAYQDYTARAQLSERMTLASGLKTKVSDIFSQ
DGSCPANXAAXAGIEKDXDINGKYVAKVXXGGXAAASGGCXIVAXMKASDVAXPLRGKXL
XLXLGNADKGSYXWACXSNADNKYLPKXCQXAXXXXP
>DP00334
E-rich region from 111 to 429 corrected with score 106
S-rich region from 111 to 408 corrected with score 56
MCNTNMSVPTDGAVTTSQIPASEQETLVRPKPLLLKLLKSVGAQKDTYTMKEVLFYLGQY
IMTKRLYDEKQQHIVYCSNDLLGDLFGVPSFSVKEHRKIYTMIYRNLVVVNQQXXXDXGT
XVXXNRCHLXGGXDQKDLVQXLQXXKPXXXHLVXRPXTXXRRRAIXXTXXNXDXLXGXRQ
RKRHKXDXIXLXFDXXLALCVIRXICCXRXXXXXXTGTPXNPDLDAGVXXHXGDWLDQDX
VXDQFXVXFXVXXLDXXDYXLXXXGQXLXDXDDXVYQVTVYQAGXXDTDXFXXDPXIXLA
DYWKCTXCNXMNPPLPXHCNRCWALRXNWLPXDKGKDKGXIXXKAKLXNXTQAXXGFDVP
DCKKTIVNDXRXXCVXXNDDKITQAXQXQXXXDYXQPXTXXXIIYXXQXDVKXFXRXXTQ
DKXXSVXSSLPLNAIEPCVICQGRPKNGCIVHGKTGHLMACFTCAKKLKKRNKPCPVCRQ
PIQMIVLTYFP
>DP00359
MMLTKSVVISRPAVRPVSTRRAVVVRASGQPAVDLNKKVQDAVKEAEDACAKGTSADCAV
AWDTVEELSAAVSHKKDAVKADVTLTDPLEAFCKDAPDADECRVYED
>DP00518
P-rich region from 21 to 68 corrected with score 44
MAKQPSDVSSECDREGGQLQXAERXXQLRXGAXTSLQTEXQGNXDGEGDRCXHGSXQGXL
AXXASXGXFATRSPLFIFVRRSSLLSRSSSGYFSFDTDRSPAPMSCDKSTQTPSPPCQAF
NHYLSAMASIRQSQEEPEDLRPEIRIAQELRRIGDEFNETYTRRVFANDYREAEDHPQMV
ILQLLRFIFRLVWRRH
```

FIGURE 3.1: An example of CAST predictions.

$$\sum_{a,b} p_{ab} m_{a,b} = l \sum_{a} f_a \sum_{b} p_b m_{a,b} = \sum_{a} f_a C_a \tag{3.2}$$

where C_a is a parameter of the residue type a, $m_{a,b}$ are the elements of matrix M and $f_a \in [0,1], \sum_a f_a = 1$ are the frequencies of each residue type a. In other words, the above equation is the weighted score of all parameters C_a of the 20 amino acids. This is equivalent as searching across a database with 20 homopolymers in order to find the bias regions. To accelerate the searching computational complexity dynamic programming is adopted to model the above equations. Because the sequence locations of homopolymers are not important, the search technique based on the Smith-Waterman algorithm is simplified using only one iteration for each search. In order to find the bias for each residue type a of a sequence $r = [r_1, r_2, ..., r_n]$, the score at position i is computed as:

$$s_i^a = m_{a,r_i} + \begin{cases} s_{i-1}^a & \text{if } s_{i-1}^a \geq 0 \\ 0 & \text{if } s_{i-1}^a < 0 \end{cases} \tag{3.3}$$

The above procedure is applied for all types of compositional bias to detect all possible low complexity regions for each amino acid. Finally, CAST has an option to mask biased regions with the undefined residue type X that can be ignored in further database searches thus improving the specificity of the search strategy, i.e. reducing false positive hits.

3.1.2 SEG

SEG is a compositional bias masking algorithm, which segments amino acid sequences into subsequences of varying complexity. SEG adopts several metrics to calculate local complexity and extract compositional bias. For a subsequence or a window j of length L residues in a biopolymer with N kinds of residues, observable statistical features are specified on three different levels: complexity state, composition and sequence. The complexity state vectors S_j of a window with $N = [n_1, n_2, \ldots, n_N]$ residues, is a sorted vector, which has the following properties:

$$0 \geq n_i \geq L, \ \sum_{i=1}^{N} n_i = L \ \ n_i \geq n_{i+1} \tag{3.4}$$

```
>DP00206
                                1-17   MKAAQKGFTLIELMIVV
                   aiigilaaiaipa  18-30
                                31-85   YQDYTARAQLSERMTLASGLKTKVSDIFSQ
                                        DGSCPANTAATAGIEKDTDINGKYV
                akvttggtaaasggct  86-101
                               102-147  IVATMKASDVATPLRGKTLTLTLGNADKGS
                                        YTWACTSNADNKYLPK
                      tcqtatttt  148-156
                               157-157  P

>DP00334
                                1-29   MCNTNMSVPTDGAVTTSQIPASEQETLVR
                     pkplllkllk  30-39
                                40-201  SVGAQKDTYTMKEVLFYLGQYIMTKRLYDE
                                        KQQHIVYCSNDLLGDLFGVPSFSVKEHRKI
                                        YTMIYRNLVVVNQQESSDSGTSVSENRCHL
                                        EGGSDQKDLVQELQEEKPSSSHLVSRPSTS
                                        SRRRAISETEENSDELSGERQRKRHKSDSI
                                        SLSFDESLALCV
               ireiccersssses  202-215
                               216-251  TGTPSNPDLDAGVSEHSGDWLDQDSVSDQF
                                        SVEFEV
                esldsedyslsee  252-264
                               265-393  GQELSDEDDEVYQVTVYQAGESDTDSFEED
                                        PEISLADYWKCTSCNEMNPPLPSHCNRCWA
                                        LRENWLPEDKGKDKGEISEKAKLENSTQAE
                                        EGFDVPDCKKTIVNDSRESCVEENDDKITQ
                                        ASQSQESED
              ysqpstsssiiyssq  394-408
                               409-491  EDVKEFEREETQDKEESVESSLPLNAIEPC
                                        VICQGRPKNGCIVHGKTGHLMACFTCAKKL
                                        KKRNKPCPVCRQPIQMIVLTYFP
```

FIGURE 3.2: An example of SEG predictions.

These complexity states depend only on the numbers N, L and n_i and are independent of the residue composition and the states' likelihood of occurring. In addition, the composition measurement F for a complexity state is defined as:

$$F = \frac{N!}{\prod_{k=0}^{L} r_k!}, \tag{3.5}$$

where r_k are the counts of how many times each occurrence appears in the complexity state and are bound to the following restrictions.

$$0 \leq r_k \leq N, \;\; 0 \leq k \leq L, \;\; \sum_{k=0}^{L} r_k = N \tag{3.6}$$

Furthermore, the total number of all possible different sequences for each complexity state is defined as:

$$\Omega = \frac{L!}{\prod_{i=1}^{N} n_i!}, \tag{3.7}$$

while the total number for all complexity states is equal to N^L. The formal definitions of complexity (K_1) is based on the number of all possible different sequences (ω) and is given from:

$$K_1 = \frac{1}{L} \log \Omega, \tag{3.8}$$

while the formal definition of entropy (K_2) from information theory is:

$$K_2 = -\sum_{i=1}^{N} \frac{n_i}{L} \left(\log \frac{n_i}{L}\right) \tag{3.9}$$

Entropy practically describes the log probability of the complexity state. Complexity and entropy measures are comparable in nature and can both be used to describe the compositional bias of proteins.

3.1.3 IUPred

To distinguish between ordered and disordered regions in proteins, IUPred [32] uses a statistical interaction potential to predict the potential of polypeptides to form stabilizing interactions. The total interaction energies can be calculated using a quadratic expression in the amino acid composition, which accounts for the fact that an amino acid's contribution to order/disorder is dependent on its own chemical type and on its potential interaction partners. A 20×20 energy predictor matrix is used in the calculation, which is configured by a statistical method, in order to get an estimate of the expected

pairwise energy of known-structured globular proteins. When globular and disordered proteins are compared, a clear distinction in energy content is discovered. This distinction highlights that the lack of a well-defined three-dimensional structure is an intrinsic trait of certain proteins because no training on disordered proteins is involved. Furthermore, by evaluating the local sequential environment of residues within 2–100 residues in either direction, this methodology was developed into a position-specific strategy to predict protein disorder. After this, the score is smoothed across a window size of 21. A web server has also been created that accepts a single amino acid sequence as an input and calculates the pairwise energy profile along it. After that, the energy values are converted into a probabilistic score ranging from 0 (complete order) to 1 (complete disorder). Residues with a score greater than 0.5 are considered disordered. The prediction of long disorder, short disorder and structured domains is optional and requires different parameters for each category. The server's main profile is to forecast context-independent global disorder that spans at least 30 consecutive residues. In contrast, to predict short and context-dependent residues, i.e., missing residues in X-ray structure, only the sequential neighborhood of 25 residues is evaluated. Another potential application of this method is to find putative structured domains appropriate for structure determination. More specifically, the method analyses the energy profile and finds continuous regions that are confidently predicted to be sorted. Neighboring regions are combined, whereas sections that are less than 30 residues in length are omitted. The region(s) anticipated to correspond to structured domains are returned, when this prediction type is selected.

3.1.4 IUPred2A

IUPred2A [3] is an updated version of IUPred [32] that is combined with ANCHOR2 to create energy estimation-based predictions for ordered and disordered residues and disordered binding areas. The revised web server keeps the old programs' resilience while adding a few additional functions. The new version has fixed minor bugs of IUPred, while the next version of ANCHOR is vastly enhanced because of a new architecture that has been optimized on new datasets. IUPred2A employs an energy estimating approach by adopting a low-resolution statistical potential to quantify the propensity of amino acid pairs to establish interactions in globular protein structures. The statistical potential permits the estimation of the energy for each residue based on its interactions

with other contacting residues in the structure. The overall stabilizing energy contribution of intrachain interactions in a particular protein structure may be calculated using the sum of these residue-level energy components. A unique approach was devised to estimate these energy directly from the amino acid sequence without a known structure. The energy of each residue in the amino acid sequence is determined in this model using the formula:

$$e_i^k = \sum_{j=1}^{20} P_{ij} c_j^k, \tag{3.10}$$

where e_i^k is the energy of the type i residue in position k, P_{ij} is the ij-th element of the energy prediction matrix and c_j shows the j-th element of the amino acid composition vector. P is an energy prediction matrix that links the amino acid composition vector to the energy of a particular residue. Its parameters were tuned to minimize the discrepancy between the energy computed from known structures using the statistical potential and the energy estimated from the amino acid sequence. Residues with high energies are projected to be ordered, whereas those with low energies are predicted to be disordered, based on the above calculation. The energy calculated for each amino acid residue is smoothed using the window size (w_0) and translated into a score between 0 and 1, allowing them to be interpreted as quasi-probabilities of a certain residue being disordered.

3.1.5 MobiDB-lite

MobiDB-lite, integrates 8 cutting-edge IDR predictors into a consensus that has been carefully verified over the complete PDB X-ray dataset. IDR predictions for an input protein are created using three forms of ESpritz-D, ESpritz-N, ESpritz-X, IUPred-long, IUPred=short, DisEMBL-465, DisEMBL-HL and GlobPlot. To identify a residue as disordered, the raw ID predictions are integrated into a consensus using a threshold of at least 5 out of 8 techniques. To improve its robustness, MobiDB-lite incorporates a filtering phase that ensures that predicted areas include at least twenty consecutive residues. The procedure begins with an iterative filtering phase that removes short spans of up to

three disordered residues inside ordered sections, as well as ordered regions inside disordered ones. If two disordered areas of at least 20 residues flank structured sequences of up to ten consecutive residues, they are classified as disordered. MobiDB-lite is meant to be very specific in this approach, complementing traditional domain-centric annotation of protein sequences.

3.1.6 ESpritz

For the recurrent neural network input, ESpritz uses five Atchley sequence metrics as numerical sequence attributes [50]. Almost 500 different amino acid scales from the AAindex database [51] were clustered to create each scale. The scales were found to reflect polarity, secondary structure, molecule volume, codon diversity and electrostatic charge, suggesting that a more diverse amino acid representation is possible. Because the five scales have uneven ranges, they must be normalized before being used as neural network inputs. Normalization is carried out in the same way as before, with the squares of the scales adding up to 1:

$$\sum_{t=1}^{20} A_t(x)^2 = 1,\ (t = 1, \ldots, 5), \tag{3.11}$$

where x is one of the 20 amino acids and $A_t(x)$ is the sequence metric for x. For each sequence point k, it contains five inputs i to the neural network, each indicating a normalized Atchley scale [52]. The five inputs to this system are as follows if position k in the sequence includes amino acid x:

$$i_k^t = A_t(x),\ (t = 1, \ldots, 5), \tag{3.12}$$

SSpritz [12], on the other hand, considers the 20 amino acids in 'one-hot' encoding. It has 20 inputs i with each unit for sequence position k being assigned to one of the 20 amino acids.

$$i_k^{1-20} = R_k(x) \tag{3.13}$$

where $R_k(x)$ is an alphabetically sorted vector corresponding to the 20 amino acids (i.e., [A, C, D, E,..., W, Y]), where x represents the residue at position R_k^j. If the position amino acid is present in the sequence at position k, $R_k^j = 1$ otherwise, $R_k^j = 0$. The results of the above functions are merged to create the final method ESpritz. Multiple sequence alignments, which contain evolutionary information, are frequently employed to improve prediction performance. Sequence profiles are handled by combining the two sequence encodings. We denote as $p_k(x)$ the probability of finding amino acid x at position k along the sequence in a multiple sequence alignment. The profile-based predictor for the Atchley scales has six inputs i for each position k computed as:

$$i_k^t = \sum_{x \in C_k(x)} A_t(x) p_k(x), \quad (t = 1, \ldots, 5), \quad i_k^6 = \frac{g}{n + l} \tag{3.14}$$

where $C_k(x)$ is the set of amino acids for position k, g is the number of gaps, n is the number of non-gaps and l is the total number of sequences involved in the multiple sequence alignment. In the case of the 20 amino acids, the sequence profile $p_k(x)$ is multiplied by the input vector as:

$$i_k^{1-20} = \sum_{x \in C_k(x)} R_k(x) p_k(x) \tag{3.15}$$

3.1.7 GlobPlot

Linding *et al.* in [53], proposed the GlobPlot method that aims to find globular areas and intrinsic disorder in protein sequences. The main concept of GlobPlot is to use the propensities P of all amino acids. Then, given a propensity $P(a_i) \in \mathrm{R}$ for each amino acid a_i and the input protein sequence of length L, we calculate the following plot function:

$$\Omega(a_i) = \sum_{j=1}^{i=1} \Omega(a_j) + \ln(i + 1) P(a_i), \quad i = 1, \ldots, L \tag{3.16}$$

where ln is the natural algorithm. To smooth the curve Ω and obtain a numerical estimate of the first order derivative, a digital low-pass filter is used. Then, a basic peak finding method is used to select putative globular and disorder segments (referred to as PeakFinder). When the first derivative has positive (disorder) or negative (globular) values along a continuous stretch of the minimal length, the peaks are selected.

3.1.8 FlDPnn

Hu *et al.* in [7], propose FlDPnn method that uses a variety of input features to predict disordered regions accurately. First, multiple tools process the input sequence to generate relevant putative structural and functional information, which is then used to create the sequence profile. Putative disorder functions are produced by DisoRDPbind, DFLpred and fMoRFpred tools and are important since intrinsic disorder carries a multitude of cellular functions that are linked to various sequence patterns. Next, the input features are encoded by combining the profile data at three levels: residue, window and protein. Protein-level characteristics are also incorporated to express the overall bias of a particular sequence to be disordered or structured, in contrast to current disorder predictors that use only residue and window-level encodings. In addition, there is a function prediction module that emphasizes, aligns and enhances the predicted disordered regions. In the final step, a deep learning model is combined with random forest algorithm for function prediction.

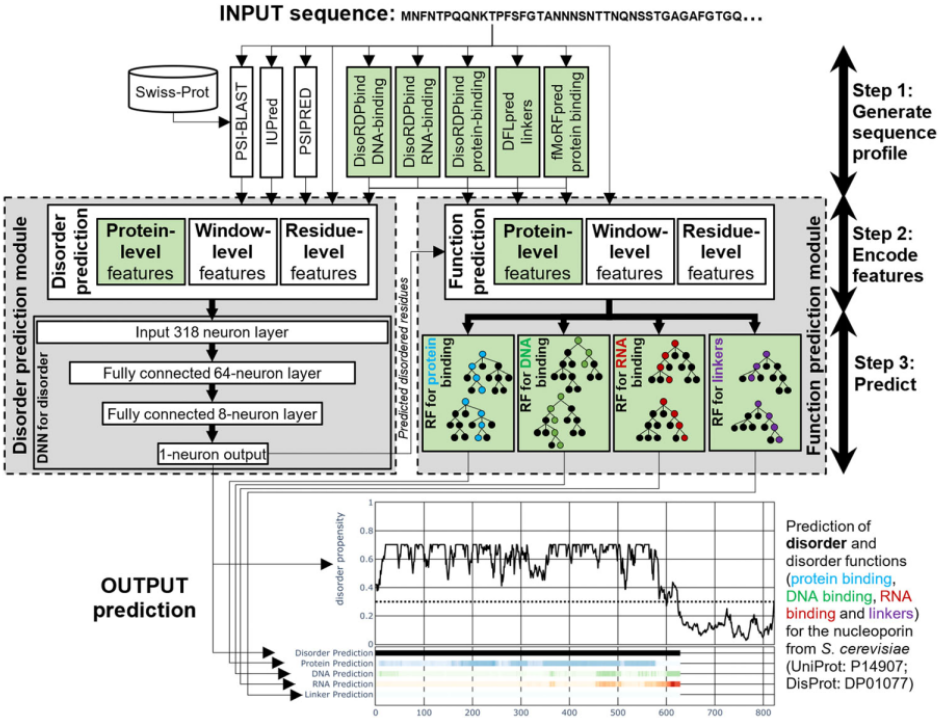

FIGURE 3.3: Overview of FlDPnn algorithm [7].

Chapter 4

Data and Methods

In this Chapter, we will initially describe our proposed method for predicting intrinsically disordered regions that is based on a Transformer network. Then, we analyze our framework that evaluates the predictions of several state-of-the-art IDP methods. Finally, we describe the implemented platform for simulating metagenomes, assembling proteins and identifying disordered regions for the first time.

4.1 Proposed method for IDP/IDR prediction

Previous research focused on algorithmic approaches or simple machine learning methods in order to predict intrinsic disorder. In this work, we adopt a Transformer, which has proven very successful in Natural Language Processing and Computer vision tasks and propose a new deep learning framework that trains a Transformer network to predict intrinsic disorder from amino acid sequences. At first, embedding layers learn informative representations of proteins at amino-acid level. These features are able to capture the biological function and structure. Then, the Transformer aims at capturing short- and long-term relationships between amino acids. More specifically, the Transformer employs the self-attention mechanism, which enables the network to construct very informative representations that include context from several regions in the sequence. Self-attention is able to find local and global interactions and identify disordered regions. Finally, a binary classifier is employed to predict IDRs.

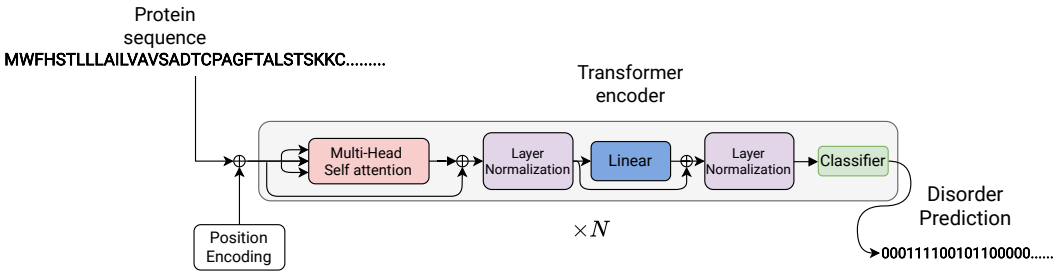

FIGURE 4.1: Overview of our proposed Transformer-based method that predicts disordered regions from proteins sequences.

4.1.1 Self-Attention mechanism

The attention function concept comes from information retrieval systems and is basically a transformation layer that maps an input sequence $X \in \mathbb{R}^{N \times d_x}$, where N is the sequence length and d_x the vector dimension, to three different vectors the query Q, the key K and the value V, respectively. These vectors are generated as:

$$Q = XW_q,\ K = XW_k,\ V = XW_v \tag{4.1}$$

where $W_q \in \mathbb{R}^{d_x \times d_q}$, $W_k \in \mathbb{R}^{d_x \times d_k}$, and $W_v \in \mathbb{R}^{d_x \times d_v}$ are 3 different weight matrices.

Having the query, value and key matrices, we can now apply the self-attention layer using the following function, namely the scaled dot-product attention:

$$Attention(Q, K, V) = \text{softmax}\,(\frac{QK^T}{\sqrt{d_k}})V. \tag{4.2}$$

The obtained attention weights are assigned to the elements of the value V and show in what information the layer attends. These attention weights show the correlations between the amino acids of a protein sequence as well as their importance on the final disorder classification.

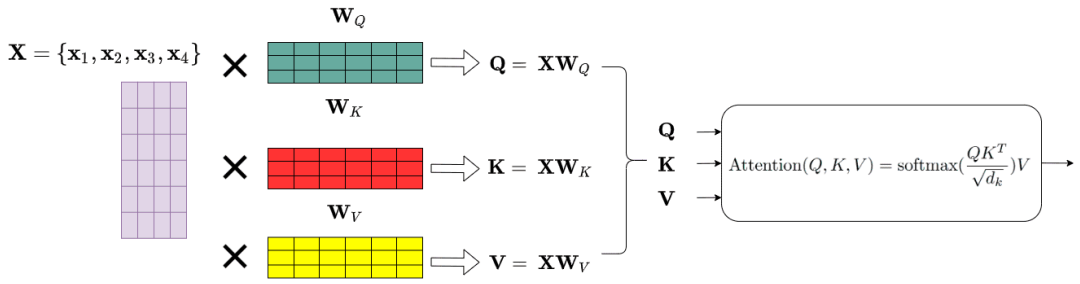

FIGURE 4.2: Overview of the Multi head attention mechanism [8].

4.1.2 Multi-Head Attention Mechanism

The modeling capabilities of a single self-attention block is coarse because to the limited feature subspace. To this end, Vaswani *et al.* [18] present a multi-head self-attention mechanism (MHSA) to address this problem, which projects the input into many subspaces, which are by parallel attention layers called heads. MHSA is computed as:

$$Q_i = XW_q^i, \quad K_i = XW_k^i, \quad V_i = XW_v^i \tag{4.3}$$

$$S_i = Attention(Q_i, K_i, V_i), \quad i = 1, 2, \ldots, h \tag{4.4}$$

$$MHSA(Q, K, V) = concat(S_1, S_2, \ldots S_h)W_o, \tag{4.5}$$

where h is the total number of heads, $W_o \in \mathbb{R}^{hd \times d_v}$ is the weight projection matrix, S_i is the attention matrix of each head and $W_q^i \in \mathbb{R}^{d \times d_v}$, $W_k^i \in \mathbb{R}^{d \times d_v}$, $W_v^i \in \mathbb{R}^{d \times d_v}$ are the weight matrices of each for the query, key and value of each head, respectively. Multi-head attention allows the model to pay attention to different elements of the sequence in different ways each time. This means that the model will be able to gather more positional data because each head will focus on various regions of the input and have a more comprehensive representation after the combination of the vectors.

4.1.3 Feed Forward Networks

Then, the output of MHSA is fed to a fully connected feed-forward network applied to each location individually and identically in each of the layers in our encoder as:

$$\mathrm{FFN}(X) = \mathrm{ReLU}(W_1 X + b_1)W_2 + b_2 \tag{4.6}$$

In addition, skip connections are used to tackle the vanishing gradient problem and give the ability to allow the representations of different levels of processing to interact. Layer Normalization (LN) is also applied on the outputs of the fully connected layer. In LN, the mean and variance are computed across channels and spatial dims.

4.1.4 Positional encoding

The Transformer model operates on the input embeddings without any prior knowledge of the sequence position, since it does not contain recurrent or convolutional layers. To this end, to make use of sequential information, positional vectors are commonly used and added to the input embeddings. One of the most common techniques is to use sine and cosine functions for different frequencies as:

$$\mathrm{PE}(y, i) = \begin{cases} \sin(y\omega_k) & \text{if } i = 2k \\ \cos(y\omega_k) & \text{if } i = 2k+1 \end{cases} \tag{4.7}$$

$$\omega_k = \frac{1}{10000^{2k/d}}, \quad k = 1, \ldots, d/2 \tag{4.8}$$

where i is the index and d the length of the positional vector and y is the current position.

4.1.5 Classifier

The classifier is a fully connected layer that maps the output representation H for each amino acid of the input sequence X into probability $\mathbf{y} \in \mathrm{R}^{N \times c}$, where c are the number of the output classes. In intrinsic disorder prediction, there are two output classes for ordered and disordered amino acids, respectively. The output probability sequence is extracted as:

$$\mathbf{y} = \text{softmax}\,(W_o H + b_o), \tag{4.9}$$

where $W_o \in \mathbb{R}^{d \times c}$ is the weight projection matrix of the classifier and $b_o \in \mathbb{R}^c$ is the bias that is added to the output.

4.1.6 Optimization strategy

4.1.6.1 Pre-training

The masked language modeling strategy is used to pretrain the Transformer model in a self-supervised manner. A portion of the amino acids in the sequence is deleted and replaced with a special mask token. Then, the network is optimized in order to generate the masked characters with the following objective function, which is the negative log probability of the real amino acid x_i given the masked sequence $x = M$ as context for each masked token:

$$L_{Mask} = \sum_{i \in M} -\log\, p(x_i | x_M), \tag{4.10}$$

where M is a set of token that will be masked, x is the input sequence and i is the index of the masked token. With this training strategy, the model learns to model dependencies between the masked and unmasked tokens of the input sequence. The following masking procedure is adopted, where at 80% of the iterations a token is replaced with the special mask token, at 10% of the iterations the token is replaced with a random token and at 10% of the iterations the token is not changed.

4.1.6.2 Supervised training for IDR

After the self-supervised training procedure, the model is trained in order to detect intrinsic disordered regions. More specifically, the model outputs probabilities for each class (0 for ordered and 1 for disordered amino acids) and is trained with cross entropy loss as:

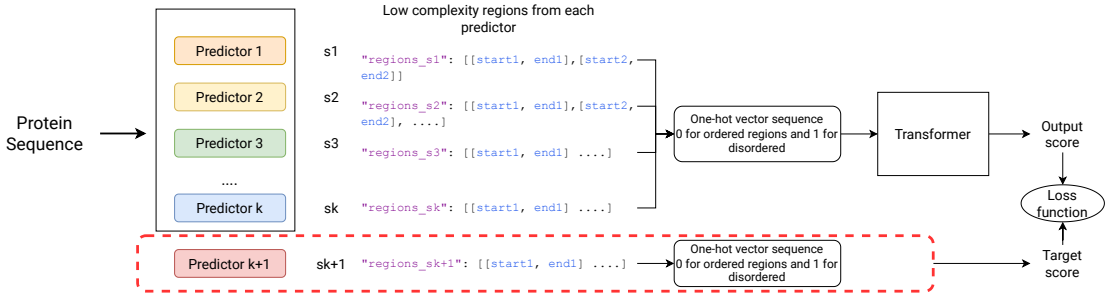

FIGURE 4.3: Proposed framework for IDP comparison.

$$L_{CE} = -\frac{1}{C}\sum_{i=1}^{C} y_i \log(\hat{y}_i), \tag{4.11}$$

where C is the number of classes ($C = 2$ in IDR task), y_i is the target class and $\hat{y}_i$ is the predicted class.

4.2 Framework for comprehensive comparison of intrinsic disorder prediction methods

In this section, we will describe the proposed framework in order to evaluate the performance and robustness of K different IDP predictors. Given a protein sequence $\mathbf{x} \in \mathbb{R}^N$, where N is the sequence length, each predictor k outputs the disordered regions $\mathbf{r} = [r_1, r_2, \dots r_j]$. The predicted regions are transformed into a one-hot vector sequence with 0 representing ordered amino acids and 1 representing disordered amino acids. By aggregating all the predictions, we construct a matrix $\mathbf{P} \in \mathbb{R}^{N \times K}$. This matrix is the transformed dataset, where the predictions of one method are the target data and the rest are used as input. Then, a deep neural network is used in order to combine the predictions of the first $K-1$ predictors and output a score for this input. More specifically, a shallow Transformer network with 2 layers is adopted and aims to predict the score of the last predictor. In this way, the Transformer will calculate the similarities between the input predictions and the scores from the test predictor and help us understand the advantages and disadvantages of each method. An overview of this framework is shown in Figure 4.3.

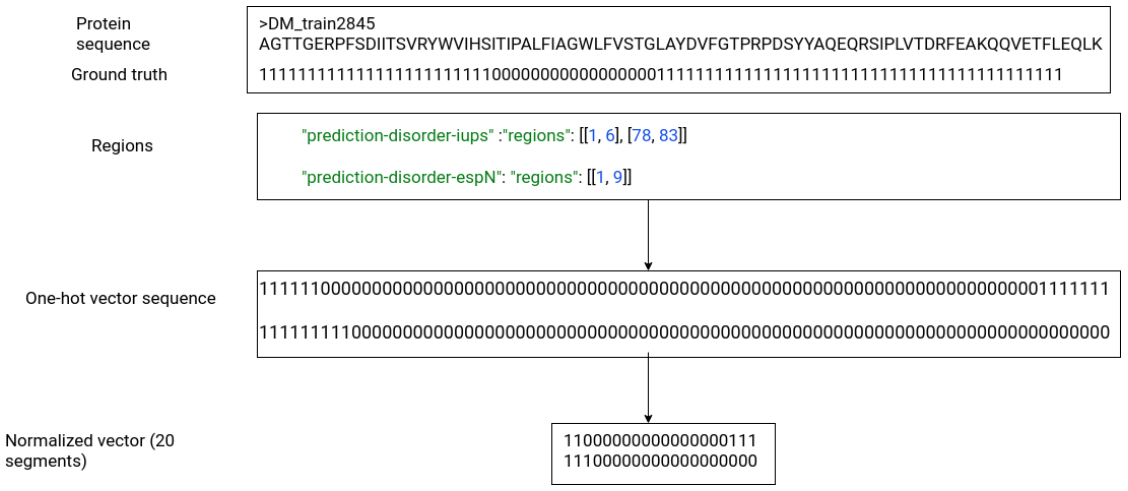

FIGURE 4.4: Data normalization applied on predictions.

The neural network is trained with cross entropy loss as:

$$L_{CE} = -\frac{1}{C}\sum_{i=1}^{C} y_i \log(\hat{y}_i), \tag{4.12}$$

where C is the number of classes ($C = 2$ in IDR task), y_i is the target score and $\hat{y}_i$ is the predicted score.

4.2.1 Data Discretization

Instead of using raw input sequences, we convert the predicted regions from predictors to compact vector representations. More specifically, we transform sequences to vectors of standard length (segments / length bins). Then, when a predictor predicts disorder for a specific bin (e.g., for 20 bins, predictor i hits 13, 14, predictor j hits 13, 14, 15 and predictor k hits 12, 14) we set this bin equal to 1 and the other bins equal to 0. So for that entry, we will have a table of 3 predictors $\times$ 20 with 7 pixels on. The process of data discretization is depicted in Figure 4.4.

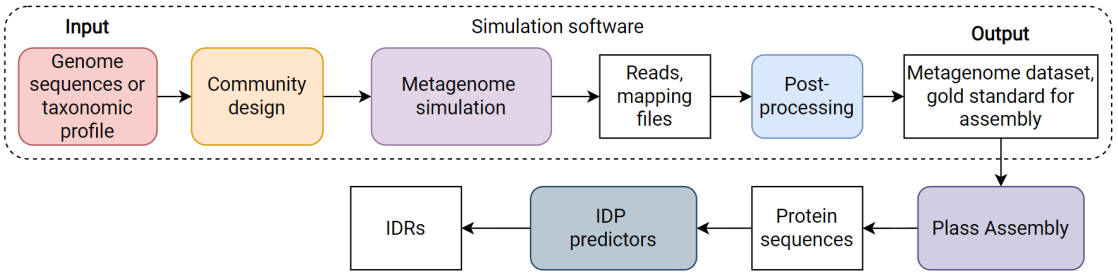

FIGURE 4.5: Framework for intrinsic disorder prediction on metagenomic data.

4.3 Intrinsic disorder prediction on metagenomes

In this section, we outline our framework that we will use to predict intrinsic disorder on metagenome datasets. The goal of this framework is to investigate the effect of various sequencing parameters on the intrinsic disorder content of the assembled proteins. For this purpose we adopt the simulation software CAMISIM [49] to simulate metagenomes and generate the dataset. Then, a protein assembler is used to process the simulated dataset and generate the output proteins. Finally, the aforementioned IDP predictors will be used to predict IDRs. The workflow of this process is shown in Figure 4.5.

4.3.1 Simulation of metagenomic data

In the first stage of the simulation process, we define the ”community design” which is the *de novo* design in our method and the simulation parameters that are shown in Figure 4.6. The input genome sequence and a mapping file that specifies the taxonomy, the novelty category and the Operational Taxonomic Unit (OTU), are defined. Then we assign abundances to the community genomes by specifying the sample's type and the number of samples n. Multi-sample data sets are also widely adopted in actual sequencing investigations, in addition to single samples. The following parameters for multi-samples are also defined by the user.

1. For single sample experiments, the relative abundances are chosen from a log-normal distribution with a mean equal to 1 and the standard deviation equal to 2.

```
dataset_id=RL
output_directory=out
temp_directory=/tmp
gsa=True
pooled_gsa=True
anonymous=True
compress=1

[ReadSimulator]
readsim=tools/art_illumina-2.3.6/art_illumina
error_profiles=tools/art_illumina-2.3.6/profiles
samtools=tools/samtools-1.3/samtools
profile=mbarc
size=0.1
type=art
fragments_size_mean=270
fragment_size_standard_deviation=27

[CommunityDesign]
#distribution_file_paths=out/abundance0.tsv,out/abundance1.tsv,out/abundance2.tsv,out,
ncbi_taxdump=tools/ncbi-taxonomy_20170222.tar.gz
strain_simulation_template=scripts/StrainSimulationWrapper/sgEvolver/simulation_dir
number_of_samples=10

[community0]
metadata=defaults/metadata.tsv
id_to_genome_file=defaults/genome_to_id.tsv
id_to_gff_file=
genomes_total=24
genomes_real=24
max_strains_per_otu=1
ratio=1
mode=differential
log_mu=1
log_sigma=2
gauss_mu=1
gauss_sigma=1
```

FIGURE 4.6: Simulation parameters of CAMISIM.

2. The second option is the differential abundance mode, which simulates a population that has been sampled several times after environmental circumstances or DNA extraction procedures have been changed. This setup creates n different log-normal distributions for the relative abundances.

3. The replicates mode may be used to produce metagenome data sets with many

samples with highly comparable genomic abundance distributions. More specifically, n samples are generated from an initial log-normal distribution D_0 by adding Gaussian noise.

4. The last option is to generate time series metagenome data sets with numerous linked samples. For this, a Markov model-like simulation is used, with each of the n samples' distribution determined by the distribution of the preceding sample plus an extra log-normal or Gaussian component. This mimics the natural variability of species richness over time and guarantees that the abundance modifications to the previously measured metagenome do not become too high.

The genomic abundance profiles from the community design process are used to create metagenome data sets. The number of produced reads n_t is determined by the genome size s_t and the total number of reads n in the sample for each genome-specific taxon t and its abundance $(t, abt) \in P_{out}$. The total number of reads n is calculated by dividing the whole sequence size of the sample by the mean length of the read of the sequencing method used as:

$$n_t = n \frac{ab_t s_t}{\sum_{i \in P_{out}} ab_i s_i} \tag{4.13}$$

ART [54] is used by default to generate Illumina 2×150 bp paired-end reads with a HiSeq 2500 error profile. MBARC-26 [55], a specified fake community that has already been used to benchmark bioinformatics software and a full-length 16S rRNA gene amplicon sequencing methodology, was utilized to train the profile.

4.3.2 Protein assembly

To assemble protein sequences from reads, we will use the Plass tool [9]. Plass employs a graph-free, greedy iterative assembly technique that grows linearly in time and memory. At first, Plass translates all putative open reading frames (ORFs), i.e., sections of a DNA molecule that have no stop codons when translated into amino acids, containing 45 codons into protein sequences by merging overlapping read pairs. Then, it chooses the m k-mers with the lowest hash values for each of these N sequences and creates

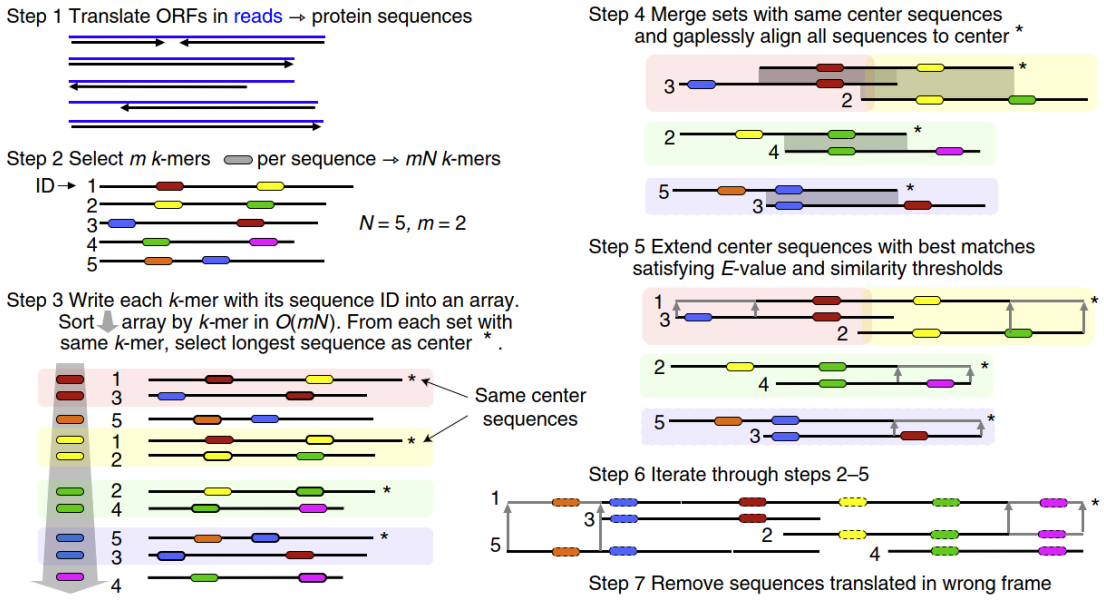

FIGURE 4.7: Workflow of Plass assembler [9].

an array containing the mN k-mers and the sequence identifiers. The default values for these experiments are $m = 60, k = 14$ and the alphabet size is reduced to 13. Then, we sort the array by the k-mer value to locate the sequences that contain each k-mer and each sequence is aligned at the center of the set. The resulting array is divided into groups based on the center sequence, and the center sequence is aligned to each group member (mN alignments) without gaps. Sequences having an E-value greater than 105 should be removed from the group. In the next step, each center sequence is extended repeatedly by the group member with the highest sequence similarity until no further extensions are available with a default minimum similarity threshold of 90%. The above procedure is iterated for 12 times by default. Finally, after the iterations, a neural network deletes sequences that were translated in the wrong frame. The workflow of this method is shown in Figure 4.7. The output proteins are then fed to the IDP predictor mentioned in Section 3 to predict the ratio of the intrinsically disordered regions.

Chapter 5

Experimental results

In this section, we will evaluate our proposed method for IDR prediction on several datasets. Then, we will use our proposed framework to compare several IDP methods and validate their predictions. Finally, we will run IDP methods on simulated metagenomic data with different simulation parameters.

5.1 Intrinsic disorder prediction

5.1.1 Metrics

The following metrics are used to evaluate the intrinsic disorder prediction performance of the implemented methods and at first we will define the fundamentals of those metrics. Subsequently, a true positive (TP) is when the model predicts the positive class properly, while a true negative (TN) is a result in which the model predicts the negative class. On the other hand, a false positive (FP) occurs when the model outputs the positive class inaccurately and a false negative (FN) is when the model inaccurately predicts the negative class. In the experiments we define the ordered amino acids as positive class the (i.e., denoted as 1) and the disordered amino acids as the negative class (i.e, denoted as 0).

TABLE 5.1: Information of datasets

Dataset	Residue level		Protein level	
	Disordered	Ordered	LDRs	SDRs
Train	74170 (10,1%)	656634 (89.9%)	342 (11.4%)	2658 (88.6%)
Validation	29082 (9.5%)	276748 (90,5%)	144 (11.7%)	1085 (88.3%)
MXD494	44087 (22.4%)	152414 (77.6%)	246 (49.8%)	248 (50,2%)
DISORDER723	13526 (6.3%)	201703 (93.7%)	56 (7.7%)	667 (92.3%)

Sensitivity (SN) or true positive ratio (TPR) is the fraction of samples that are actually positive and the model correctly classifies them. TPR is defined as:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{5.1}$$

On the other hand, specificity (SP) or true negative ration (TNR) is the fraction of samples that are actually negative and the model correctly classifies them. TNR is defined as:

$$\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \tag{5.2}$$

Balanced accuracy metric is used to evaluate binary classifiers and is suitable for tasks with imbalanced number of samples for each class such as intrinsic disorder prediction.

$$\mathrm{BACC} = \frac{\mathrm{TPR} + \mathrm{TNR}}{2} \tag{5.3}$$

The Matthews Correlation Coefficient (MCC) measures the quality of the confusion matrix of binary classifiers and is defined as:

$$\mathrm{MCC} = \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}} \tag{5.4}$$

5.1.2 Datasets

The main dataset (namely as DM3000 for the training set and DM1229 for the test set) used in this study was constructed by Zhang *et al.* [37]. It comprises 4229 protein sequences with a sequence similarity of less than 25% between any two proteins. The dataset is split into two sections: a training dataset with 3000 proteins and a validation dataset with 1229 proteins. The training dataset and validation dataset are further separated into LDR training dataset, SDR training dataset, LDR validation dataset and SDR validation dataset based on the different types of IDRs. Proteins in LDR datasets must have at least one LDR, whereas proteins in SDR datasets must only have SDRs. In this work, we adopt only the combined training and test sets with LDRs and SDRs. Then, the performance of the implemented models was evaluated on three widely used independent test datasets, MXD494 [56], Disorder723 [57] and DisProt [58]. These datasets contain various various ratios of Long Disordered Regions (LDRs) and Short Disordered Regions (SDRs). The main statistical information and characteristics about the datasets are shown in Table 5.1.

5.1.3 Implementation details

Two Transformer networks are used: ESM1-t6-43M, which has 6 Transformer layers with 43 million parameters and ESM1-t12-85M, which has 12 Transformer layers with 85 million parameters. These networks have been pretrained on 86 billion amino acids across 250 million protein sequences from UniRef50 database [26] and their weights are available in [16]. The pretrained models are converted for IDP by using a new classifier that will be trained to predict intrinsic disordered regions. The input layer is an embedding layer with input dimension equal to 35 (equal to total number of amino acids and special tokens ($< pad >, < mask >, < start >, < end > \dots$). Dropout layer are used with a dropout rate of $0,1$. All models were trained with Adam optimizer [68] with a constant learning rate $lr = 1e^{-4}$. The model implementation was based on PyTorch library [69].

5.1.4 Evaluation

In the first set of experiments, the proposed method will be compared against a set of several state-of-the-art methods. Table 5.2 presents the evaluation on the MXD494 dataset. FIDPnn outperforms all methods with $BACC$ and MCC scores of $0,766$ and $0,548$, respectively. This can be explained from the fact that this method uses structural and functional information. Our proposed method with the two different configurations manage to achieve comparable results with ESM1-t6-43M achieving an MCC score of 0,460 and AUC of 0,756 and ESM1-t12-85M achieving an MCC score of 0,467 and AUC

Table 5.2: Comparison of state-of-the-art methods on the MXD494 test dataset

Method	SN	SP	BACC	MCC	AUC
PROFbval [59]	0,835	0,387	0,611	0,196	0,697
Ucon [60]	0,554	0,787	0,671	0,313	0,741
DISpro [57]	0,303	0,940	0,622	0,318	0,775
NORSnet	0,532	0,829	0,681	0,347	0,738
RONN [61]	0,664	0,754	0,709	0,368	0,764
IUPred-short [32]	0,522	0,866	0,694	0,389	0,781
VSL2B [62]	0,774	0,698	0,736	0,401	0,793
IUPred-long [32]	0,581	0,841	0,711	0,405	0,784
DISOPRED2 [63]	0,647	0,800	0,724	0,406	0,781
DISOPRED3 [13]	0,622	0,820	0,721	0,410	0,800
SPINE-D [37]	0,787	0,698	0,742	0,411	0,803
AUCpreD [38]	0,521	0,881	0,701	0,411	0,800
PONDER-FIT [64]	0,631	0,821	0,726	0,419	0,790
RFPR-IDP [65]	0,749	0,758	0,754	0,442	0,821
MD [66]	0,673	0,813	0,743	0,444	0,821
MFDp [67]	0,746	0,768	0,757	0,451	0,821
SPOT-Disorder [14]	0,626	0,851	0,739	0,457	0,813
IDP-Seq2Seq [39]	0,743	0,791	**0,767**	**0,475**	**0,825**
CAST	0,128	0,968	0,548	0,177	-
SEG	0,194	0,939	0,566	0,193	-
IUPred2A-long	0,572	0,845	0,708	0,402	0,785
IUPred2A-short	0,521	0,867	0,694	0,390	0,782
flDPnn	0,645	0,548	0,620	0,912	0,766
ESM1-t6-43M	0,680	0,821	0,750	0,460	0,756
ESM1-t12-85M	0,633	0,854	0,743	0,467	0,749

Table 5.3: Comparison of state-of-the-art methods on the Disorder723 dataset

Method	SN	SP	BACC	MCC	AUC
IUPred-long [32]	0,298	0,949	0,623	0,247	0,721
SPINE-D [37]	0,779	0,840	0,810	0,376	0,891
IUPred-Short [32]	0,495	0,943	0,719	0,382	0,810
IDP-Seq2Seq[39]	0,618	0,955	**0,787**	0,511	**0,906**
RFPR-IDP [65]	0,522	0,974	0,748	0,517	0,898
SPOT-Disorder [14]	0,470	0,983	0,726	0,531	0,898
DISOPRED3 [13]	0,452	0,986	0,719	0,536	0,899
AUCpreD [38]	**0,580**	0,974	0,777	**0,564**	0,914
CAST	0,031	**0,997**	0,514	0,101	-
SEG	0,093	0,975	0,534	0,097	-
IUPred2A-long	0,292	0,950	0,621	0,237	0,719
IUPred2A-short	0,494	0,945	0,720	0,387	0,801
ESM1-t6-43M	0,553	0,972	0,761	0,534	0,763
ESM1-t12-85M	0,565	0,973	0,770	0,548	0,769

of 0,749, respectively. IUPred2A-short and IUPred2A-long have a comparable performance. CAST and SEG do not achieve a good performance, which is explained by the fact that these methods are constructed for compositional bias detection and masking. Compositional bias is an orthogonal property to protein disorder, despite the fact that there is significant overlap between these two properties.

In Table 5.3, a comparison of our method against stat-of-the-art methods on the Disorder723 dataset, is conducted. On this dataset ESM1-t6-43M achieves an MCC score of 0,534 and AUC of 0,763 and ESM1-t12-85M achieves an MCC score of 0,548 and AUC of 0,769, respectively. Our methods perform comparable with the rest methods with slightly lower balanced accuracy. On the Disorder723 dataset, the best performing setup is AUCpred compared to the other methods achieving a MCC score of 0,564 demonstrating that protein representation contain useful information for IDP. Again, CAST and SEG perform worse since they are computational methods and not designed for intrinsic disorder prediction.

Finally, in Table 5.4, our methods are tested against several state-of-the-art methods on the Critical Assessment of protein Intrinsic Disorder prediction (CAID) dataset on proteins from the DisProt dataset. From the experimental results, it shown that the best

Table 5.4: Comparison of state-of-the-art methods on the CAID-2018 dataset

	BACC	F1-S	FPR	MCC	PPV	TPR	TNR
VSL2B [62]	0,684	0,408	0,341	0,277	0,286	0,709	0,659
IUPred2A-long [3]	0,685	0,416	0,299	0,285	0,302	0,670	0,701
IUPred-long [32]	0,686	0,418	0,294	0,287	0,305	0,666	0,706
IsUnstruct [70]	0,689	0,418	0,311	0,288	0,300	0,688	0,689
ESpritz-X [35]	0,689	0,418	0,309	0,288	0,301	0,686	0,691
IUPred-short [32]	0,688	0,418	0,304	0,288	0,302	0,679	0,696
MobiDB-lite [36]	0,688	0,420	0,296	0,289	0,305	0,673	0,704
IUPred2A-short [3]	0,688	0,420	0,297	0,290	0,305	0,674	0,703
DisoMine	0,698	0,424	0,326	0,299	0,300	0,721	0,674
AUCpreD-np	0,699	0,424	0,327	0,301	0,300	0,725	0,673
Predisorder	0,691	0,435	0,280	0,301	0,324	0,661	0,720
ESpritz-D [35]	0,703	0,428	0,325	0,307	0,303	0,731	0,675
SPOT-Disorder-Single	0,710	0,432	0,341	0,315	0,302	0,760	0,659
AUCpreD [38]	0,712	0,433	**0,376**	0,318	0,297	0,801	0,624
RawMSA [71]	0,714	0,445	0,297	0,328	0,321	0,725	0,703
fIDPlr	0,693	0,452	0,184	0,330	0,374	0,570	**0,816**
SPOT-Disorder1 [14]	0,723	0,434	0,386	0,330	0,294	**0,832**	0,614
SPOT-Disorder2 [14]	**0,725**	0,469	0,343	0,349	0,333	0,794	0,657
fIDPnn [7]	0,720	**0,483**	0,189	**0,370**	**0,392**	0,629	0,811
ESM1-t6-43M	0,713	0,459	0,340	0,360	0,352	0,662	0,763
ESM1-t12-85M	0,715	0,460	0,341	0,365	0,362	0,691	**0,772**

methods use deep learning techniques and notably outperform physiochemical methods. Across the different performance measures, the methods SPOT-Disorder2, fIDPnn, RawMSA and AUCpreD and our methods ESM1-t6-43M and ESM1-t12-85M are found among the top predictors. Overall, the Transformer network has high prediction performance, which proves that it is suitable for intrinsic disorder prediction tasks. Transfer learning from large scale protein databases is beneficial and helps methods in identifying disordered regions.

5.1.5 Comparison of IDP methods

In this subsection, we will describe a comprehensive comparison of several IDP predictors to estimate the similarity of their outputs and test if they produce similar or dissimilar intrinsic disordered regions. We tested the predictors in two datasets, MXD494

and Disorder723, respectively.

In Table 5.5, a comparison of IDP predictors is conducted with raw input data on the MXD494 dataset. From the experimental results, it shown that the outputs of better performing methods are more predictable than predictors with lower performance. More specifically, the IDRs of inferior methods do not match with the predicted IDRs of state-of-the-art methods. IUpred-short and ESpritz-Xray have the best performance with MCC scores of 0,769 and 0,741, respectively and their predicted regions match better with the regions of the other methods. In contrast, it is hard to predict the regions of SEG since its predictions differ significantly from other methods.

In Table 5.6, we compare the IDP predictors using the transformed data after discretization of the MXD494 dataset. In more detail, the scores of the IDP predictors are discretized to a standard length and then are used as input for the test framework. We observe a significant increase on the prediction performance due to the normalization on segments with standard length. This demonstrates that some methods tend to predict neighbouring regions as disordered and explains the high relative increase of the performance especially on inferior methods such as SEG, CAST and GlobPlot.

Then, in Table 5.7, a comparison of IDP predictors is conducted with raw input data on the Disorder723 dataset. Again, from the results, it shown that the outputs of better performing methods are more predictable than predictors with lower performance. IUpred-short and ESpritz-Xray have the best predictions about the disordered regions, which match better with the regions of the other methods. In contrast, it is hard to predict the regions of SEG since its predictions differ significantly from other methods.

Table 5.5: Results on comparison of IDP methods with unnormalized data on MXD494.

Test Predictor	**F1-score**	**MCC**	**AUC**
ESpritz Xray	0,747	0,741	0,834
ESpritz NMR	0,559	0,544	0,711
ESpritz Disprot	0,395	0,427	0,823
IUPred long	0,694	0,687	0,808
IUPred short	0,777	0,769	0,856
Globplot	0,392	0,380	0,629
SEG	0,142	0,250	0,539
CAST	0,389	0,476	0,623

Table 5.6: Results on comparison of IDP methods with discretized data on MXD494.

Test Predictor	F1-score	MCC	AUC
ESpritz Xray	0,929	0,905	0,950
ESpritz NMR	0,865	0,774	0,883
ESpritz Disprot	0,865	0,836	0,917
IUPred long	0,926	0,894	0,943
IUPred short	0,930	0,901	0,949
Globplot	0,750	0,604	0,798
SEG	0,768	0,737	0,840
CAST	0,845	0,818	0,891

Table 5.7: Results on comparison of IDP methods with raw data on Disorder723

Test Predictor	F1-score	MCC	AUC
ESpritz Xray	0,821	0,809	0,877
ESpritz NMR	0,692	0,662	0,783
ESpritz Disprot	0,845	0,838	0,937
IUPred long	0,826	0,813	0,883
IUPred short	0,852	0,839	0,908
Globplot	0,406	0,416	0,658
SEG	0,416	0,457	0,640
CAST	0,437	0,467	0,657

Then, experiments are conducted after normalization in Table 5.8. It is shown that most methods predict disordered regions in the same segments since the output score of the framework improves significantly. CAST shows a large improvement with an increase on MCC from 0,467 to 0,793. Overall, it is shown that IDP methods with robust performance are easily predictable and their output regions overlap and can be extracted from combinations of the other methods.

5.1.6 Intrinsic disorder prediction on metagenomic data

In this section we evaluate several IDP methods on assembled proteins from metagenomic data. We simulate metagenomic data using different parameters for simulation. In the first experiment, we run simulations using different number of samples to test the disordered regions on multi-sample co-assembly of proteins. In Table 5.9, it is shown

Table 5.8: Results on comparison of IDP methods with normalized data on Disorder723

Test Predictor	**F1-score**	**MCC**	**AUC**
ESpritz Xray	0,891	0,878	0,925
ESpritz NMR	0,748	0,681	0,819
ESpritz Disprot	0,869	0,858	0,916
IUPred long	0,787	0,764	0,851
IUPred short	0,909	0,894	0,940
Globplot	0,633	0,523	0,738
SEG	0,535	0,551	0,697
CAST	0,791	0,793	0,863

Table 5.9: Evaluation of IDP methods on assembled proteins from simulations with different number of samples

	Simulation assembly 10 samples		Simulation assembly 2 samples	
	101041 proteins		31942 proteins	
	IDRs	Fraction (%)	IDRs	Fraction (%)
IUPred long	382843	0,138	99788	0,133
IUPred short	449999	0,190	120066	0,187
ESpritz NMR	431670	0,228	110500	0,219
ESpritz Xray	264879	0,208	73212	0,206
ESpritz Disprot	95001	0,450	29143	0,478
Globplot	835903	0,194	225478	0,192
SEG	109416	0,066	30405	0,068
CAST	15396	0,009	4335	0,01

that the major difference when using more samples for the simulation, the Plass assembler finds more protein sequences. Furthermore, the IDP predictors predict the same ratio of disordered regions.

In the following experiment, we change the used read simulator error profile to observe if there is any difference on the percentage of the intrinsically disordered proteins. We run simulations with different error profiles and did not observe any significant change on the number of IDRs.

Table 5.10: Intrinsic disorder prediction on simulated metagenomic data with different read profile errors during simulation

	Mi		**Hi**		**Hi-150**	
	30160 proteins		31942 proteins		31942 proteins	
	IDRs	Fraction (%)	IDRs	Fraction (%)	IDRs	Fraction (%)
IUPred long	79177	0,143	90007	0,137	99709	0,134
IUPred short	94224	0,198	104157	0,187	119949	0,187
ESpritz NMR	85247	0,230	96308	0,220	110275	0,220
ESpritz Xray	57864	0,218	61971	0,204	73174	0,206
ESpritz Disprot	24.112	0,507	23708	0,462	29255	0,480
Globplot	175.525	0,199	191722	0,191	225257	0,192
SEG	24.343	0,072	27509	0,071	30346	0,069
CAST	3.512	0,010	3944	0,01	4294	0,009

Chapter 6

Conclusions & Future Work

In this work, we addressed the problem of recognizing intrinsic disordered proteins using deep learning methods as well as evaluating current state-of-the-art methods. We proposed a new method for predicting disordered regions from protein sequences by using Transformer networks. Moreover, the proposed method used self-supervised learning on large-scale datasets to learn prior biological information and to achieve a robust performance on predicting intrinsic disorder. The experimental results on three datasets demonstrated its effectiveness. Then, we compared several IDP predictors to compare their performance using a novel systematic framework, which computes their similarities and the agreement of their predictions. Finally, we tested IDP methods on a simulated metagenomic dataset using different configuration parameters.

A few different experiments and methods modifications have been left for future work. We are going to mention some ideas, that could be useful for obtaining a deeper analysis especially on metagenomic data. One extension of our proposed method would be to employ evolutionary information or combine other methods with the Transformer to improve prediction performance. Several machine learning methods could be used to evaluate the predictions of IDP predictors instead of a recurrent network, such as support vector machines. In addition, a fusion technique could also be adopted to combine several IDP methods and improve the overall prediction accuracy. Ultimately, better frameworks for the systematic evaluation of the wide range of available algorithms and

their application to genome and metagenome data will yield new insights into the detection and interpretation of intrinsic protein disorder.

Bibliography

[1] "Intrinsically disordered proteins." `https://weisgroup.ku.edu/intrinsically-disordered-proteins`, 2021. Accessed: 2021-08-12.

[2] P. M. Harrison, "flps 2.0: rapid annotation of compositionally-biased regions in biological sequences," *PeerJ*, vol. 9, p. e12363, 2021.

[3] B. Mészáros, G. Erdős, and Z. Dosztányi, "Iupred2a: context-dependent prediction of protein disorder as a function of redox state and protein binding," *Nucleic acids research*, vol. 46, no. W1, pp. W329–W337, 2018.

[4] R. Linding, L. J. Jensen, F. Diella, P. Bork, T. J. Gibson, and R. B. Russell, "Protein disorder prediction: implications for structural proteomics," *Structure*, vol. 11, no. 11, pp. 1453–1459, 2003.

[5] J. Hanson, K. K. Paliwal, T. Litfin, and Y. Zhou, "Spot-disorder2: improved protein intrinsic disorder prediction by ensembled deep learning," *Genomics, proteomics & bioinformatics*, vol. 17, no. 6, pp. 645–656, 2019.

[6] M. Escalona, S. Rocha, and D. Posada, "A comparison of tools for the simulation of genomic next-generation sequencing data," *Nature Reviews Genetics*, vol. 17, no. 8, pp. 459–469, 2016.

[7] G. Hu, A. Katuwawala, K. Wang, Z. Wu, S. Ghadermarzi, J. Gao, and L. Kurgan, "fldpnn: Accurate intrinsic disorder prediction with putative propensities of disorder functions," *Nature Communications*, vol. 12, no. 1, pp. 1–8, 2021.

[8] N. Adaloglou, "Transformers in computer vision," *https://theaisummer.com/*, 2021.

[9] M. Steinegger, M. Mirdita, and J. Söding, "Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold," *Nature methods*, vol. 16, no. 7, pp. 603–606, 2019.

[10] A. K. Dunker, M. M. Babu, E. Barbar, M. Blackledge, S. E. Bondos, Z. Dosztányi, H. J. Dyson, J. Forman-Kay, M. Fuxreiter, J. Gsponer, *et al.*, "What's in a name? why these proteins are intrinsically disordered: Why these proteins are intrinsically disordered," *Intrinsically disordered proteins*, vol. 1, no. 1, p. e24157, 2013.

[11] C. J. Oldfield, V. N. Uversky, A. K. Dunker, and L. Kurgan, "Introduction to intrinsically disordered proteins and regions," in *Intrinsically Disordered Proteins*, pp. 1–34, Elsevier, 2019.

[12] A. Vullo, O. Bortolami, G. Pollastri, and S. C. Tosatto, "Spritz: a server for the prediction of intrinsically disordered regions in protein sequences using kernel machines," *Nucleic acids research*, vol. 34, no. suppl_2, pp. W164–W168, 2006.

[13] D. T. Jones and D. Cozzetto, "Disopred3: precise disordered region predictions with annotated protein-binding activity," *Bioinformatics*, vol. 31, no. 6, pp. 857–863, 2015.

[14] J. Hanson, Y. Yang, K. Paliwal, and Y. Zhou, "Improving protein disorder prediction by deep bidirectional long short-term memory recurrent neural networks," *Bioinformatics*, vol. 33, no. 5, pp. 685–692, 2017.

[15] W. Wardah, M. G. Khan, A. Sharma, and M. A. Rashid, "Protein secondary structure prediction using neural networks and deep learning: A review," *Computational biology and chemistry*, vol. 81, pp. 1–8, 2019.

[16] A. Rives, J. Meier, T. Sercu, S. Goyal, Z. Lin, J. Liu, D. Guo, M. Ott, C. L. Zitnick, J. Ma, *et al.*, "Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences," *Proceedings of the National Academy of Sciences*, vol. 118, no. 15, 2021.

[17] A. Elnaggar, M. Heinzinger, C. Dallago, G. Rihawi, Y. Wang, L. Jones, T. Gibbs, T. Feher, C. Angerer, M. Steinegger, *et al.*, "Prottrans: towards cracking the language of life's code through self-supervised deep learning and high performance computing," *arXiv preprint arXiv:2007.06225*, 2020.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[20] I. Papastratis, K. Dimitropoulos, and P. Daras, "Continuous sign language recognition through a context-aware generative adversarial network," *Sensors*, vol. 21, no. 7, p. 2437, 2021.

[21] M. Heinzinger, A. Elnaggar, Y. Wang, C. Dallago, D. Nechaev, F. Matthes, and B. Rost, "Modeling aspects of the language of life through transfer-learning protein sequences," *BMC bioinformatics*, vol. 20, no. 1, p. 723, 2019.

[22] E. C. Alley, G. Khimulya, S. Biswas, M. AlQuraishi, and G. M. Church, "Unified rational protein engineering with sequence-based deep representation learning," *Nature methods*, vol. 16, no. 12, pp. 1315–1322, 2019.

[23] R. Rao, N. Bhattacharya, N. Thomas, Y. Duan, X. Chen, J. Canny, P. Abbeel, and Y. S. Song, "Evaluating protein transfer learning with tape," *Advances in neural information processing systems*, vol. 32, p. 9689, 2019.

[24] T. Bepler and B. Berger, "Learning protein sequence embeddings using information from structure," in *International Conference on Learning Representations*, 2018.

[25] M. Gardner, J. Grus, M. Neumann, O. Tafjord, P. Dasigi, N. Liu, M. Peters, M. Schmitz, and L. Zettlemoyer, "Allennlp: A deep semantic natural language processing platform," *arXiv preprint arXiv:1803.07640*, 2018.

[26] B. E. Suzek, Y. Wang, H. Huang, P. B. McGarvey, C. H. Wu, and U. Consortium, "Uniref clusters: a comprehensive and scalable alternative for improving sequence similarity searches," *Bioinformatics*, vol. 31, no. 6, pp. 926–932, 2015.

[27] Y. Liu, X. Wang, and B. Liu, "A comprehensive review and comparison of existing computational methods for intrinsically disordered protein and region prediction," *Briefings in bioinformatics*, vol. 20, no. 1, pp. 330–346, 2019.

[28] J. C. Wootton and S. Federhen, "Statistics of local complexity in amino acid sequences and sequence databases," *Computers & chemistry*, vol. 17, no. 2, pp. 149–163, 1993.

[29] V. J. Promponas, A. J. Enright, S. Tsoka, D. P. Kreil, C. Leroy, S. Hamodrakas, C. Sander, and C. A. Ouzounis, "Cast: an iterative algorithm for the complexity analysis of sequence tracts," *Bioinformatics*, vol. 16, no. 10, pp. 915–922, 2000.

[30] D. Harbi, M. Kumar, and P. M. Harrison, "Lps-annotate: complete annotation of compositionally biased regions in the protein knowledgebase," *Database*, vol. 2011, 2011.

[31] P. M. Harrison, "flps: Fast discovery of compositional biases for the protein universe," *Bmc Bioinformatics*, vol. 18, no. 1, pp. 1–9, 2017.

[32] Z. Dosztányi, V. Csizmok, P. Tompa, and I. Simon, "Iupred: web server for the prediction of intrinsically unstructured regions of proteins based on estimated energy content," *Bioinformatics*, vol. 21, no. 16, pp. 3433–3434, 2005.

[33] Z. Dosztányi, B. Mészáros, and I. Simon, "Anchor: web server for predicting protein binding regions in disordered proteins," *Bioinformatics*, vol. 25, no. 20, pp. 2745–2746, 2009.

[34] B. Mészáros, I. Simon, and Z. Dosztányi, "Prediction of protein binding regions in disordered proteins," *PLoS computational biology*, vol. 5, no. 5, p. e1000376, 2009.

[35] I. Walsh, A. J. Martin, T. Di Domenico, and S. C. Tosatto, "Espritz: accurate and fast prediction of protein disorder," *Bioinformatics*, vol. 28, no. 4, pp. 503–509, 2012.

[36] M. Necci, D. Piovesan, Z. Dosztányi, and S. C. Tosatto, "Mobidb-lite: fast and highly specific consensus prediction of intrinsic disorder in proteins," *Bioinformatics*, vol. 33, no. 9, pp. 1402–1404, 2017.

[37] T. Zhang, E. Faraggi, B. Xue, A. K. Dunker, V. N. Uversky, and Y. Zhou, "Spine-d: accurate prediction of short and long disordered regions by a single neural-network based method," *Journal of Biomolecular Structure and Dynamics*, vol. 29, no. 4, pp. 799–813, 2012.

[38] S. Wang, J. Ma, and J. Xu, "Aucpred: proteome-level protein disorder prediction by auc-maximized deep convolutional neural fields," *Bioinformatics*, vol. 32, no. 17, pp. i672–i679, 2016.

[39] Y.-J. Tang, Y.-H. Pang, and B. Liu, "Idp-seq2seq: identification of intrinsically disordered regions based on sequence to sequence learning," *Bioinformatics*, vol. 36, no. 21, pp. 5177–5186, 2020.

[40] S. G. Tringe and E. M. Rubin, "Metagenomics: Dna sequencing of environmental samples," *Nature reviews genetics*, vol. 6, no. 11, pp. 805–814, 2005.

[41] C. Y. Chiu and S. A. Miller, "Clinical metagenomics," *Nature Reviews Genetics*, vol. 20, no. 6, pp. 341–355, 2019.

[42] T. Thomas, J. Gilbert, and F. Meyer, "Metagenomics-a guide from sampling to data analysis," *Microbial informatics and experimentation*, vol. 2, no. 1, pp. 1–12, 2012.

[43] A. Escobar-Zepeda, A. Vera-Ponce de León, and A. Sanchez-Flores, "The road to metagenomics: from microbiology to dna sequencing technologies and bioinformatics," *Frontiers in genetics*, vol. 6, p. 348, 2015.

[44] F. Sanger and A. R. Coulson, "A rapid method for determining sequences in dna by primed synthesis with dna polymerase," *Journal of molecular biology*, vol. 94, no. 3, pp. 441–448, 1975.

[45] V. Kunin, A. Copeland, A. Lapidus, K. Mavromatis, and P. Hugenholtz, "A bioinformatician's guide to metagenomics," *Microbiology and molecular biology reviews*, vol. 72, no. 4, pp. 557–578, 2008.

[46] J. Alneberg, B. S. Bjarnason, I. De Bruijn, M. Schirmer, J. Quick, U. Z. Ijaz, L. Lahti, N. J. Loman, A. F. Andersson, and C. Quince, "Binning metagenomic contigs by coverage and composition," *Nature methods*, vol. 11, no. 11, pp. 1144–1146, 2014.

[47] D. C. Richter, F. Ott, A. F. Auch, R. Schmid, and D. H. Huson, "Metasim—a sequencing simulator for genomics and metagenomics," *PloS one*, vol. 3, no. 10, p. e3373, 2008.

[48] A. Shcherbina, "Fastqsim: platform-independent data characterization and in silico read generation for ngs datasets," *BMC research notes*, vol. 7, no. 1, pp. 1–12, 2014.

[49] A. Fritz, P. Hofmann, S. Majda, E. Dahms, J. Dröge, J. Fiedler, T. R. Lesker, P. Belmann, M. Z. DeMaere, A. E. Darling, *et al.*, "Camisim: simulating metagenomes and microbial communities," *Microbiome*, vol. 7, no. 1, pp. 1–12, 2019.

[50] W. R. Atchley, J. Zhao, A. D. Fernandes, and T. Drüke, "Solving the protein sequence metric problem," *Proceedings of the National Academy of Sciences*, vol. 102, no. 18, pp. 6395–6400, 2005.

[51] S. Kawashima, P. Pokarowski, M. Pokarowska, A. Kolinski, T. Katayama, and M. Kanehisa, "Aaindex: amino acid index database, progress report 2008," *Nucleic acids research*, vol. 36, no. suppl_1, pp. D202–D205, 2007.

[52] I. Walsh, A. J. Martin, T. Di Domenico, A. Vullo, G. Pollastri, and S. C. Tosatto, "Cspritz: accurate prediction of protein disorder segments with annotation for homology, secondary structure and linear motifs," *Nucleic acids research*, vol. 39, no. suppl_2, pp. W190–W196, 2011.

[53] R. Linding, R. B. Russell, V. Neduva, and T. J. Gibson, "Globplot: exploring protein sequences for globularity and disorder," *Nucleic acids research*, vol. 31, no. 13, pp. 3701–3708, 2003.

[54] W. Huang, L. Li, J. R. Myers, and G. T. Marth, "Art: a next-generation sequencing read simulator," *Bioinformatics*, vol. 28, no. 4, pp. 593–594, 2012.

[55] E. Singer, B. Andreopoulos, R. M. Bowers, J. Lee, S. Deshpande, J. Chiniquy, D. Ciobanu, H.-P. Klenk, M. Zane, C. Daum, *et al.*, "Next generation sequencing data of a defined microbial mock community," *Scientific data*, vol. 3, no. 1, pp. 1–8, 2016.

[56] Z.-L. Peng and L. Kurgan, "Comprehensive comparative assessment of in-silico predictors of disordered regions," *Current Protein and Peptide Science*, vol. 13, no. 1, pp. 6–18, 2012.

[57] J. Cheng, M. J. Sweredoski, and P. Baldi, "Accurate prediction of protein disordered regions by mining protein structure data," *Data mining and knowledge discovery*, vol. 11, no. 3, pp. 213–222, 2005.

[58] M. Necci, D. Piovesan, and S. C. Tosatto, "Critical assessment of protein intrinsic disorder prediction," *Nature methods*, vol. 18, no. 5, pp. 472–481, 2021.

[59] A. Schlessinger, G. Yachdav, and B. Rost, "Profbval: predict flexible and rigid residues in proteins," *Bioinformatics*, vol. 22, no. 7, pp. 891–893, 2006.

[60] A. Schlessinger, J. Liu, and B. Rost, "Natively unstructured loops differ from other loops," *PLoS computational biology*, vol. 3, no. 7, p. e140, 2007.

[61] Z. R. Yang, R. Thomson, P. McNeil, and R. M. Esnouf, "Ronn: the bio-basis function neural network technique applied to the detection of natively disordered regions in proteins," *Bioinformatics*, vol. 21, no. 16, pp. 3369–3376, 2005.

[62] K. Peng, P. Radivojac, S. Vucetic, A. K. Dunker, and Z. Obradovic, "Length-dependent prediction of protein intrinsic disorder," *BMC bioinformatics*, vol. 7, no. 1, pp. 1–17, 2006.

[63] J. J. Ward, J. S. Sodhi, L. J. McGuffin, B. F. Buxton, and D. T. Jones, "Prediction and functional analysis of native disorder in proteins from the three kingdoms of life," *Journal of molecular biology*, vol. 337, no. 3, pp. 635–645, 2004.

[64] B. Xue, R. L. Dunbrack, R. W. Williams, A. K. Dunker, and V. N. Uversky, "Pondr-fit: a meta-predictor of intrinsically disordered amino acids," *Biochimica et Biophysica Acta (BBA)-Proteins and Proteomics*, vol. 1804, no. 4, pp. 996–1010, 2010.

[65] Y. Liu, X. Wang, and B. Liu, "Rfpr-idp: reduce the false positive rates for intrinsically disordered protein and region prediction by incorporating both fully ordered proteins and disordered proteins," *Briefings in bioinformatics*, vol. 22, no. 2, pp. 2000–2011, 2021.

[66] A. Schlessinger, M. Punta, G. Yachdav, L. Kajan, and B. Rost, "Improved disorder prediction by combination of orthogonal approaches," *PloS one*, vol. 4, no. 2, p. e4433, 2009.

[67] M. J. Mizianty, W. Stach, K. Chen, K. D. Kedarisetti, F. M. Disfani, and L. Kurgan, "Improved sequence-based prediction of disordered regions with multilayer fusion of multiple information sources," *Bioinformatics*, vol. 26, no. 18, pp. i489–i496, 2010.

[68] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[69] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, pp. 8026–8037, 2019.

[70] M. Y. Lobanov, I. V. Sokolovskiy, and O. V. Galzitskaya, "Isunstruct: prediction of the residue status to be ordered or disordered in the protein chain by a method based on the ising model," *Journal of Biomolecular Structure and Dynamics*, vol. 31, no. 10, pp. 1034–1043, 2013.

[71] C. Mirabello and B. Wallner, "Rawmsa: End-to-end deep learning using raw multiple sequence alignments," *PloS one*, vol. 14, no. 8, p. e0220182, 2019.