Università degli Studi di Salerno

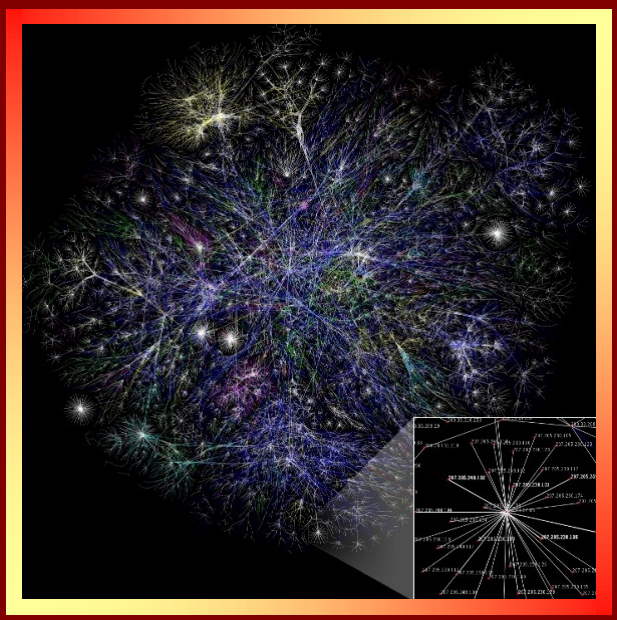
Dipartimento di Informatica

Dottorato di Ricerca in Informatica – XXXIV Ciclo

Tesi di Dottorato/Ph.D. Thesis

Application of machine learning techniques to biological big data

Alessia Auriemma Citarella



Supervisor: **Prof. Genoveffa Tortora**

Ph.D. Program Director: **Prof. Andrea De Lucia**

AA 2020/2021

Curriculum Computer Science and Information Technology

Università degli Studi di Salerno

Dipartimento di Informatica

Dottorato di Ricerca in Informatica
XXXIV Ciclo

TESI DI DOTTORATO / PH.D. THESIS

Application of machine learning techniques to biological big data

ALESSIA AURIEMMA CITARELLA

SUPERVISOR: **PROF. GENOVEFFA TORTORA**

PHD PROGRAM DIRECTOR: **PROF. ANDREA DE LUCIA**

A.A 2020/2021

To my family,
always present every day of my life.

Dedicated to the loving memory of my Ancestors.

In Memoriam of Professor Michele Risi.
You touched our heart with kindness,
warmth and a blessed love.

The important thing is not to stop questioning.
Curiosity has its own reason for existence.
One cannot help but be in awe when he contemplates
the mysteries of eternity, of life, of the marvelous
structure of reality.
It is enough if one tries merely to comprehend
a little of this mystery each day.

— Albert Einstein in *Life Magazine* (2 May 1955)

ACKNOWLEDGMENTS

I'd like to express my gratitude to everyone who accompanied me on this adventure. This work is dedicated to them.

This dissertation contains the scientific results achieved, clearly visible, but also many experienced emotions which can not be described in words. This PhD was a turning point in my life, both professionally and personally. It was difficult for me to get closer to a new world as a biologist. To investigate well-known application fields, I had to learn to see things through new eyes. I needed to strike a balance between what I wanted to do with my life and what I was doing. I frequently felt as if I was missing something or lacked expertise in the field. Nobody forced me to consider how I felt. Every person who has crossed my path has always accentuated my strengths, particularly while I was only focused on my own weaknesses.

I want to express my gratitude to my scientific advisor, Professor Genoveffa Tortora, who has guided me to this outcome with patience, passion and love.

During these three years, Professor Michele Risi was a carrier column. There would have been no way forward without him. You've been a beacon in the darkness, a trustworthy confidant to entrust both tiny wins and failures, including anxiety and fear of not being up to the task.

A special thanks go to my colleagues, Luigi di Biasi, Luigi De Maio, Marcello Barbella and Fabiola De Marco. All of them have supported me and supported in the worst moments of this path.

It was not a way on uphill but thanks to everyone's uniqueness, I never felt alone. In these three years, the most significant gift I received was their soul, which was priceless and immeasurable, as well as the wealth derived from their unconditional friendship. In the name of this gift, I hope you remember us and never forget us, any way we will take.

Looking at these three years I leave behind me, my greatest conquest is the awareness that I can always do something. I can learn, I can experiment new horizons, I can succeed where I am afraid of failure. Personally, I pared down my self-baggage. esteem's. This is a work in progress and I believe it will last a lifetime.

In this last year I found a safe harbor in three angels on the ground: Roberta D., Roberta and Mary. They are my family of souls. It is not the blood to make us sisters but it is our shared heart. I know they will always be there, in this and I hope in other lives because what binds us is indissoluble. It's that besides that gives meaning to life.

Not least, thanks to the love of my life, Gianluigi, who always encouraged me to follow my dreams. Faced with my fears, anxieties and inner collapses, I find my stillness in you. Every day with you, I understand that love is an impetus to rebuild yourself, to downsize your blind spots and to let the other see them without shame. I divided the weights and cultivated joy with you. I cultivated joy by dividing the weights with you. You have been the best gift life has given me.

The best part of these three years remains my family. It was there while the world collapsed on me. He keeps me always by hand, in good and bad fate. The voice of the loving verb is this: it combines with the word family. My parents give me to the world in my integrity of human being. I am the irrepressible part of me, the one that I will always carry with me.

ABSTRACT

To date, has been the primary driver of global innovation, competitiveness and cultural development. It is also a powerful engine for creating new job opportunities, expanding market segments and inspiring new horizons where new skills and specialties can compete. From this perspective, we are constantly pushed to investigate the ICT industry and its interconnections with other areas, such as new biomedical technology.

In the past twenty years, the development and increase of new diagnostic methods in the medical field has made available a huge amount of data capable of being stored and analyzed in order to extract important new knowledge.

In the biological field, the data produced by the sequencing techniques and the available databases provide a lot of information on multiple levels that can be integrated with each other. The ability to integrate and analyze data from multiple sources is vital in order to collect real benefits and speed up outputs thanks to the high computational possibilities of some tools.

Technology has the potential to dramatically change the conception of medicine and, at the same time, it plays a critical role in advanced diagnostics systems in making decisions intrinsic to patient care. Developing high-quality, accurate Artificial Intelligence (AI) resources improves work of clinicians by intervening on prevention, diagnosis and treatment of many pathologies. Some modern AI and computer science technologies, in general, encompass the power of clinical laboratory devices, allowing diagnostic activity to be carried out even outside of laboratories. Medical aids and new advanced diagnostic equipment are increasingly relying on qualified experts in the field to supplement medical evaluations and assist in diagnosis.

In this context, the focus of this work was on two main topics. First one, we explored additional Machine Learning and Deep Learning techniques that can guarantee a better classification of melanoma images even on clinical datasets with lower image quality. The goal is to improve melanoma early detection,

which is now a limiting factor for first-line therapies in this tumor pathology. Many of the research in the literature utilize similar strategies but use various approaches: some try to extract information directly from the image (such as color, plot and pixel density), while others try to extract functions based on guided lines of dermatologists (such as ABCDE and the Seven-Point Checklist). The majority of these researches are conducted using higher-resolution dermoscopic pictures. The purpose of the research is to identify novel features for melanoma classification that may be applied to less detailed images using advanced learning techniques.

The second contribution of this thesis is addressed to the classification of proteins. Researches focused on the possibility of exploring further molecular descriptors in addition to those already present in the literature to classify these proteins and to build new tools able to explore the complex interaction between proteins in a visual and intuitive way. In this line of research, the visualization of biological data was also taken into consideration. The work has mostly concentrated on the presentation tools of biological ontologies in order to develop user-friendly systems that allow end users to interact and extrapolate information more easily. This is useful for the complexity of the biological system that can be explored by the integration of omics disciplines. These sciences attempt to analyze the biological system holistically using biological Big Data, mainly proteomic, genomic, transcriptomic and metabolomic data. The latter are the most important groupings of organic compounds for the study of the functioning of living organisms.

ABSTRACT IN ITALIANO

Finora, la tecnologia è stata il motore principale dell'innovazione, della competitività e dello sviluppo culturale globali. È anche un potente motore per creare nuove opportunità di lavoro, espandere segmenti di mercato e ispirare nuovi orizzonti in cui nuove competenze e specialità possono competere. Da questo punto di vista, siamo costantemente spinti a indagare l'industria ICT e le sue interconnessioni con altre aree, come le nuove tecnologie

biomediche. Negli ultimi vent'anni, lo sviluppo e l'incremento di nuove metodiche diagnostiche in campo medico ha reso disponibile un'enorme quantità di dati in grado di essere archiviati e analizzati al fine di estrarre nuove importanti conoscenze. In campo biologico, i dati prodotti dalle tecniche di sequenziamento e le banche dati disponibili forniscono molte informazioni su più livelli che possono essere integrate tra loro. La capacità di integrare e analizzare dati provenienti da più fonti è fondamentale per raccogliere benefici reali e velocizzare gli output grazie alle elevate possibilità computazionali di alcuni strumenti. La tecnologia ha il potenziale per cambiare radicalmente la concezione della medicina e, allo stesso tempo, svolge un ruolo fondamentale nei sistemi diagnostici avanzati nel prendere decisioni intrinseche alla cura del paziente. Lo sviluppo di risorse di intelligenza artificiale (AI) accurate e di alta qualità migliora il lavoro dei medici intervenendo sulla prevenzione, la diagnosi e il trattamento di molte patologie. Alcune moderne tecnologie di intelligenza artificiale e informatica, in generale, racchiudono la potenza dei dispositivi di laboratorio clinico, consentendo di svolgere attività diagnostiche anche al di fuori dei laboratori. I presidi medici e le nuove apparecchiature diagnostiche avanzate si affidano sempre più a esperti qualificati del settore per integrare le valutazioni mediche e assistere nella diagnosi. In questo contesto, il focus di questo lavoro è stato su due temi principali. Innanzitutto, abbiamo esplorato ulteriori tecniche di Machine Learning e Deep Learning in grado di garantire una migliore classificazione delle immagini del melanoma anche su set di dati clinici con una qualità dell'immagine inferiore. L'obiettivo è migliorare la diagnosi precoce del melanoma, che ora è un fattore limitante per le terapie di prima linea in questa patologia tumorale. Molte delle ricerche in letteratura utilizzano strategie simili ma usano approcci diversi: alcune cercano di estrarre informazioni direttamente dall'immagine (come colore, trama e densità di pixel), mentre altre cercano di estrarre funzioni basate su linee guidate dai dermatologi (come l' ABCDE e la Seven-Point Checklist). La maggior parte di queste ricerche viene condotta utilizzando immagini dermoscopiche ad alta risoluzione. Lo scopo della ricerca è identificare nuove caratteristiche per la classificazione del melanoma che possono essere applicate a immagini meno dettagliate utilizzando tecniche di

apprendimento avanzate. Il secondo contributo di questa tesi è rivolto alla classificazione delle proteine. La ricerca si è concentrata sulla possibilità di esplorare ulteriori descrittori molecolari oltre a quelli già presenti in letteratura per classificare queste proteine e per costruire nuovi strumenti in grado di esplorare la complessa interazione tra proteine in modo visivo e intuitivo. In questo filone di ricerca è stata presa in considerazione anche la visualizzazione dei dati biologici. Il lavoro si è concentrato principalmente sugli strumenti di presentazione delle ontologie biologiche al fine di sviluppare sistemi user-friendly che consentano agli utenti finali di interagire ed estrapolare le informazioni più facilmente. Ciò si inserisce nell'ottica della complessità del sistema biologico che può essere esplorato dall'integrazione delle discipline omiche. Queste scienze tentano di analizzare il sistema biologico olisticamente utilizzando Big data biologici, principalmente dati proteomici, genomici, transcriptomici e metabolomici. Questi ultimi sono i raggruppamenti più importanti di composti organici per lo studio del funzionamento degli organismi viventi.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

The Chapter provides an overview of the content of this work, the main issues addressed, and how the thesis is structured. We start with a brief description of Big Data (see Section 1.1) and how they can modify our perception and understanding of the world (in Section 1.2), in particular in relation to treated fields of application (Section 1.3). This research adds to the fields of Big Data mining because the processed data represent a great opportunity to improve our knowledge and encourage further progress in the biomedical field. Then, we discuss our contributions in Section 1.4 and how the thesis is organized (Section 1.5).

1.1 INTRODUCTION TO BIG DATA

Big Data has revolutionized the world we live in over the previous few decades, opening up new opportunities in a variety of fields, from business to industry and public sectors [27]. Today, the world is saturated with information like it has never been before and the amount of knowledge available is growing at an exponential rate, particularly in the last twenty years. The expansion of Big data has been made possible by three basic conditions: an increase in the availability of information, an improvement in the processing and storage capacity of the data and, finally, the economic convenience of obtaining the two aspects mentioned above, as compared to the past [64]. Two factors are causing an increase in the availability of information: datization and the Internet of Things (IoT). The process of transforming given phenomena into a quantitative form so that it may be tabulated and examined is known as *datification*. It reflects the driving force behind the Big Data phenomena, enriching the information with new forms of value, including economic ones [59]. IoT extends the ability to collect, process and exchange data from a multitude

of sources to real-world objects, which are often equipped with ubiquitous intelligence [187].

The exploration of this large amount of data poses several challenges due to some of their properties [40], including:

- *Variety*: Many data are semi-structured, raw, structured and even unstructured, and they often come from different sources;
- *Volume*: Big Data are considerably voluminous, and it is assumed that in the future they will reach the size of the zettabytes. These dimensions cannot be analyzed with current traditional systems;
- *Complexity*: Connect the data from different sources and different formats is one of the intrinsic complexities of the large volumes of data;
- *Velocity*: In some contexts, Big Data are generated in real-time and this is advantageous for some types of analysis, but this opens up a challenge to current realities and technologies to exploit data coming at high speeds just as quickly;
- *Value*: It refers to the process of finding a high value hidden within numerous different and rapidly growing data. This is closely related to the veracity and quality of the data.

All of these traits, as well as other aspects of Big Data, make traditional methodologies less efficient in terms of analysis [92].

1.2 HOW BIG DATA CAN CHANGE *know-how*

Big Data and, in particular, the Internet of Things (IoT) allow us to make data more accessible to better understand and enhance our understanding of the real world. They enable us to extract data for decision-making processes, assist firms in pursuing digital business innovation paths, generate new knowledge, get new insights and convert and reuse metadata in a new productive factor. In the context of this thesis, we are interested in the concept of Data Science, namely the combination of different disciplines

such as statistics, data mining, databases and distributed systems that represents a new data-intensive approach to scientific discovery [179]. It is especially in the field of healthcare that Big Data opens up to new opportunities of value, especially for their possibility of being used in decision support systems. Adding value and enhancing competence are the main consequences of extracting new knowledge. This concept is particularly prevalent in the medical industry, as gaining new knowledge can also imply improving the quality of life for a patient. This adding value is possible with the use of the Big data analytics, the application of advanced analytics techniques to large data sets. In this field, several applications are widely used, including predictive models, statistical tools, algorithms trained on this large amount of data [44]. These applications must take into account the nature (structured, unstructured, and semi-structured data) of the big data on which they operate. In fact, they can be complex to manage, especially when they come from multiple resources. In healthcare, where biomedical data is processed, the many steps of pre-processing, selection, transformation, extraction, assessment, and representation of data are becoming increasingly significant.

Generally, biological big data has qualities similar to the 4Vs of Big data. We can summarize their characteristics in [115]:

- *hierarchy*: Biological big data reflect the normal structural hierarchy (molecules, cells, tissues, and systems) present in our body;
- *heterogeneicity*: Because they are generated in different methods ranging from genetics, physiology, anatomy to imaging;
- *complexity*: Biological big data are multi-level information formed by relationships between many molecular interactors, atomic or even more;
- *dynamics*: Each process changes over time depending on the conditions of each part of the biological system.

One of the greatest issues of biological big data is represented by the great effort employed in analyzing the complex networks of the living organism and their connections to discover new non-casual relationships. The goal of contemporary challenges

is to provide relevant tools to the scientific community, as well as infrastructure such as cloud computing, in order to study biological big data. The ability of diverse methodologies to process heterogeneous data, to employ algorithms that guarantee efficiency and scalability to optimize the potential value of the data, all affect the analytical phases in this field. This study of bioinformatics data adds value in the form of increased *know-how*, the amount of knowledge and skills required to understand a biological phenomenon. Because biological data is so extensive and heterogeneous, new knowledge derived from it aids in a better understanding of gene expression, regulation, and hereditary disorders associated with them in the case of mutations or metabolic dysfunctions. This allows to developing a global vision (also defined as *holistic*) of the biological system. This new knowledge must be part of the cultural heritage of every medical actor, as they will involve more and more the appearance of new scenarios and new diagnostic and therapeutic approaches.

1.3 FIELDS OF APPLICATIONS

Over the last two decades, the creation and expansion of new diagnostic procedures in the biomedical field have created an immense amount of data that may be saved and analyzed to extract new significant knowledge in this field of application. This exponential increase began with the sequencing of the human genome, a project that aimed to map the nucleotides contained in a human genome. Over time, this has permitted the development of online databases containing biological Big Data, primarily proteomics, transcriptomics, genomics, and metabolomics data (considered as *omics* sciences), frequently accompanied by massive data tables of experimental observations. At the same time, the diffusion of Electronic Health Records (EHR), appropriately anonymized, has made greater availability of information on the health of individuals or a population. These records include demographic information, medical history, diagnosis, potential therapies, laboratory test results, biomedical images (processed using radiodiagnostics, computerized tomography, ultrasound, nuclear medicine techniques and magnetic resonance), vital signs, and personal information such as age and weight. So, also the

value of digitized medical images has grown over time since their processing and analysis using advanced mathematical algorithms allows researchers to obtaining information about underlying physiopathological events that are not detectable by visual examination alone. The ability to capture in vivo snapshots of common physiological or alternative pathological processes using specific and advanced sensors and computerized technology has highlighted the significance of being able to apply virtual reality, computer vision and robotics to biomedical imaging problems. Biomedical research is rapidly responding to the convergence of the biomedical and Information Technology (IT) sectors, which opens up new perspectives and opportunities to expand existing knowledge. Machine learning (ML), in particular Deep Learning (DL), is supporting traditional biomedical research by developing new features extraction techniques from images and new statistical classifiers for the detection and diagnosis of pathologies through the use of instruments. This provides for a better representation of accessible data and subsequent analysis, thanks to an integrated strategy that ensures capturing information at different levels.

1.4 CONTRIBUTIONS OF THIS THESIS

The biological networks of the omics science are very large and complex. We are in the presence of extremely complex problems for the number of combinations of possibilities to be analyzed. In the works presented in this thesis, it was necessary to perform a training and study phase on large datasets (such as ISIC[1], HAM10000 [175] and so on) in order to examine the small datasets presented here (i.e, MED-NODE [68]). Then, we employed the Transfer Learning to boost our performance.

The first contribution of this thesis regards the development and exploration of new techniques for extracting features from images and data and new classifiers for the detection and early diagnosis of the melanoma through Machine Learning and Deep Learning. In particular, we explored the use of deep neural networks and its combination with genetic algorithms, and an ap-

1 https://challenge.isic-archive.com/

proach based on a hybrid architecture system. The ability to employ classification algorithms on clinical images is one of the issues that has emerged in the field of melanoma detection. Unlike dermoscopic images, these clinical images are low-resolution images whose use is more prevalent and is directly tied to their use via Smart Device. It is precisely this open challenge that guides the search in the following chapters of this thesis. The second contribution relates to new insights for extracting characteristics in order to classify and visualize some protein families. We concentrated on the application of deep learning and machine learning to the problem of protein classification and prediction of protein angles, as well as an interactive system for visualizing biological data. The significance of tracing back to the structure of a protein from its amino acid sequence is due to the intimate relationship between structure and function. Structure is thought to affect the function and properties of proteins, as well as their functioning within biological processes [138, 140]. At the same time, improving the classification of a protein and tracing it back to a family of proteins is advantageous for the evolutionary reconstruction of the protein itself. In fact, a family of proteins is made up of proteins that perform slightly similar functions and, over time, preserve the three-dimensional conformation rather than the sequence of amino acids. This information is useful for finding significant sites, patterns, and profiles that affect the functionality of the protein, even if the sequence similarity between multiple proteins of the same family is low. The visualization of biological data occurs in the context of the overall vision provided by systems biology, which considers how an organism functions as a whole. In different domains of systems biology, bioinformatics tools for analysis, interpretation, and prediction of biological data provide this *omic* vision of biology and allow us improving our knowledge of produced data.

1.5 THESIS OUTLINE

The rest of the thesis is organized as follows:

- In *Chapter 2* there is a general overview of the state of the art relating to the classification problems faced for melanoma

and for proteins. We discuss some general concepts indispensable for understanding the following chapters of our work. It begins with a review of some processes required for melanoma detection and protein identification, and ends with a description of the classification algorithms utilized to address these issues.

- In *Chapter 3*, the state of the art on the detection of melanoma and the approaches explored in this work are presented. The proposal of a Fog/Cloud/Edge hybrid architecture can handle the pre-processing and the next classification of melanoma clinical images, reducing the execution time and continuous iterations of the training set necessary to provide robust models of forecasting. Secondly, the experimental results obtained using the functions of genetic algorithms (selection, mutation, and crossover) are presented with the Neural AlexNet network in an evolutionary Convolutional Neural Network (CNN) design approach.

- In *Chapter 4*, the use of a bidirectional *Long Short-Term Memory* (LSTM) neural network is proposed for prediction of protein torsional angles. In this context, the addition of four new molecular descriptors is examined to improve the performance of the network. In details, the chapter describes the LSTM architecture and all details related to the implementation of three variants of this structure for our experiments. Furthermore, we apply two methods to visualize the data obtained from the predictions of the two torsional angles.

- In *Chapter 5*, the introduction of new molecular descriptors, called SNARER descriptors, is evaluated to improve the quality and efficiency of binary classifiers focused on the protein family called SNARE (*Soluble N–ethylmaleimide sensitive factor Attachment protein Receptor*). In particular, we used three classification algorithms, Random Forest, AdaBoost and k-nearest neighbors in order to compare their performance on balanced and unbalanced datasets.

- In *Chapter 6*, based on the components of the Gene Ontology (cellular component, molecular function and biological

process), the largest resource accessible for enriching biological analyses, we presented a way to assess graphically the similarity data between Parkinson's and Alzheimer's proteins. In this work we compare a partitional group analysis method, the K-Means, with our Dynamic Distance-Graph-based approach that takes into account the similarities between the components of the Ontology Gene. In this method, we hope to retrieve the biological information of proteins from a global perspective, which ties together the three ontological domains and allows us for a more comprehensive overall view.

- Finally, conclusions and future studies follow in *Chapter 7*. In this chapter, the contributions proposed by this work and any future directions are highlighted.

2

RELATED WORKS

This Chapter starts with a description of the skin structure (see Section 2.1) and melanoma (see Section 2.2), in order to address the key point of melanoma detection. The prediction of the proteins' structure remains one of the most interesting and most complex arguments in structural biology and bioinformatics. So, we present an overview of the structural characteristics of proteins in Section 2.3, necessary to understand the motivations underlying two of the topics covered in the following chapters. Then, we discuss the most commonly used classification methods for the classification of melanoma and proteins (in Section 2.4), with particular attention to deep learning methods for skin lesions and proteins classification (see Section 2.5 and Section 2.6).

2.1 SKIN STRUCTURE

The skin is a protective covering that surrounds and protects the human body. Appearance, thickness, color, elasticity, and extensibility are its key macroscopic features. The skin is characterized by a stratified structure articulated in:

- *epidermis*, which is the most superficial layer;
- *dermis*, under the epidermis;
- *hypoderma*, the deeper layer.

In addition to having multiple sensory terminations, the skin has a barrier role that protects the body from potentially dangerous substances, a thermoregulation function that helps to maintain body temperature, a secretory, a metabolic and immunological functions [126]. Melanin is a dark brown pigment produced by melanocytes, which are a type of cell present in the basal layer of epidermis. It regulates the color of the skin, of the hair and of the eyes, absorbs solar radiation and protects the

genome from UV rays while neutralizing free radicals. Melanin has been discovered to play a role in determining the behavior of skin cancer in recent years, owing to its role in controlling epidermal homeostasis [164].

2.2 MELANOMA

Skin cancer is described as the uncontrollable proliferation of skin cells caused by DNA damage. Melanoma is a form of skin cancer that arises from melanocytes and is one of the deadliest tumor in the world [110] because it has the ability to rapidly metastasize to different tissues [45]. Melanoma development is influenced by both genetic and environmental risk factors. Caucasian race, light-colored skin, the number, and kind of nevi and a positive family history of melanoma are all genetic variables. We prioritized burns, UV rays exposure which has genotoxic effect and sunburn history among the environmental risk variables [148]. In most cases, a first visual assessment of a dermatologist is used to diagnose melanoma, in clinical practice. The capacity of a physician to distinguish between different forms of skin lesions is also dependent on his level of experience. The tumor thickness, ulceration, and metastasis to lymph nodes or other regions of the body are all taken into account when determining the stage of melanoma. It has five main stages, based on the *American Joint Committee on Cancer Staging Manual* [66]:

- *Stage 0*, also called *melanoma in situ*, where the epidermis, the outer layer of the skin, contains abnormal melanocytes. These cells have the potential to become cancerous;
- *Stage I*, divided in *Stage IA* and *Stage IB*, where we can consider thickness and ulceration. In Stage IA, there is no ulceration and the tumor is less than 1 mm thick, but we can see the formation of a skin break. In Stage IB, either the tumor has ulceration, but it is not more than 1 mm thick, or it is between 1 and 2 mm thick, but there is no ulceration;
- *Stage II*. This stage is divided into *Stage IIA*, *Stage IIB* and *Stage IIC*. In Stage IIA, the tumor has ulceration, but it is not more than 1 mm thick, or it is between 1 and 2 mm thick,

but it does not have ulceration. In Stage IIB, the tumor is ulcerated and it is greater than 2 mm thick but not more than 4 mm thick, or it is thicker than 4 mm but does not have ulceration. In Stage IIC, the tumor is thicker than 4 mm and exhibits ulceration;

- *Stage III*: we can consider the spread of cancer, a process called *metastasis*. With or without ulceration, a tumor might be any thickness. It could have spread to one or more lymph nodes or cancer cells could be at least 2 cm away from the initial tumor in a lymph vessel or there could be smaller tumors on/under the skin in a 2 cm radius surrounding the primary tumor;

- *Stage IV*: melanoma may have progressed to other regions of the body, such as the lungs, brain and liver, which are often far from the main tumor.

It is critical to understand the stages of tumors in order to evaluate its therapy and prognosis. To assess the stages of melanotic cancer, a variety of approaches can be used: lymph node mapping [12], Computed Tomography (CT) scan and Positron Emission Tomography (PET) scan [159], Magnetic Resonance Imaging (MRI), blood chemistry tests [43]. The excision of the lesion alone is insufficient for treating a melanoma in Stage III or Stage IV. In this case, chemotherapy [6], radiation therapy [186], immunotherapy [157] and targeted therapy [15] are required treatments. For prevention, the more effective treatment is surgical removal of the original tumor before tumor cells detach the lymph nodes, causing the tumor to spread quickly.

2.3 PROTEINS STRUCTURE

Proteins are biopolymers made up of 20 separate components, all of which are known as amino acids. The set of amino acid symbols are: $\{a, r, n, d, c, e, q, g, h, l, i, k, m, f, p, s, t, w, y, v\}$.

Covalent connections (peptide links), various types of ties (disulfide bridges) and non-covalent bonds (saline bridges, hydrogen links, van der Waals interactions and hydrophobic forces)

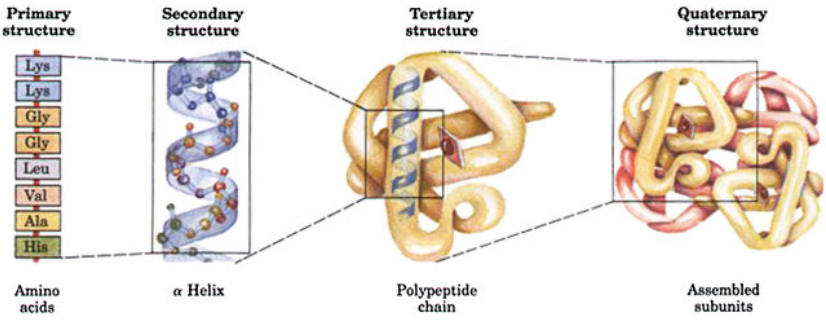


Figure 2.1: Four types of protein structures.

hold amino acids together. Proteins are characterized by four fundamental structures [19], as depicted in Figure 2.1 [132]:

1. *primary structure* is the sequence of various amino acids, which is determined by the nucleotide sequence in the coding gene;

2. *secondary structure*: α-helix, β-sheet and coiled coils are three different types of secondary structure. The angles must be taken into account when the protein is folded locally. The main angles are:

   - ϕ angle between N and Cα;
   - ψ angle between Cα and carbonyl carbon;
   - ω angle between carbonyl carbon and N, commonly fixed at 180°.

   The ϕ and ψ angles, called *torsional* or *dihedral* (showed in Figure 2.2 [69]) , do not take random values; instead, they must distribute in a highly exact range in order for the proteins to fold correctly. These ranges are explained by the Ramachandran plot [23], which showing the combinations of the two torsional angles admitted inside a protein structure, according to the different secondary structures;

3. *tertiary structure*: it indicates the three-dimensional organization of the protein. In a single protein, it reflects the folding of multiple secondary structural parts. At this level, a correctly folded protein is correlated to its own specific function;

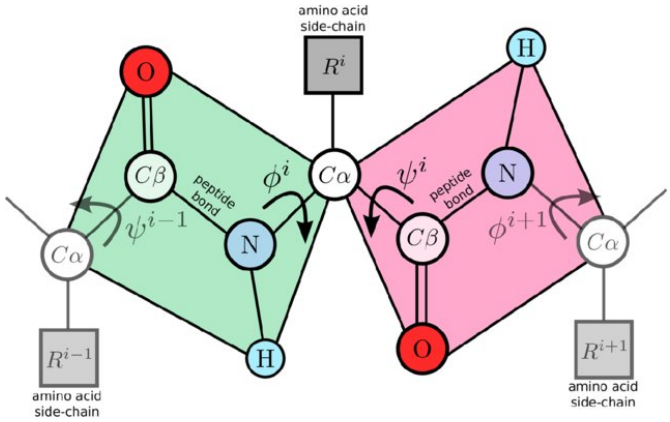


Figure 2.2: Torsional angles in proteins.

4. *quaternary structure*: the spatial organization of several protein molecules in multi-subunit complexes can be seen at this level of protein organization. Each protein represents a subunit and proteins interacting in the quaternary complex can be similar or dissimilar.

2.3.1 *Traditional experimental methods*

The process of determining the primary structure of a biopolymer is referred to as *sequencing*. The sequencing of genetic material has paved the ground for modern biotechnology in recent decades. If the sequenced material is a nucleic acid (DNA or RNA), this sequence is made up of nucleotides. If it is a protein, it is made up of amino acids. Over time, a series of advancements in sequencing techniques have enabled them to become simple enough to integrate into a laboratory's routine. The current techniques work in parallel, simultaneously performing millions of reactions and generating a large amount of data [76]. Bioinformatics has enabled the development of tools for generating and maintaining data from other sources, in addition to sequencing techniques. It has enabled the creation of calculation programs for sequence analysis, the creation of specialized and integrated database structures and the provision of IT tools such as software, hardware, and algorithms for data organization and analysis [14].

It is critical to comprehend the structure of the gene or protein sequence after it has been obtained. The term *folding* refers to the process by which proteins are molecularly folded into their final three-dimensional structure starting from its primary structure. It's crucial to know a protein structure in order to comprehend its role in biological processes [39].

The structure of a protein can be determined using a variety of traditional approaches. X-ray crystallography [87] provides very detailed information on the atomic arrangement within a protein. Electronic microscopy [174] studies these complexes in their physiological environment. Nuclear magnetic resonance spectroscopy (NMR) [24] allows characterizing the structure and dynamics of biological macromolecules with atomic resolution. These methods are among the most commonly used techniques. There are some experimental procedural issues with these approaches. This is the case with X-ray crystallography, which involves forcing structures into crystals that do not necessarily represent accurate representations of proteins in their active conformation. Furthermore, the variability of experimental conditions (temperature, concentration, presence of solutes and cofactors and so on) makes this passage long and difficult, and it is a limiting factor in the application of this technique. NMR structures are not as precise as those obtained with X-rays, but they do utilize proteins in solution, in their natural habitat. The dimension of the residues studied, which cannot exceed 300, is the limiting element in this scenario. As a result, these approaches can be costly, labor-intensive, time-consuming and not always feasible.

2.4 CLASSIFICATION METHODS

During this thesis, various classification approaches are employed throughout for melanoma and proteins classification, which we shall discuss in more detail in the following sections.

2.4.1 *Machine learning algorithms*

Machine learning is a growing field of computational algorithms that aims to mimic human intelligence by learning from their surrounding environment and generalizes the results in tasks still

not seen. Pattern recognition, computer vision, spacecraft engineering, finance, entertainment, computational biology and many other fields have all benefited from machine learning techniques. The performance of traditional Machine Learning algorithms is highly dependent on the data representation provided to them during the training phase, which is derived from a set of characteristics extracted for the specific executed task [17].

We have three main learning paradigms, indicated below:

- *Supervised Learning*: in presence of a set of learning data, which form the *training set*, with relative outputs, the network can use the training set in order to learn to infer the relationship between related inputs and outputs. Subsequently, the network is trained by an appropriate algorithm, which uses data to modify the weights and other parameters of the network to minimize the prediction error. The network is able to recognize the relationship that links the input data and output data, and it is able to make predictions on unknown data, having an adequate ability to generalize. This paradigm is used for regression or classification. The most common used supervised learning methods are decision trees, Support Vector Machines (SVM), Naive Bayes;

- *Unsupervised Learning*: a learning of this type uses training algorithms which modify the weights of the network by referring only to the presence of input data. These algorithms attempt to group the incoming data and group them in appropriate clusters. These algorithms learn few features from the data, using topological or probabilistic methods. This approach is mainly used for clustering and feature reduction. The most famous unsupervised learning are Principal Component Analysis (PCA), K-Means Clustering and Self-Organizing Maps (SOM);

- *Semi Supervised Learning*: this paradigm draws its basis from both the supervised and non-supervised learning. Semi-supervised models aim to use a small amount of training data labeled together with a large amount of input data without label. This often occurs in real situations where

data labeling is very expensive and we can obtain a constant flow of data.

2.4.2 *Deep Neural Networks*

This research focuses on a subset of Machine Learning models known as Convolutional Neural Networks in order to classify images. We give an overview about this method.

Artificial Neural Networks (ANN) are mathematical models inspired by the biological neurons of the human visual cortex. In an ANN, an artificial neuron is the essential unit for processing information. [83]. ANN with multiple layers are referred to as *deep neural networks* (DNN). *Convolutional Neural Network* is one of the most often used deep neural networks in image classification, object detection, semantic segmentation and so on [108]. A CNN, as depicted in Figure 2.3 [7], has a hierarchical structure consisting of:

- *input layer*, connected to the pixels of the image;
- *intermediate layers*, in which there are three repeated layers: *convolutional*, *pooling* and *RELU* layer;
- *fully-connected layers*, which represents the completely connected layer acting as a classifier;
- *output layer* that processes the output class as result.

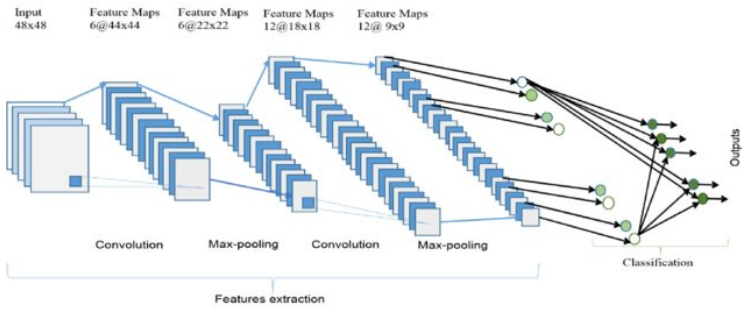


Figure 2.3: CNN architecture.

The primary goal of training a neural network is to optimize the synaptic weights of each layer in order to determine the

parameters that allow the network to achieve the desired mapping between inputs and outputs. A number of filters, known as kernels, are slid over the input images in each convolutional layer to build a number of feature maps. As a result, this technique separates the image into numerous overlaid fragments, which are then examined to determine the unique characteristics of the images before being transferred to the next layer. Convolutional layers can be thought as feature extractors that learn the typical representations of the images they receive. Each neuron in a feature map has a receptive field that is related to the previous layer's neighborhood of neurons through a set of training weights. All neurons in a feature map have weights that are the same. In this way, different features maps in the same convolutional layer have different weights and the essential characteristics of the image can be recovered at each position. A two-dimensional kernel is used to implement the convolution operation in the simplest situation of images with gray levels. The filter will be three-dimensional in the case of a color image stored by a combination of the three RGB (Red, Green, and Blue) basic colors. A two-dimensional activation map can be created by doing the sum element for element along the depth of the obtained volume.

The pooling layer is used to aggregate the properties of the previous layer by *downsampling* them into features of smaller size. As a result, the goal is to reduce dimensionality while keeping the most relevant discrimination information. There are two typical operations for pooling: the *max pooling*, that considers the maximum input from a fixed region, called window, and the *mean pooling*, that takes the average of the window inputs.

The RELU layer employs a *sigmoid* activation function, which is difficult to implement in deep networks due to the *vanishing gradient* problem [100]. The sigmoid function has a lower derivative than 1 and this helps to reduce the gradient values in distant levels from the output. As a result, it saturates as it moves away from 0 and the gradient is neutralized in the saturation regions. The derivative hence has a value of 0 for all negative or null values and 1 for all positive values. This sigmoid tendency results in scattered activations with portion of neurons which are off throughout the network, which can help to increase the

robustness of the network (lower *overfitting*). One of the most typical issues with deep neural networks is overfitting. In certain circumstances, networks store training data rather than learning characteristic features from it, reducing its capacity to generalize.

The fully-connected layer employs a *softmax* activation function, which conducts a normalization for each neuron k, with output values ranging from 0 to 1 and interpreted them as a probability. *Cross-entropy* is used as a cost function between two discrete p and q distributions and, fixed p, measures the difference between the two distributions.

It is required using data that is not included in the training set to evaluate generalization capacity of a model. Generally, datasets are divided into three portions:

- *training set*, typically 80% for network training;
- *test set* in order to evaluate the performance of the trained network;
- *validation set*, used to search for better hyperparameters.

It is tough to construct these three datasets when the training data have small sizes. As a result, the *K-fold cross-validation* technique is employed, which divides the dataset into k equal-sized subgroups. K training cycles are completed as a result of this method. Each fold is then used once as a validation, while the k-1 remaining folds form the training set.

2.4.2.1 *Transfer Learning*

The training of complex CNNs on very large datasets is an expensive time process. *Transfer learning* is a method in which a model created for one task is utilized as the basis for a model for a different task. It is a popular approach in deep learning where pre-trained models are used as the starting point on computer vision and *natural language processing* (NLP) tasks. A transfer learning problem can be solved using a variety of methodologies and implementations. In particular, this technique can be used when a network trained to solve problem A is re-used to solve a new problem B that must be related to problem A. As an example, we can consider the network N trained to classify whether an

image contains or not a car: following the working hypothesis of TL, we could re-use the same network to check if an image contains or not truck. In general, when TL is used, there is the possibility of performance degradation: to increase performance, we could perform an additional step called *fine-tuning*. Fine-tuning is the action related to re-train a pre-trained network by using the pre-trained weights into network filters instead of using random values. In that case, only eventual additional filter added to the network will start with random weights.

Most homogeneous transfer learning solutions use one of three general strategies: attempting to correct for the source's marginal distribution difference, attempting to correct for the source's conditional distribution difference, or attempting to correct both the marginal and conditional distribution differences. The bulk of heterogeneous transfer learning systems rely on aligning the source and target domain input spaces under the premise that the domain distributions are the same [191].

Deep Learning algorithms, particularly CNNs, have become increasingly significant in clinical practice, particularly in bio-imaging analysis.

2.4.3 *Performance measures*

In order to systematically evaluate various classifiers, different metrics are used. The chosen metrics are described in the equations below (Equations 2.2-2.6).

$$Accuracy = \frac{TP + TN}{TN + FP + FN + TP} \tag{2.1}$$

$$Sensitivity(TPR) = \frac{TP}{TP + FN} \tag{2.2}$$

$$Specificity(TNR) = \frac{TN}{TN + FP} \tag{2.3}$$

$$Precision(PPV) = \frac{TP}{TP + FP} \tag{2.4}$$

$$FDR = \frac{FP}{FP + TP} \tag{2.5}$$

$$FPR = \frac{FP}{FP + TN} \tag{2.6}$$

$$FNR = \frac{FN}{FN + TP} \tag{2.7}$$

where TP, TN and FP are the numbers of properly predicted true positives and true negatives, respectively and FP and FN are the numbers of incorrectly predicted false positives and false negatives. We have reported the metrics most used in this contribution. *Accuracy* indicates the degree to which a quantity's measured value matches its true value. *Sensitivity* or *True Positive Rate* (TPR) is a measurement of how well a test can detect true positives. *Specificity* or *True Negative Rate* is a measure of how well a test can detect true negatives. *Precision* or *Positive Predictive Value* is a statistical measure which indicates the proportions of true positive values in a test. When conducting multiple comparisons, this is a means of conceptualizing the rate of type I errors in null hypothesis testing. When performing multiple comparisons, *False Discovery Rate* identifies the rate of type I error in null hypothesis testing. During the verification of a statistical hypothesis, a type I error arises when the hypothesis nothing, which is actually true, is incorrectly rejected. Type I errors are sometimes known as "false positives" because they occur when a positive effect is detected when it is not actually present. *False Positive Rate* (FPR) and *False Negative Rate* (FNR) are the percentage of all negative results that result in positive test outcomes and the proportion of positives which yield negative test outcomes with the test, respectively.

2.5 DEEP LEARNING FOR MELANOMA DETECTION

Early detection of melanoma is still a limiting issue for first-line therapy in this tumor pathology. The presence of regions that

can display anatomical-morphological characteristics that are extremely similar to those of a benign nevus can make early detection of melanoma difficult. Visually sifting images is a challenging task and there is a good possibility that mistakes they can be made in the evaluation. In fact, diagnostic images frequently contain noise and the contrast between tissues may not always be sufficient for a clear interpretation.

Computer-Aided Decision systems (CAD) that operate as assistance for clinical decisions are currently one of the most prevalent lines of action for using acquired melanoma biomedical information and facilitating diagnosis. The adoption of a computerized system, characterized by excellent reproducibility and stability that supports the dermatologist, can result in a faster diagnosis and a higher standard of accuracy. This enables for the automatic detection of melanoma using IT systems that take lesion images as input and output with a melanoma or non-melanoma diagnosis.

The melanoma detection process workflow consists of the following five computationally expensive basic steps:

1. a pre-processing phase with removal of all artifacts from the images;
2. segmentation of the lesion in order to separate the melanoma from its background;
3. a post-processing phase to further improve the image quality of images;
4. a phase of selection of clinical characteristics for the recognition of melanoma on the basis of dermatological guidelines and on Computer Vision techniques;
5. classification phase with a validation step and a test step performed in order to measure the model performance.

2.5.1 *Pre-processing*

Pre-processing is the initial step in the image processing process. Noise reduction, shutting and opening, increasing or decreasing

operational degree of contrast and saturation and so on are common steps in order to improve the quality of image. It is possible to apply image pre-processing [182] to the original image or post-processing to the segmented lesion, or both. The main aim is to remove some typical artifacts such as dark corners, marker ink, gel bubbles, color chart, ruler marks and skin hairs. Many approaches can be utilized at this level: median filtering can be used for noise reduction and smoothing, histogram adjustment, color correction and contrast enhancement methods, border expansion and region merging [135].

2.5.2 *Lesion segmentation*

Lesion segmentation is the first step in automating melanoma detection and it is also the most crucial. At this stage, the lesion is isolated from its background (i.e., skin) and other artifacts. As a result, segmentation is a procedure that divides an image into meaningful regions (separating foreground and background), which are useful for analyzing and recognizing an object in the image. For lesion segmentation, several approaches have been developed: thresholding [25], clustering [122], fuzzy logic [11], graph theory [200], deep learning based approaches [10, 35, 36, 116] and combination of these methods [135].

2.5.3 *Clinical features*

A typical melanoma case can be recognized once the lesion has been segmented by looking for numerous clinical features that may exist in the segmented region. These characteristics can be global or local. Local features appear on a single area or a group of spots on the lesion, whereas global features exist all across the lesion. *Texture*, *Shape* and *Color* are the three basic categories in which these clinical aspects can be classified. Different feature selection techniques can be applied. The statistics of the gray-scale version of the input dermoscopic image, for example, can be used to detect texture features. Clustering algorithms can be used to extract color features [135].

Dermatoscopy, which allows the observation of patterns not visible to the human eye, is the most commonly used and non-invasive approach for the early identification of melanoma [20].

When evaluating a suspected melanotic lesion, the ABCDE rule is an appropriate monitor control. **A** represents the symmetry of the lesion. **B** indicates its regular or irregular border. **C** captures the colors and the **D** is the diameter of the lesion. These properties of a lesion are all explained by this basic dermatological guideline. The **E** in this case stands for *evolution*, which refers to signals that the lesion is rapidly expanding. The evolution is widely acknowledged as the most distinguishing feature for early diagnosis, but it is also the most difficult to quantify. Follow-up with the patient over time can help identify lesions that are described as *featureless*, without atypical dermoscopic criteria, which could be false negatives. [152].

The Menzies technique is based on a set of 11 characteristics that are either present or absent. This approach employs *negative* and *positive* characteristics. Negative features are symmetry of pigmentation pattern and the single color (black, gray, blue, red, dark brown and tan). Because malignant melanocytes generally retain cellular melanin and can be found at different depths in the skin, melanomas usually appear in multiple colors. Positive features are: blue-white veil, multiple brown dots, pseudopods (foot-like projections present at the edge of a lesion), radial streaming, scar-like depigmentation, peripheral black dots/globules, multiple colors, multiple blue-gray dots and broadened network [124]. To diagnose melanoma, at least one of the nine positive features must be present and none of the two negative features must be present.

Another methods dermatologists use to evaluate lesions is the 7-point Checklist. This method uses *minor* and *major* criteria for the melanoma detection. Major criteria are: atypical pigment network, gray-blue areas and atypical vascular pattern, all with scores equal to 2. Minor criteria have score equal to 1 and they are: radial streaming (streaks), irregular diffuse pigmentation (blotches), irregular dots and globules and regression pattern. The ability of each criterion to increase the likelihood of a positive melanoma diagnosis is measured using odds ratios. The odds ratio is used to calculate the score for criterion existence. A total

score of 3 is necessary for the diagnosis of melanoma based on the simple adding of the criterion scores [184].

Many studies in recent decades focused on the use of Deep Learning and, in particular, CNNs are used for the classification of melanoma.

Yu et al. [198] have proposed a two stage approach with a very deep fully convolutional residual network (FCRN) with more 50 layers and residual learning technique to overcome the degradation problem. They first have segmented and then classified lesions in melanoma/not melanoma output. The used dataset is a public challenge released with ISBI 2016. In the proposed FCRN is incorporated a multi-scale contextual information integration scheme. Then, for the classification stage, Yu et al. have explored Softmax classifier and SVM classifier in order to obtain their average predictions value. The proposed method was implemented with C++ and Matlab based on Coffee library. They have compared the performance of FCRN with different depths, in particular 38, 50 e 101, the VGG-16 network and fully convolutional GoogleNet. The better accuracy (ACC), sensibility (SE) and specificity (SP) is reached by FCRN-50. They have also established the importance of the multi-scale integration scheme to retrieve local image information and that the fusion of the simple average performance of SVM and Softmax can improve the classification performance.

In 2018, Li and Shen [116] improved deep learning network for segmentation, feature extraction and classification for skin lesion analysis. Using the ISIC 2017 dataset [1], they proposed two deep learning frameworks: the Lesion Indexing Network (LIN) and the Lesion Feature Network (LFN), inserting more internal convolutional layers and an extra residual link in ResNet. They obtained 85.7%, 49% and 96.1% for accuracy, sensitivity and specificity, respectively.

In 2018, Yu et al. [199] introduced a deep residual neural network, ResNet, consisting of a set of residual blocks, composed of several stacked convolutional layers. They chose this architecture because residual links can speed up deep network convergence while maintaining accuracy advantages obtained by significantly increasing network depth. They also used a local descriptor en-

1 https://www.isic-archive.com

coding strategy. Deep representations of a rescaled dermoscopic images are recovered first using the deep residual neural network. Then, using orderless visual statistic features based on Fisher vector (FV) encoding, these local deep descriptors are combined to provide a global image representation. Finally, using a support vector machine with a Chi-squared kernel, the FV encoded representations are employed in order to classify melanoma images. They reached 86.81% of accuracy. For performance comparison, Yu et al. explored many alternative CNN models with varying depths, including 8-layer AlexNet, 16-layer VGGNet (VGG-16), and considerably deeper 101-layer ResNet (ResNet-101).

In 2019, Albahar explored the use of a deep CNN with a novel regularizer technique for skin classification, based on the standard deviation of the weight matrix of the classifier. In particular, this regularizer penalizes the dispersion of the weight matrix values. The used dataset was taken from ISIC archives, which contains 4533 malignant and 19373 benign skin lesion images. This network uses pooling and dropout layers that follow two convolution layers in this architecture. The term *dropout* refers to units in a neural network that are no longer active (both hidden and apparent). This filter is used to prevent overfitting in particular. During training, units (and their connections) are dropped randomly from the neural network. This action should keep units from over-adapting to one other [167]. In addition, a new hyperparameter is introduced, which determines the likelihood of the layer's outputs being dropped out or, conversely, the probability of the layer's outputs being maintained. After dropout, the 2-D outputs are flattened in a 1-D array and fully coupled with the following layer, which has 128 neurons. Each class has one output neuron in the last layer. The innovative regularizer is integrated in each convolution layer [5]. This CNN reached an accuracy of 97.49%, a sensitivity of 94.3% and a specificity of 93.6%.

Fujisawa et al. [58] used a GoogLeNet DCNN model architecture trained on a dataset of clinical images with malignant melanoma (MM), squamous cell carcinoma (SCC), Bowen disease, actinic keratosis, basal cell carcinoma (BCC), naevus cell naevus (NCN), blue naevus, congenital melanocytic naevus, spitz naevus, sebaceous naevus, poroma, seborrhoeic keratosis, naevus

spilus and lentigo simplex. In particular, there are 540 malignant melanoma images in total and they reached an accuracy of 72.6%.

Kawahara et al. [94], have proposed a multitask deep convolutional neural network trained on multimodal data (clinical and dermoscopic images and patient metadata). They used the 7-point melanoma checklist criteria. Their neural network generated multimodal feature vectors for image retrieval and identification of clinical discriminant regions, using numerous multitask loss functions, each of which takes into account distinct combinations of input modalities.

Sarkar et al. [158], presented a model of neural network with two main methodologies: residual learning and depthwise separable 3D convolution, which is faster and needs less parameter space, proving itself a more effective alternative to its traditional counterpart. On a dermoscopic dataset with 4000 dermoscopic images (1950 of melanoma images and 2050 benign images), they reached an accuracy of 99.5%, a sensitivity of 99.3% and a precision of 99.6%.

Zhang et al. [201] have constructed an attention residual learning convolutional neural network, called ARL-CNN, in order to avoid the problem of little data available, the interclass similarity and intra-class variation. They based their network on an attention mechanism capable of increasing the possibility of discriminating the information available by focusing on their semantic meaning. No new extra learnable layers are introduced in the network, but the possibility of grasping the semantic meaning is delegated to the more abstract feature maps of the higher layers. For the experiments, dataset ISIC 2017 was used and 1320 additionally dermoscopy images, including 466 melanoma. The proposed ARL-CNN network, consisting of 50 layers, in the melanoma classification, has obtained an ACC of 85% a specificity of 89.6% and a sensitivity of 65.8%.

In 2020, the study of [89] used a dataset of more than 12000 skin images between malignant and benign tumors, from which they extracted 5846 clinical images of pigmented skin lesions from 3551 patients. The dataset contains 1611 malignant melanoma images. A faster, region-based CNN (FRCNN) model was chosen because it consistently demonstrated good classification accuracy, robustness, and speed. The authors consider the accuracy of the

classification of the FRCNN model overall for the prediction of six classes, two malignant (malignant melanoma and basal cell carcinoma) and benign tumors (nevus, seborrheic keratosis, senile lentigo and hematoma / hemangioma). They achieve an accuracy of 86.2%. The accuracy, sensitivity, and specificity for two-class classification (benign or malignant) were 91.5 % 83.3 % and 94.5%, respectively.

In 2020, we presented a comparison of neural network approaches for melanoma classification [56] on HAM10000 (Human Against Machine with 10.000 training images) dataset [175], where we compared three different neural networks, with and without pre-processing: 2D-CNN, ResNet and Self-Organizing Map (SOM).

Approach	Accuracy	Sensitivity	Specificity
2D-CNN	71.9%	69%	**92.8%**
ResNet	**79%**	**80%**	78%
SOM	66.8%	61.7%	63.5%

Table 2.1: Results without pre-processing.

Approach	Accuracy	Sensitivity	Specificity
2D-CNN	74.1%	**89.4**%	72.1%
ResNet	**81.5%**	85%	**79%**
SOM	69%	64%	68%

Table 2.2: Results with pre-processing.

The result are reported in Table 2.1 and Table 2.2. The ResNet network shows greater accuracy on the dataset than the other competing approaches. In particular, it is evident that the preprocessing increases the accuracy up to 2.5% for the ResNet.

2.6 DEEP LEARNING AND MACHINE LEARNING FOR PROTEINS CLASSIFICATION

Because of the need of determining the structure of a protein in order to understand its function, significant resources and en-

ergy have been devoted to the development of IT approaches for predicting protein structure that can guide traditional methods of research. The fundamental advantage of computational techniques is that they are generally simple and quick, allowing them to avoid the time-consuming and tedious experimental processes required for protein structure determination.

We can consider comparative or homology modeling as one of the most prevalent computational methods. It is used when there are known structures, defined *templates* with sequences comparable to the *target* protein, with the unknown structure. In this case, the method is *template based*, and it is based on the concept that, during evolution, structures are most preserved in comparison to primary sequences of proteins, which might change over time. So, proteins with a good level of sequence similarity are also structurally equivalent [103]. When a target protein does not have a strong sequence similarity to a protein with a known structure, a method called as *fold recognition* is used. The core idea is that a protein has a finite number of folds it can assume. In this case, the two most used approaches are those based on structural profiles and so called threading [54].

Profile-based approaches are based on the possibility of deducing some properties features for each amino acid from protein structure analysis [54]. In *threading* methods, many possible protein models are generated utilizing a reference (template) structure of a known protein and a vast number of possible alignments. The best models are picked from among these obtained by doing energy evaluations on the structures [154].

Different bioinformatic tools are developed from these different approaches for folding recognition, making it possible for researchers to predict the structure of a protein starting with its sequence [73].

Proteins have a variety of functions in the cell. All proteins play a role in the operation of biological systems in a broad sense. The goal of the post-genomic era is to learn more about the molecular mechanisms that control the biological activity of all the proteins encoded by each sequenced genome. In this direction, one of the most successful research subjects has been the prediction of the 3D structure of proteins. This problem, like any other, can be broken down into subproblems. In order to resolve the

protein structure, in this thesis, we focused on determining one of these subproblems in this thesis: the classification of the torsional angles of the proteins backbone.

Following the advancement of sequencing, a collection of protein sequences is compiled. The torsional angles (ϕ and ψ angles) of proteins can be determined using a variety of experimental and computational techniques. This vast amount of data is fed into systems that use automated learning computational models.

In bioinformatics and computational biology research, deep learning and machine learning have become highly popular. Nowadays, we are witnessing an important exponential increase rate of biomedical data from various sources. We can count data from patients level such as electronic medical records, to micro molecular level (gene functions, protein interactions, etc.), available through experimental studies or data acquired by technologies. These heterogeneous data are sent to ML and DP, which create predictive models in medicine and health care [4]. Protein data, like all sorts of data, has missing values, noise and anomalous values that compromise the validity of the data. As a result, a data cleaning process must be performed prior to their use. In order to merge data from numerous sources, data integration algorithms are frequently required. Through smooth aggregation, data generalization and normalization, data can be turned into a specific form that is ideal for mining. The substitution of a latent pattern distribution shape or connection for the sake of computational ease is the nature of transformation. As a result, more powerful theoretical techniques and practical tools for evaluating and extracting useful information from the complex biological Big data outlined above are required [106]. One of the major challenges is deducing the properties correlated to function, secondary structure and 3D structure of a protein from its amino acid sequence (primary structure). This received a huge boost as a result of technological advancements in sequencing, which have made a lot of protein data with more unknown features available. Protein classification tasks can be based on protein feature discovery, as well as the analogy of amino acids to words, protein domains to sentences and proteins to text paragraphs [168]. These last methods are often *annotation-based*, where there is the use most homology information from

different types of annotation, including Gene Ontology [65]. Appropriate descriptors for the protein sequences are required for the application of machine learning and deep learning to protein classification. Some of these descriptors have already been successfully applied to sequence-based protein classification [34]. Among these, we can consider: amino acid composition (dipeptide, tripeptide composition), predicted secondary structure and predicted solvent accessibility, k-Spaced Amino Acid Pairs and Conjoint Triad, other physio-chemical features [95]. Feature representation remains a research challenge that necessitates the use of customized labels [188].

Many of the new computational approaches are utilized to solve two fundamental problems: protein classification and prediction of a protein's fold from its sequence. Different features of the latter can be investigated. In this thesis, we focused on the prediction of the torsional angles of the protein backbone. The challenge of predicting the secondary structure of a protein include the prediction of these angles, which are likewise described as dihedral angles (indicated as ϕ and ψ). Permitted protein conformations are generated by specific combinations of these angles in each secondary structure. Early techniques of protein classification depended on pairwise sequence comparisons, which were based on sequence alignment [131] and used exhaustive dynamic programming approaches or heuristic algorithms [181]. More recently, deep learning has demonstrated the ability to acquire valuable feature representations from input data, and it is extremely useful for describing linear, nonlinear and complex interactions. In particular, we can note some common classification techniques, such as k-nearest neighbor (KNN) [96], Naïve Bayes (NB) [166], decision tree (DT) [127], support vector machine (SVM) [133], neural network (NN) and ensemble (EM) [106] and Hidden Markov Models (HMM) [47].

3

MELANOMA DETECTION

In this Chapter, our contribution to the melanoma detection search lodging is proposed. We concentrate on two important themes in particular. The first work is based on the use of smart tools for health that are driven by the so-called *Internet of Medical Things* (IoMT), which is a collection of integrated medical data that has to do with both individual health and, in a broader sense, health organizations. This work contributes to a Smart Healthcare that is becoming increasingly connected, in which all data, particularly the *Electronic Health Record* (EHR), represents digital information that allows the AI to construct prediction models capable of identifying diseases. So, we start with an overview about the early diagnosis of melanoma in Section 3.1 and the used classification methods 3.2. Then, we describe the dataset and training options 3.3, the proposed design of the hybrid architecture 3.4 and our research questions 3.5. Experimental results of this work follow in Section 3.6.

The second focus is on using genetic algorithms in combination with neural networks in order to identify melanoma. Despite the abundance of diagnostic technologies currently in use, we always start with the hypothesis of work that early melanoma detection remains an open challenge. As a result, experimenting with new ideas in this field can help to improve predictive models.

3.1 STATE OF THE ART

Melanoma is a type of skin cancer that arises from melanocytes, the epidermal cells responsible for the synthesis of melanin pigment. Despite the fact that this type of tumor accounts for a small percentage of all cutaneous malignancies, it is the leading cause of death [49].

Melanoma of the skin has increased rapidly in the last 30 years, however the trends differ according to the age group. Between

2007 and 2016, the rate for those under 50 years old declined by 1.2% each year, while the rate for those 50 and up grew by 2.2%. According to the American Cancer Society, 100350 new cases and 6850 deaths in both sexes were estimated in 2020, only in the United States[1].

Early detection of melanoma is well established in the literature, yet it is still a difficult task [2]. The physician's ability to distinguish between different forms of skin lesions is also dependent on his level of experience. A biopsy is still required to give the final word on a poor diagnosis. Early detection of melanoma is becoming increasingly important, especially in those who have a high risk of acquiring cancer, as it allows for a higher cure rate. In most cases, a dermatologist's first visual assessment, typically with the help of polarized light magnification dermoscopy, is utilized to diagnose melanoma in clinical practice [26]. Technology has the potential to transform the way we think about medicine, while also playing a crucial role in sophisticated diagnostics systems that make judgments that are critical to patient care [125]. Simultaneously, as technology advances, the number of intelligent devices connected to the Internet capable of creating large amounts of data has increased exponentially. This parallelism might be seen in dermatology, with the possibility of using basic devices like cellphones to take clinical photos and as sensors for remote skin abnormality screening [85]. In recent years, a variety of computer software has been created with the goal of assisting dermatologists in better (and faster) determining if a skin lesion is, is not or could become a melanoma [71, 160]. The majority of this software is based on computer vision techniques such as boundary detection, symmetry/asymmetry analysis, color analysis and dimension detection [71]. Other types of information, such as EHR, are also used by some technologies to improve prediction accuracy. Overall, existing melanoma detection methods must account for the complexity of the images to be processed, which may result in challenges such uneven fuzzy lesions boundaries, noise and artifact presence, low contrast, or poor image lighting [3].

1 https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/annual-cancer-facts-and-figures/2020/cancer-facts-and-figures-2020.pdf

The following is a summary of how these systems are constructed. The initial step is to download or construct an image dataset with melanoma and non-melanoma images. Dermoscopic, clinical and histological images are available: the first are more detailed, but the need for a dermoscopy may limit the dataset's size; the second are less detailed, but more readily available; and the third are related to the highest image resolution. The images are then processed using one or more Computer Vision and image processing algorithms to extract features. These characteristics will serve as training inputs. Unfortunately, the prior workflow may have at least two significant flaws. The first is related to the amount of storage space and computational power required to train a sophisticated model on massive datasets and obtain satisfactory results. To converge, the pure segmentation approach based on k-means clustering, for example, may require a Running Time (RT) of $O(2^n)$ [63]. When the amount of data increases, we need more computing power and better technology (such as GPUs) to get the results of some computations in human time.

The second disadvantage is the time and effort necessary to maintain one or more models. There are no simple ways to update and improve the performance of a model once it has been trained and deployed without going through the training process again. It's worth noting what happened in the ISIC 2019 Challenge: the performance of the winning algorithm in ISIC 2018 dropped from 88.5% to 63.6% only due to the addition of new categories and images [71]. The authors proposed three reasons for this finding:

- the quality of the images and the training dataset structure (balanced/unbalanced) which affect the performance of the deep learning techniques;
- intra-class dissimilarities and interclass similarities can have an impact on the performance of the system;
- in order to learn how to discriminate the same item from diverse points of view, deep learning demands that the input undergo a data augmentation procedure (stretched, rotated, illuminated and so forth).

Before network training, the three datasets (training, validation, and test) are frequently fixed. It's possible that the fact that a small change in the subsets can affect prediction accuracy is hidden. This indicates that Transfer Learning, which involves using a pre-trained network to solve additional issues, is still unreliable [141].

In this work, we study how the two drawbacks impact the design and implementation of an AI-based detection system and we propose a three layers architecture that can be used in a real environment and not only in a controlled one. We show how a simple dataset modification can impact the classifier performance, and that a distributed and cooperative system is needed to enable deploying a melanoma classifier usable into the real world.

3.2 CLASSIFICATION METHODS

For the high accuracy scores provided, we used the following neural networks. We downloaded the Google InceptionV3 [171], GoogleNet neural network [171] and AlexNet [84], publicly available. These networks were pre-trained to deal with a wide range of image types. Instead of 1000 classes, we changed the last layers to classify between two classes. This phase entails replacing the original `SoftLayer` and `ClassificationLayer` of the networks with a new layer that has two output classes (melanoma/non-melanoma).

3.2.1 *Related Works*

There have been numerous methods developed for the automatic detection and classification of melanoma. We can count decision trees [203], Support Vector Machines (SVM) [67], logistic regression [173] and Bayesian classifiers [155]. In the image-based detection of a pathology, Convolutional Neural Networks (CNNs) are crucial [85]. Their utility in the detection [130], segmentation, and categorization of melanocytic lesions has been well documented [202].

The study of Haenssle *et al.* [74] reports on a comparison of the performance of dermatologists with that of a widely used convolutional neural network in detecting skin lesions in order

to give a diagnosis conclusion and management decision. Dermatologists were divided into three categories based on their dermoscopy experience (beginner, less than two years; skilled, between two and five years; expert, more than five years) and had access to two levels of information: dermoscopic image (level I) and dermoscopic image with clinical close-up image and textual information (level II). Finally, the study found that dermatologists and convolutional neural networks had identical outcomes when dealing with a broader range of diagnoses. In addition, dermatologists have demonstrated their capacity to synthesize data from a variety of sources in order to give an accurate diagnosis. This information validates the surge in research towards melanoma diagnosis using Machine/Deep Learning techniques.

Simoyan [163] demonstrated how the architecture of the deeper Visual Geometry Group model (VGG), which is based on the learning of models with a bigger number of picture descriptors used as inputs (such as color, symmetry, contour and so on), can provide superior melanoma detection efficacy. The VGGs can also be applied to the search box in question, depending on the blocks and the filter used. The most popular models are VGG 11, 16 and 19, which differ in the number of convolutional layers they contain: 8, 13 and 16, respectively.

The use of AlexNet, an eight-layer convolutional neural network, is demonstrated in [84]: the first five levels were convolutional, some of them followed by max-pooling layers and the last three layers were completely connected. It used the non-saturating RELU activation function, which outperformed *tanh* and *sigmoid* in terms of training performance. The accuracy of the network with these parameters was 96.86%, 97.70% and 95.91%, respectively, utilizing Transfer Learning and the data augmentation approach, testing and verifying it on the three MED-NODE, Derm (IS-Quest) and ISIC datasets.

GoogleNet is a deep convolutional neural network made up of around 100 different types of building blocks, including convolutions, average pooling, max pooling and contacts. This network is based on the primary Inception architecture, which was first introduced in 2015 as a computationally efficient network that could run on constrained resources. Google Cloud Platforms offers GoogleNet executions on Cloud TPU [171].

Esteva *et al.* compared the performance of the Google InceptionV3 network in skin cancer classification to the knowledge of 21 dermatologists, indicating how the network outperformed specialists in this endeavor [49]. Google InceptionV3 is based on the Inception Architecture's advancement. It is a widely used image recognition model that has been found to achieve higher than 75% accuracy on the ImageNet dataset. Convolutions, average pooling, max pooling, concerts, dropouts and fully linked layers are among the symmetric and asymmetric building blocks of Google InceptionV3. In this network, the batchnorm filter is applied to activation inputs and is used extensively throughout the model: batchnorm pruning goal is to find and eliminate irrelevant filters from the CNN to make them more efficient without sacrificing performance by finding information that can help establish how significant or useful each filter is concerning the final output of neural networks. Finally, Softmax is used to calculate loss [171].

3.3 DATASET AND TRAINING OPTIONS

The used dataset is MED-NODE, presented in computer-assisted system for melanoma diagnosis [68]. It consists of 170 clinical photos (70 melanoma images and 100 nevi images) from the Department of Dermatology's digital image repository at the University Medical Center Groningen (UMCG). These are clinical images taken with a Nikon D3 or Nikon D1x camera and a Nikkor 2.8/105 mm microlens, with an average distance of roughly 33 cm between the lens and the lesion in 95% of the images in the collection. An example of MED-NODE images are depicted in Figure 3.1 and Figure 3.2.

Dermatologists double-checked each photograph to ensure it was appropriately labeled. The photos were taken from a variety of Caucasian patients and have been anonymized and preprocessed. Hair removal has already been accomplished using the Dullrazor software [109].

The Otsu method was used for the segmentation process, which can minimize intra-class variance separating the two classes (melanoma and non-melanoma) [139].

Otsu segmentation was performed using the following code:

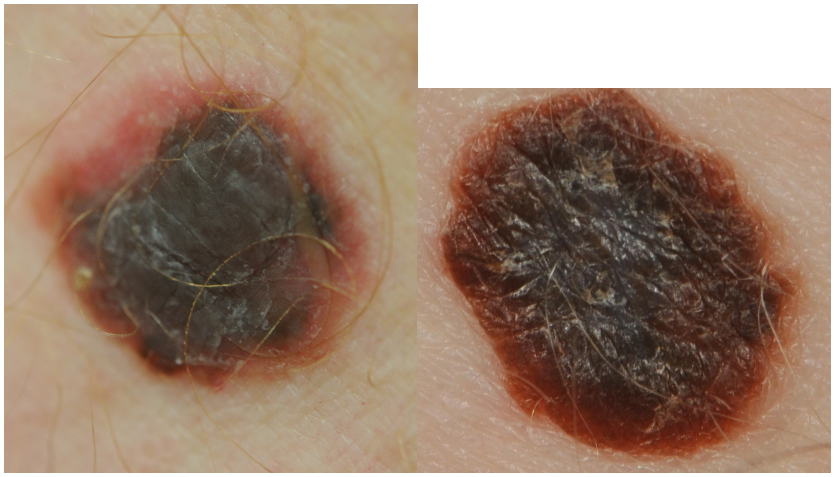


Figure 3.1: Melanoma images in MED-NODE dataset.

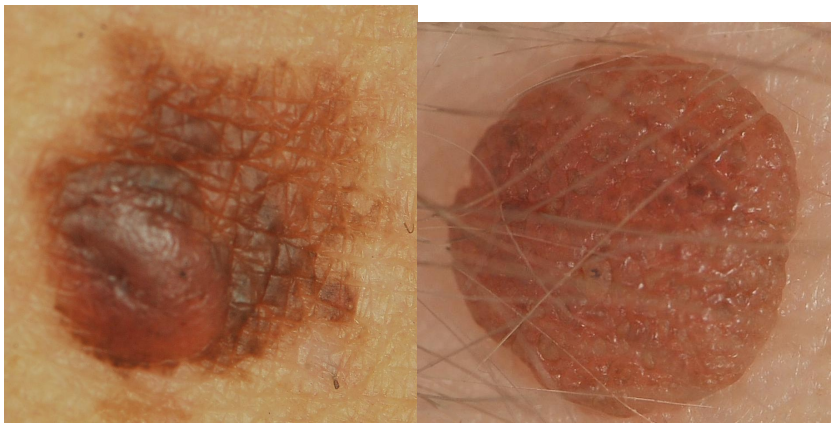


Figure 3.2: Nevi images in MED-NODE dataset.

```
[input_image,map] = imread(F);
bw_input = rgb2gray(input_image);
[T, EM] = graythresh(bw_input);
BW = imbinarize(bw_input, T);
mask_otsu = BW;
mask_otsu= ~mask_otsu;
new_image = input_image * mask_otsu;
```

In order to avoid the problem of unbalanced dataset, we chose the data augmentation technique [162] to gain additional variations through artificial alterations of the images because the dataset we used only had 170 total images (70 malignant and

100 benign). Data augmentation was performed by using Matlab `imageDataAugmenter` object, with the following configuration:

```
'RandRotation', [-180, 180], ...
'RandScale', [1, 10], ...
'RandXTranslation', [-180, 180], ...
'RandYTranslation', [-180, 180]
```

To decrease noise, a Gaussian filter was applied before each network training: the *imgaussfilt* function was employed with a dynamic sigma value between 1 and 7 [13].

The training options were the following, with 30 epochs, a N initial learning rate of 10^{-4} and the *stochastic gradient descent with momentum* (SGDM) [120]:

```
('sgdm', ...
'MaxEpochs', 30, ...
'MiniBatchSize', 12, ...
'Shuffle', 'every-epoch', ...
'InitialLearnRate', 0.0001, ...
'Verbose', true, ...
'ValidationData', imdsValidation, ...
'ValidationFrequency', 1, ...
'VerboseFrequency', 1, ...
'Plots', 'training-progress',
    'ExecutionEnvironment', 'gpu')
```

3.4 THE PROPOSED DESIGN OF A HYBRID ARCHITECTURE

The proposed hybrid architecture for the melanoma detection is divided into:

- *Edge layer*: consists of all smart devices (Edge Devices) of the IoT architecture. At this level, the data are processed by the edge device (smartphone) or transmitted via a local server to the Fog Layer;
- *Fog layer*: includes server systems distributed on the network which receive data from the Edge layer, pre-process and upload them into the cloud;
- *Cloud layer* represents the central management level of data from previous levels.

Within the Cloud, data buckets are maintained and systems training is performed. The orchestrator takes care of the distribution of the optimized services after each formation in the Fog area. The offered services are performed in the Fog area. Local calculations on IoTM devices (smartphones) are performed in the Edge area. The generic user uses the services to get the output, and he contributes to the growth of the knowledge base of the system, while loading data.

We may deduce that imagining a distributed architecture could bring a number of advantages to the end user by allowing:

- the collecting and aggregation of data "on the network" to aid in the early detection of melanoma, while also supplementing image databases with additional information;
- processing critical data locally, at the network's edge, with local data storage, resulting in lesser bandwidth, faster data access and reduced data processing delay;
- a huge number and mobility of Fog nodes, as well as interoperability, allow for widespread distribution of resources and computing services.

In the pre-processing and classification of melanoma images, this sort of architecture implementation responds to a fresh need and data management methods that are more advantageous than standard methods. It specifically handles the issue of transferring images for processing to a central data server or Cloud service. Furthermore, decentralizing them increases capacity and, as a result, reduces calculation times.

3.4.1 *Related Works*

In the framework of the IoMT, the first architectures built of Edge, Fog and Cloud resources that facilitate anticipatory learning surfaced in 2017 [22]. A recent research offered an architecture that permits modeling solutions for lung and skin illness classification without confining itself to IoMT data security testing, bringing the possibility of offering flexibility in the adoption and integration of AI techniques [145].

The majority of IoMT data management and analysis approaches in the literature are based on Cloud computing. Individual user data security, resolution in the exchange of medical images, data archiving and the ability to improve diagnosis response times by decentralizing computing power for Machine and Deep learning techniques on network nodes, which are used as microdata center mesh networks, are the fundamental problems that remain unsolved in this field. There is still a scarcity of knowledge on three-layer hybrid architectures that allow for specialized computational operations on melanoma images and the creation of a real-time database, starting with access to more user-friendly equipment such as a smartphone. As a result, this awareness was the driving force of our recent work.

3.5 RESEARCH QUESTIONS

3.5.1 *First goal: Transfer Learning reliability evaluation*

We have conducted two experiments to reach two goals in this work: the first one is to show how changes to the dataset structure can decrease overall system performance. In particular, we want to show that, to address the melanoma detection problem, Transfer Learning is not completely trustworthy if we rely only on final layers fine-tuning and the pre-trained weights provided with the pre-trained network. Small changes can impact the generalization capabilities of the network, causing high-performance degradation. On the contrary, we suggest that a global network, subject to continuous retraining, should be used to address the melanoma detection problem. We ran continuous retraining on three classifiers while slightly changing the dataset's structure. We determine the dataset structure term's training, validation, and test set compositions in this scenario.

For this purpose, we created four new datasets (MDS) as a result of our assumption on Transfer Learning reliability:

1. MD1 - contains MED-NODE original images;
2. MD2 - contains MED-NODE images segmented with the Otsu method;

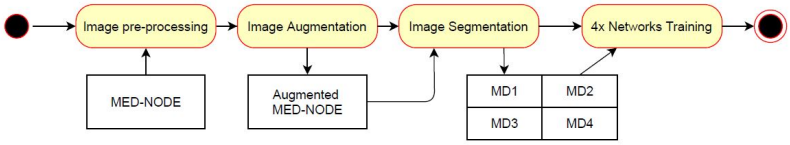


Figure 3.3: The sequential pipeline used in experiment one for performing the continuous retraining.

3. MD3 - contains MED-NODE images and augmented images without segmentation;

4. MD4 - contains MED-NODE images and augmented images segmented with the Otsu method.

The primary hypothesis is that the marginal distributions of the source and destination domain data may differ when evaluating the four datasets generated, but the reference labels are always the same. This experiment was used to collect data on classifier performance and assess the effort (measured in time) required to retrain without using the distributed technique. The system training pipeline is depicted in Figure 3.3. In the setup, we used a single Intel Scientific Workstation with 16 core, 16GB RAM, and one GPU GTX980. This experiment involved the execution of 8400 training steps (each with 30 epochs) on a single workstation environment.

We simulated continuous retraining for each dataset D in MDS by repeating the training phase 700 times. The dataset was split into three parts for each iteration: 0.5 for the training set, 0.3 for the validation set and 0.2 for the test set. With the randomized option enabled, we utilized the splitEachLabel method. The training set then includes 50% melanoma and 50% non-melanoma images, chosen randomly from the starting image collection, for each cycle. Although with different *ratios*, the identical technique was employed in both the validation and test sets.

3.5.2 *Second goal: Impact of the three-layers architecture*

The performance of the architecture was measured in the second experiment. To demonstrate that a distributed and coopera-

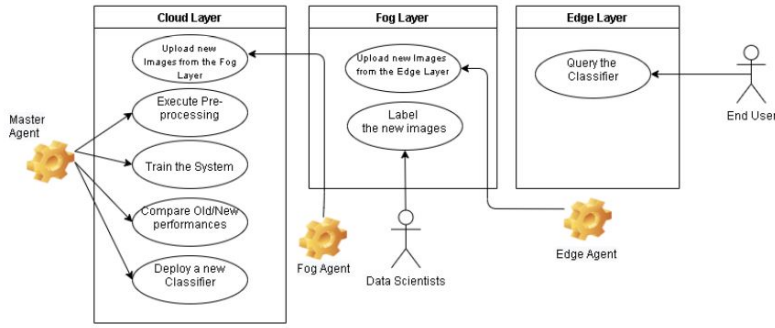


Figure 3.4: The setup of the second experiment simulates a three layers architecture.

tive system is required to deploy a melanoma classifier resilient against Transfer Learning difficulties, we specifically design the architecture to allow automatic classifier retraining and deployment. Our working hypothesis is based on the necessary significant reiteration required to find the optimal classifier if the data structure changes. We created a three-layer design, in which the Cloud layer performs the training and retraining, as we can see in Figure 3.4.

This setup was built with the GRIMD framework [143], allowing us to distribute each iteration upon multiple instances. The GRIMD instances were first deployed on Amazon AWS. After then, the steps of training, retraining, validation, testing and performance comparison were shifted to the Cloud layer. The essential concept is that when a new model is ready, it is only deployed into Fog if and only if it exceeds the preceding one in terms of accuracy. The Layer Agents, which we developed as a simple CROND instance, maintain the synchronization between each layer. First, we isolated the Cloud layer, which is completely unaffected by the classification issue. The Agent layer was then set up to send a new trainer classifier to the Fog layer if and only if the average accuracy of the new classifier outperforms the old. The training of the classifiers is then separated from their execution for prediction purposes. Finally, the classification and prediction functions were transferred to the Fog layer, which houses the web server and trained models. Every end-user in this scenario uses an app that communicates with the Fog layer. This

second experiment involves the same computations as the first one, but it allowed GRIMD to scale up to 128GB and multiple GPU, using Ec2 instance from type t2 (micro-instances: t2.micro) and m5 (balanced computation instances: m5a.2xlarge), to type c6 (optimized computation instances: c6g.16xlarge). Different instance families designed for certain tasks are available on Amazon EC2. T-instances are general-purpose instances type that provide high capabilities in terms of CPUs and medium RAM memory capabilities. The t stands for *tiny* and the m stands for *micro* or *medium*. The c-instances are optimized calculation instance that has a higher CPU to memory ratio. Here, c stands for *compute*. They are used for applying applications for calculation in the high performance computing (HPC), for high-performance analytics workloads, media transcoding and rendering, building complex machine learning models and scientific modeling [114].

In order to computationally optimize instances, we configured the training session as follows, due to the presence of t-instances and c-instances:

```
('sgdm', ...
'MaxEpochs', 30, ...
'MiniBatchSize', 12, ...
'Shuffle', 'every-epoch', ...
'InitialLearnRate', 0.0001, ...i
'Verbose', false, ...
'ValidationData', imdsValidation, ...
'ValidationFrequency', 1, ...
'VerboseFrequency', 1, ...
'Plots', 'none',
    'ExecutionEnvironment', 'auto')
```

3.6 EXPERIMENTAL RESULTS

We used the accuracy to estimate the performance of the three networks (denoted with ACC).

Following the equations in Subsection 2.4.3, we also calculated sensitivity (TPR), specificity (TNR), precision (PPV), false discovery rate (FDR), false-negative rate (FNR) and false-positive rate (FPR), shown graphically in Figures 3.5(a)-3.5(c) with Otsu segmentation and in Figures 3.6(a)-3.6(c) without Otsu.

According to the method in Equation 3.1, we also took into consideration the standard deviation (SD) to calculate, on average, how much the accuracy measures differ from one another:

$$SD = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2} \tag{3.1}$$

where n is the size of the dataset and $\overline{x}$ is $\frac{1}{n}\sum_{i=i}^{n} x_i$ the arithmetic mean of x. In Figures 3.7(a)-3.7(c) are reported the SD values for all three used networks.

The results with and without Otsu segmentation on the MED-NODE dataset have been provided in Table 3.1 and Table 3.2.

WITH OTSU SEGMENTATION

Net	Data Augmentation	ACC (min)	ACC (max)	ACC (mean)	ACC (sd)
AlexNet	None	**0.65**	**0.94**	**0.78**	0.06
	Yes	0.44	0.91	0.68	0.08
Google InceptionV3	None	**0.56**	**0.94**	**0.76**	0.07
	Yes	0.32	0.74	0.53	0.09
GoogleNet	None	**0.60**	**0.91**	**0.75**	0.07
	Yes	0.32	0.74	0.55	0.09

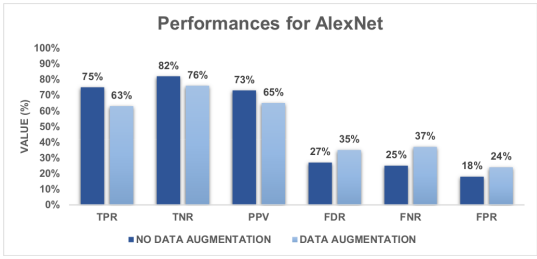
Table 3.1: Performance on MED-NODE dataset for ACCs with Otsu segmentation and with and without data augmentation.

WITHOUT OTSU SEGMENTATION

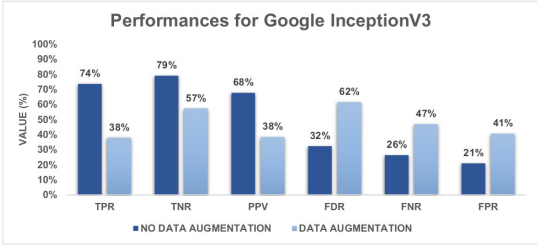
Net	Data Augmentation	ACC (min)	ACC (max)	ACC (mean)	ACC (sd)
AlexNet	None	0.68	**1**	**0.89**	0.05
	Yes	**0.76**	0.97	0.87	0.05
Google InceptionV3	None	**0.56**	**0.94**	**0.74**	0.07
	Yes	0.32	0.71	0.55	0.07
GoogleNet	None	**0.65**	**0.94**	**0.80**	0.06
	Yes	0.30	0.76	0.55	0.09

Table 3.2: Performance on MED-NODE dataset for ACCs without Otsu segmentation and with and without data augmentation.

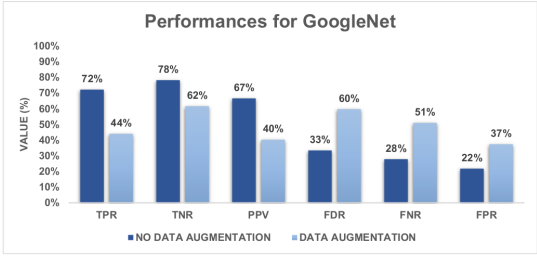
The greatest values attained by the networks in the computations of the average, maximum, minimum, and standard deviation values of the ACC have been highlighted in bold. The AlexNet network achieves the best outcome for the average ACC



(a) Performance with Otsu segmentation and with and without data augmentation for AlexNet.



(b) Performance without Otsu segmentation and with and without data augmentation for Google InceptionV3.

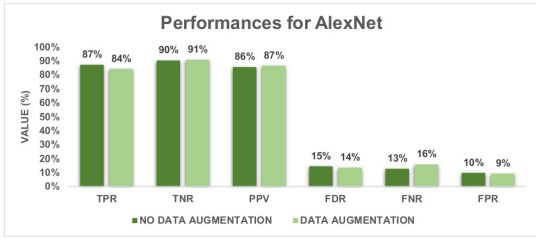


(c) Performance with Otsu segmentation and with and without data augmentation for GoogleNet.

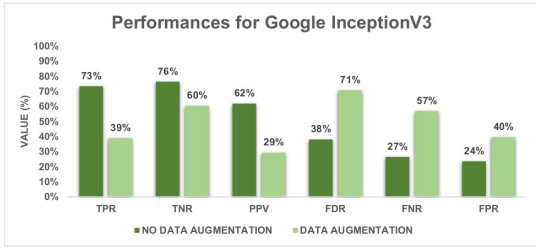
Figure 3.5: Performance for all used networks by applying Otsu segmentation.

without using data augmentation and with and without using segmentation (highlighted in red).

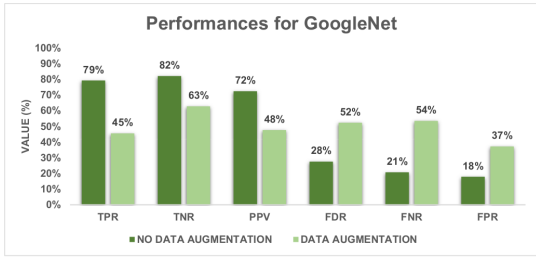
For each dataset, we assessed the performance of the networks by examining their behaviors. The results, as shown in Table 3.3, indicate that GoogleNet is the most robust network, with a mean prediction of accuracy which declines of -19.60 percent. Also, the data appear to back with what happened in the ISIC 2019



(a) Performance without Otsu segmentation and with and without data augmentation for AlexNet.



(b) Performance without Otsu segmentation and with and without data augmentation for Google InceptionV3.

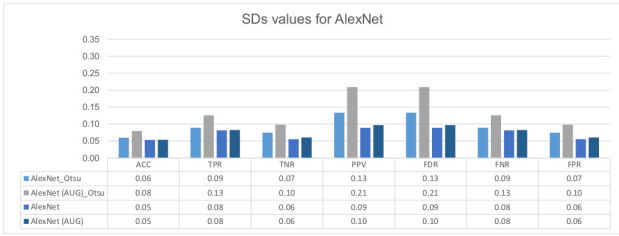


(c) Performance without Otsu segmentation and with and without data augmentation for GoogleNet.

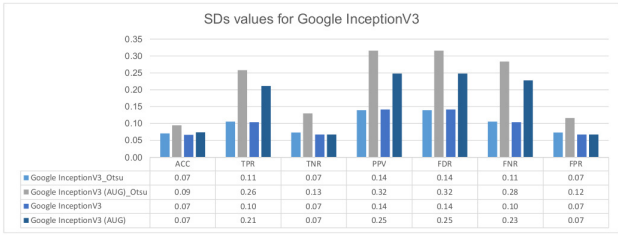
Figure 3.6: Performance for all used networks without Otsu segmentation.

challenge, where the ISIC 2018 winners saw their performance plummet by up to 28%.

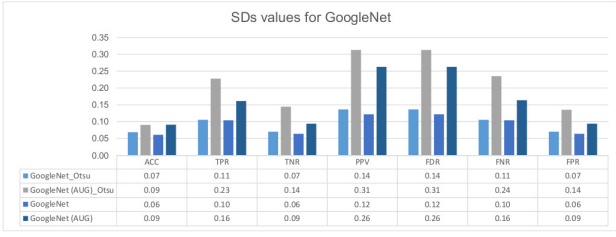
In Table 3.4, we explore the clock time for GoogleNet, Google Inception V3 and AlexNet, in the single, GRIMD (t2), GRIMD (m5) and GRIMD (c6) environments. In this case, under the two



(a) SDs values for AlexNet.



(b) SDs values for Google InceptionV3.



(c) SDs values calculated for GoogleNet.

Figure 3.7: Several SDs values computed for all networks.

Net	Measure	MD1	MD2	MD3	MD4	Mean Drop
AlexNet	Best	0.97	0.91	0.97	0.89	
	Average	0.81	0.72	0.81	0.73	
	Drop	-19.75	-26.38	-19.75	-21.91	-21.95
Google InceptionV3	Best	0.91	0.88	0.90	0.89	
	Average	0.75	0.72	0.75	0.74	
	Drop	-21.33	-22.22	-20.0	-20.27	-20.96
GoogleNet	Best	0.94	0.93	0.91	0.89	
	Average	0.81	0.77	0.75	0.74	
	Drop	-16.04	-20.77	-21.33	-20.27	**-19.60**

Table 3.3: Performance drop after 100 training steps (related to Training and Validation steps).

conditions of the tests, we intended to assess the amount of time saved by data scientists.

Environment	GoogleNet	Google InceptionV3	AlexNet
Single	82710	115200	19724
GRIMD(t2)	55140	94348	13327
GRIMD(m5)	20677	37105	6872
GRIMD(c6)	7519	17710	3171

Table 3.4: Clock time (in seconds) measured for both the experiments

We gathered the time and effort required to keep a classifier performing at its best. We spent up to 82000 seconds per retraining to achieve good results for the MED-NODE datasets, for just 170 images. The last results suggest that a tree layers hybrid architecture based on Cloud, Fog and Edge Computing must deal with the amount of data to be analyzed by reducing the running time of the continuous retrain. This step should improve the decoupling between data scientists and model training. Also, using user-generated images can speed up new model deployment.

3.7 EXPLORATION OF GENETIC ALGORITHMS

Skin cancer is one of the most dangerous and deadly cancers. Unfortunately, the incidence of skin cancer has been rising in recent years and, for some subtypes, the biggest problem is a lack of early detection, a limiting issue for first-line therapy in cases of this malignant pathology [128].

Following the results reported in [71] regarding the performance degradation in the case of small dataset changes, we started preliminary experimentation that aimed to understand if a hybrid approach merging Genetic Algorithms (GA) and standard Convolutional Neural Network (CNN) training routines could result in more robust classifier, even with a simplest NN structure. In particular, despite the CNN architectures available in the literature, in [41], we observed that small changes in the dataset could impact performance, with a mean drop of around 20%. Therefore, we assumed as a working hypothesis that for melanoma classification, current CNN architecture could be im-

proved. With what we call an Evolutionary-based CNN design approach (GA-CNN), we specifically avoided defining the NN structure a priori. In particular, we do not want to use GA to improve hyperparameter determination on a defined (and static) NN. Instead, we want to create a self-assembling NN population motivated by how effectively it solves a certain problem. Our working hypothesis is that a NN population using the GA approach can converge to a satisfactory solution (high prediction and confusion matrices accuracy), driving the NN development of layers by a scoring function: we used the accuracy parameter as the scoring algorithm of the GA evolution process. Furthermore, we used a clinical dataset (i.e., MED-NODE) for training and validation. Also, we compared the performance of the GA-CNN to that of AlexNet, both with and without Otsu segmentation. The initial findings obtained with this hybrid approach by merging the main capabilities of GA and CNN to handle the melanoma detection problem are reported in this contribution.

3.7.1 *Related Works*

In the latest years, new competitions, such as ISIC[2] and new melanoma detection tools, implemented as a statistical tool, machine/deep learning software or expert system and techniques were proposed. The most common tool's output is a binary answer: melanoma/ non-melanoma, but often, a percentage of confidence is provided.

Based on our working hypothesis, we do not intend to employ GA to improve the determination of hyperparameters on a defined (and static) NN. Instead, we want to create a self-assembling NN population motivated by how effectively it solves a certain problem.

Genetic algorithms are based on the principle of biological evolution and are used to optimize a variety of processes. Starting with a random population of network designs, the method iterates through three stages: selection, crossover and mutation [102]. Although attempts to combine neural networks and evolutionary algorithms can be found dating back to 1990 [192], the compu-

2 https://challenge.isic-archive.com/

tational efforts required to combine these techniques have only recently become more tractable thanks to a major cloud provider that allows users to borrow high computational architecture without having to build it from scratch. Over the last two decades, many studies in a variety of domains have used a mix of neural networks and evolutionary algorithms to address optimization and classification problems, ranging from river water quality prediction [42] to the most recent tuning of many hyperparameters at the same time [105].

3.7.2 *Methodology*

Genetic algorithms have been developed based on Darwin's evolutionary theories, presented in his book on the *Origin of Species by Means of Natural Selection* and the *Preservation of Favoured Races in the Struggle for Life* of 1859, and they were treated for the first time from John Holland in 1975.

Following the Darwinian principle that the most suitable elements of the environment have greater chance to survive and transmit their characteristics to the successors, these algorithms mimic these modes of evolution. Therefore, there is a population of individuals (n chromosomes), initialized randomly, which evolve from generation to generation through mechanisms similar to the natural evolutionary process. The binary string format is commonly used for chromosomes. Each locus (particular location on chromosome) has two alleles (variant versions of genes) in chromosomes: 0 and 1. Chromosomes can be seen as points in the solution space [93].

Evolution takes into account three fundamental processes:

- *selection*: the selection phase plays an important role in driving the search towards better individuals and maintaining high genotypic diversity in the population. The selection represents the choice of the most promising solutions, discarding the worst ones that do not generate individuals suitable for the environment;

- *cross over*: in order to explore other points in the search space, variation is introduced into the intermediate popu-

lation using some genetic recombination operator, such as cross over;

- *mutation*: some individual may undergo random variations or mutations. A mutation invariably causes a shift in the space of solutions, resulting in the generation of new information and in the recovery of knowledge lost in the population over time.

These evolutionary algorithms carry out heuristic exploration for new solutions to issues in which there is no complete knowledge of the search area and they explore all of it. Then, starting with the first solution, they tweak it, combine it and evolve it until they find a better result. So, GA dynamically changes the search process until it reaches an ideal solution through the probabilities of crossover and mutation. The full process is driven by the *fitness* function that assesses the survival of the chromosomes in the population.

3.7.2.1 *Experimental setting*

For the following experiment, the MED-NODE dataset was employed, which contains 170 clinical images, including 70 images of melanoma and 100 images of benign nevi [68]. We used the `Matlab 2021` environment and we defined our working objects following the GA terminology. The notation $F(t)$ indicates an object's composition at the time t. In particular, we define an entity E_i as a vector $E_i = \{F_1, \ldots, F_m\}$ of m features. We called each feature F_j of a generic entity E_i a gene of E_i. The entire set of genes is called the Genome of E_i.

In order to allow the experiment to reach a sufficiently extended network architecture, the start size of the genome was set to ten to allow at least the presence of the minimal layers needed to execute a CNN (input, convolution, RELU, softmax and Fully Connected). In addition, we allowed the genome size to grow using the merge operation (not related to the GA fundamental) to make network architecture more complex: merge operation sticks two different genomes, doubling the size of an entity genome. In our simulation, each gene can represent a Matlab CNN core object (network layer) or a pre-processing routine, specifically

Otsu segmentation [146]. Consequently, each chromosome is represented by an array in which each cell represents the presence or not of a feature within the entity. Feature indicates one of the possible layers of which a CNN can be composed. If the feature (array cell) is active, we consider the feature as expressed and therefore the layer belonging to the network. If it is not active, we consider the feature not expressed and the layer does not belong to the network, even if it could belong in the future in some evolutionary cycle. So, each feature F_j can be expressed or not by E_i. This mean that a new entity E_k could inherit a gene F_e from E_i starting to express it. Then, in our simulation, we have a silent and expressed gene. In addition, because the merging procedure has been implemented and it admits junction of two chromosomes, each chromosome does not have a fixed length. In this way, we did not want to limit the final size of the network to a fixed number. We have related the *selection* to the score function and to the capacity of the entity to survive in the execution environment. Therefore, if an entity (which represents a date structure of a CNN) allows the train function, without crashing, is temporarily selected as potential survivor. For *cross over* and *mutation*, the chromosomes to be recombined or to change (deactivate/activate or change their layer) are chosen randomly.

The set $P(t) = \{E_1, \ldots, E_n\}$ is called Population at time t. The population size $n(t)$ might vary related to t. We defined the following constraints: the first gene of each entity must be an image input (II) or one of the pre-processing routines we defined before; if the gene g is a pre-processing routine, then the gene $g + 1$ must be a II layer or another pre-processing layer; the latest gene of an entity must be a classification layer. The population in our experiment is made up of all living entities. We limited the gene types that an entity might use in the experiment to: *Convolution*, *RELU*, *Cross Channel Normalization*, *Max Pooling Grouped Convolution*, *Fully Connected Layer*, *Dropout* and *Softmax*. All entities with expressed genes that are incompatible with the environment die at the end of each evolutionary phase. Then, if an entity exposes a genes pipeline that the training function of Matlab does not allow, it will die immediately. The following configuration was used to train each compatible entity:

```
('sgdm', ...
```

```
'MaxEpochs', 16, ...
'MiniBatchSize', 12, ...
'Shuffle', 'every-epoch', ...
'InitialLearnRate', 0.0001, ...
'ExecutionEnvironment', 'auto')
```

We employed the maximizing of global population accuracy as the function to drive population evolution. Therefore, accuracy (calculated with the formula presented in Subsection 2.4.3) indicates our *fitness* function, which is currently a simple decreasing order of the accuracy of all the networks of an evolutionary cycle. We calculated the highest accuracy from each survived entity for each evolutionary phase. As a result, all entities that expose an accuracy at time t equal to or better than the maximum accuracy of the previous generation's $t-1$ will survive for each generation. In addition, a random 10% of entities were picked at random in each step to live, regardless of the accuracy at time t. The GA was terminated if no improvement in accuracy was noticed for ten consecutive evolution stages. We were obliged to limit the conceivable crossover and mutation due to the physical limitations on the cloud platform. As a result, each surviving creature was restricted to only 10 mutations and 100 crossovers. An initial randomized population of 10K entities was used to try to alleviate these restrictions. After 100 iterations, we caused the stop of evolution process.

3.7.3 *Experimental Results*

We ran the AlexNet network with and without Otsu segmentation on MED-NODE 100 times, each time repeating the training step. Then we ran the GACNN for 100 iterations, enabling the system to evolve. As a reference parameter, we used *Accuracy* (ACC). For the standard AlexNet execution, we computed average ACC (*mean ACC*), maximum ACC (*max ACC*), minimum ACC (*min ACC*) and Standard Deviation (*SD*), as reported in Table 3.5. The best AlexNet performance was 0.97%, while the mean ACC was 0.81%. For GACCN, we evaluated the ACC trend over the evolution steps, as reported in Figure 3.8. The max ACC reached by GACNN is 0.97 before reaching the 100^{th} iteration.

MED-NODE					
Net	**Segmentation**	**min ACC**	**max ACC**	**mean ACC**	**SD**
AlexNet	-	0.68	0.97	0.81	0.06
	Otsu	0.50	0.91	0.72	0.07
GACNN	-	0.68	0.97	-	-

Table 3.5: Performance of AlexNet on the MED-NODE dataset.

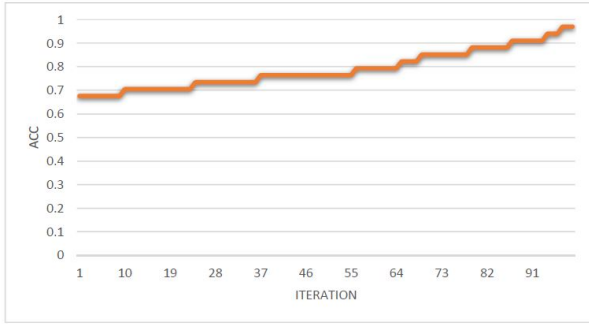


Figure 3.8: Performance of GACNN over 100 iterations.

3.8 DISCUSSION

Despite the great performance showed in the literature, the data provided in these two study approaches imply some final considerations. The Transfer Learning technique, which is widely used currently, may not be reliable. In fact, based on the first contribute to the melanoma detection in [41], the results reveal that modest changes in the training dataset cause a fluctuating performance of the classifiers, which drops substantially, and that constant retraining is essential to limit this loss. These findings suggest that a hybrid architecture based on Cloud, Edge, and Fog layers can effectively contribute to computationally onerous operations in the field of melanoma detection and classification. Moreover, when no segmentation was applied, the CNN networks performed better. According to our results, as future work we plan to design more robust neural network models to better learn from the images (and generalize from them).

According to preliminary findings in the second contribute, allowing GA to drive the design of a NN structure could result in performance comparable to traditional NN training approaches.

Also, our preliminary results suggest that the NN architecture found by GACNN is more stable than standard CNN, in this particular case, AlexNet. We observed that the GACNN outperforms AlexNet mean accuracy (computed over 100 executions), reducing the mean drop in performance caused by small dataset changes. A plateau set is also shown in Figure 3.8 that demonstrates a decreasing trend per nine iterations in mean. These findings could indicate that the population is approaching the convergence to the optimal solution. In contrast, we discovered a significant *death ratio* (up to 95 percent for each evolution step). This observation may indicate that finer definitions of beginning population or recombination stages are required. In addition, the execution times are quite long. About 8 minutes are needed to train each network. For this reason, the entire project has taken advantage of the ability to run experiments on the cloud. At the current state of the experimentation, we discovered that execution times scale linearly with the complexity of the network (chromosome length). From this work, we obtained the main advantage which derives from the observation that the drop consequent to changes in the composition of the training dataset is lower than the drop reported in our previous study in [41]. A more in-depth investigation of population evolution and behavior, particularly the *death ratio*, is required. In fact, because the work is still experimental and in progress, we are attempting to test many options (ranging from fitness to the optimum population definition) that meet the computational and temporal performance requirements for achieving the solution.

4

RECONSTRUCTION OF 3D PROTEINS STRUCTURE

This Chapter explains the use of a *Bidirectional Long Short-Term Memory* (BLSTM) neural network and discrete classes for prediction of torsional angles of a protein. First, we have an overview about the function of these angles (see Section 4.1) and some methods common used in literature in the last years (see Section 4.2). Then, we explored the dataset we used, data preparation and features used as input for BLSTM (Section 4.3). Following, we present the basic architecture of the neural network LSTM and the approach investigated in this work (Section 4.4). Then, we describe the adopted performance measures (Section 4.5) and the preliminary considerations from which our work originates (see Section 4.6). In Section 4.7 we present the final results of this work and we apply two methods for their visualization (Section 4.8). In Section 4.9, we discuss the obtained results and the concluding remarks.

4.1 INTRODUCTION

The exploration of produced protein sequences has become increasingly relevant as a result of current genome sequencing studies and the ever-increasing deposition of protein structures in the *Protein Data Bank* (PDB) [169]. In the subject of proteomics, research has focused on the creation of algorithms and bioinformatic tools that take a protein sequence as input and extract information about its structure and function, as well as its features, fold and interactions with ligands and other proteins. The protein fundamental structure, or the sequence of amino acids from which it is made, plays a critical role in this regard, as it provides the information required for protein folding in its three-dimensional structure. In structures with well-defined angles, the secondary structure tells how the protein backbone is folded locally [39]. The ϕ angle between N and Cα, ψ angle between

Cα and carbonyl carbon and ω angle between carbonyl carbon and N, commonly fixed at 180°, all characterize the structural protein backbone. These angles do not have arbitrary values; instead, they fall into specified Ramachandran plot areas for proper protein folding. The Ramachandran plot is a graph that highlights the primary chain conformation angles and assigns them to the related secondary structures [98]. Helixes, strands, and coils are the three basic types of protein secondary structures. New methodologies and tools for predicting dihedral angles have emerged over time. To improve protein torsional angle identification, techniques based on *Support Vector Machine* (SVM) and neural networks were developed.

4.2 RELATED WORKS

There are two sorts of methodologies in the scientific literature for predicting ϕ and ψ torsional angles as discrete and continuous: methods based on sin and cos prediction and methods based on discrete class prediction.

Heffernan et al. [78] presented a Bidirectional Recurrent Neural Network (BRNN) to capture non-local interactions that spread along a protein sequence between the residues with the major long-range connections. The first set of predictors are seven representative amino-acids physio-chemical properties (PP) [51], 20-dimensional Position *Specific Substitution Matrices* (PSSM) from PSI-BLAST [8] and 30-dimensional *hidden Markov Model* sequence profiles from HHBlits [149] per residue. They obtained predictions for secondary structure, *Accessible Surface Area* (ASA) [32], backbone angles, *Half Sphere Exposure* (HSE) and *Contact Number* (CN) [75]. The outputs of this first iteration are added to the PSSM, PP and HMM profile features as input for a second iteration. The model is used to predict Accessible Surface Area (ASA), Half Sphere Exposure (HSE), Contact Number (CN) and angles ψ, ϕ, θ and τ, with a total of 14 outputs: the first for ASA, the following eight for $\sin\phi$, $\cos\phi$, $\sin\psi$, $\cos\psi$, $\sin\theta$, $\cos\theta$, $\sin\tau$ and $\cos\tau$; the four successes for HSEα-up, HSEα-down, HSEβ-up, and HSEβ-down; the last for CN [78].

The second model received as input the output of SPIDER2 and the PSSM. The SPIDER2 [77] output includes expected secondary

structures, probability for the three types of secondary structures, *relative solvent accessibility* (RSA), sin and cos functions of the backbone angles ψ and ϕ, angle θ based on the Cα atom and rotation angle τ, contact numbers based on the Cα and Cβ atoms and up-and-down half-sphere exposures (HSE) based on the Cα vector -Cβ and on the vector Cα-Cα. The backbone angles are divided into 5° bin.

Raptor X-Angle is a new method which combines deep learning and clustering techniques to predict the real values of protein backbone dihedral angles. In RaptorX-Angle it is assumed that the angle pairs follow a bivariate von Mises distribution in order to take into account the circular nature of the angles [61].

4.3 METHODS

4.3.1 *Dataset e Features Description*

The entire dataset consists of 173 protein sequences for a total of 34,721 residues downloaded from the PDB [169] and belonging to *Homo Sapiens* organism. Based on this, we proposed a system for the prediction of torsional angles called *Human Proteins Angles Prediction* (HPAP).

The selection criteria for proteins in the PDB (two examples of proteins are reported in Figure 4.1) were as follows:

- X-Rays as a method used to determine the structure of the protein;
- a resolution between 1.5 and 2.0 Å to return details at the atomic level;
- R-free values, a measure of the quality of a structure, was chosen between 0.25 and 0.30 [98];
- content of the secondary Structure from minimum of 40% to maximum of 100% for both α-helix and β-sheet.

To reduce duplicated sequences, only proteins with a 30 percent pair sequence identity were chosen and only chain A was selected. For each protein, the PDB file and sequence in FASTA

format[1] have been downloaded, containing the atomic coordinates of the residues and the calculated torsion angles values for each amino acid. To estimate the distribution of errors between the expected torsion angles and their real values, 100 proteins were randomly picked from the source dataset. The proteins have an average length of 201 residues.

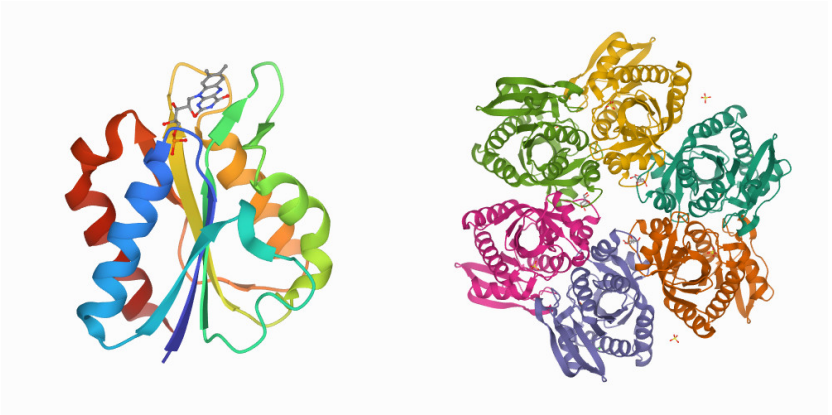


Figure 4.1: Proteins 1C7E and 1ODL in PDB and HPAP dataset.

We have used a total of 34 input features per residues:

- the protein sequence (one letter for each of the 21 amino acids);
- the three types of secondary structures (H for α-helix, E for β sheet and C for coil) calculated using the *hydrogen bond estimation* (DSSP) algorithm [91], extracted from the files PDB;
- chemical-physical properties (steric parameter, polarizability, van der Waals normalized volume, isoelectric point, α-helix and β sheet probabilities) obtained from [121] which represent universal descriptors for amino acid side chains;
- degree of hydropathy and aliphatic index [86];
- PSSM (*Position-specific Scoring Matrix*), obtained by multiple alignment with PSI-BLAST with 3 iterations, performing

1 https://zhanglab.ccmb.med.umich.edu/FASTA

the search against the NR database and with an E-value of 0.001 [90];

- *accessible surface area* (ASA) [32] both at residual level (indicated with `RES.ASA`) and fractional ASA (`FRAC.ASA`) and *occupied volume* (VOL) both at residual level (`RES.VOL`) and fractional volume (`FRAC.VOL`), calculated with VADAR [193].

ASA represents the protein area exposed to the solvent, while `FRAC.ASA` is determined by dividing the observed ASA by a given residue from the ASA calculated for that residue in the *Gly-X-Gly* tripeptide. VOL represents the volume occupied by a residue defined by its atomic radius and its neighbors and is measured in cubic Å. The volume defines the packaging density of the protein in correlation with the spatial arrangement of the atomic groups. The calculation of this parameter is useful for finding hollow areas within the protein, atomic overlaps or problematic protein regions [151].

In this work, we have introduced four new features:

1. *RES.ASA*;
2. *FRAC.ASA*;
3. *RES.VOL*;
4. *FRAC.VOL*.

These features represent the ASA and VOL values, each referred to for the single amino acid residue and in fractional value.

4.3.2 *Data Preparation*

The *one-hot* scheme was used to code the features that represent the sequence and secondary structure. The scheme provides that a vector of zeros has only a value of 1 at the index of the class [161].

The PSSM matrix was normalized according to the function in Eq. 4.1:

$$f(x) = \frac{1}{1 + e^{-x}} \tag{4.1}$$

The values of the other features have been standardized to a range of 0 to 1. The residue samples from each unique protein sequence were normalized. Similar findings were obtained using a different normalization on the complete population of residues. To standardize the lengths of the protein sequences, fictional values were created to act as a divider between the various remaining sequences. We explored a maximum length protein of 456 residues, and we settled on a fixed length of 480 residues because this figure is easily divisible by batch size. $480 - \textit{sequence length}$ is the number of fictional x values added at the beginning of each sequence. In order to recognize this fictive data, a dummy class was added to the network models. Batch sizes of 10, 32, 64, 128, 200, 256 were investigated for single output models trained on sequences without padding. Batch sizes close to the average sequence length produced the greatest results. The batch sizes that generated the greatest results when using the same models trained on padded sequences were those that split the new protein size. The same assumptions applied to the models with numerous outputs. The dataset was split using the following best ratios: 79% training set and 21% validation set, after trying multiple random breakdowns of the sequences in order to preserve protein sequences intact.

Below, the input size for the *stateless* model is reported:

```
(input_layer = Input(shape=data_x.shape[1:])
```

The input size for *stateful* model follows:

```
model.add(Bidirectional(
    LSTM(256, return_sequences=True, stateful=True),
batch_input_shape=(batch_size, data_x.shape[1], data_x.shape
[2]))
```

We can see how the size of the rows and the size of the columns are taken as input for both models.

4.4 LSTM APPROACH

The *long short-term memory* (LSTM) is a specific *recurrent neural network* (RNN) architecture consisting of memory cells organized in memory blocks, recurrently connected. Each of these contain

three multiplicative units: an input gate, an output gate and forget gate. The input and output gates supervise the input and output activation in order to control information flow into the blocks. Through the gates, the net can decide whether to access or override the memory cell's information content. Every memory cell has a core consisting of a recurrently self-connected linear unit-Constant Error Carousel (CEC) which allows us to maintain the network error constant. These networks were created to overcome the problem of gradient decay over long sequences [80] and in order to find the optimal time lag for time series problems [81]. Compared to RNNs, LSTM networks allow evaluating the evolution of inputs by capturing dependencies in long-range distances [156]. In this work, a BLSTM system is proposed, inspired by the neural network presented by [78].

The basic neural network architecture includes two BLSTM layers, with 256 nodes, followed by two fully connected layers (FC), with 1024 and 512 nodes respectively. In Figure 4.2 is represented the basic LSTM model.

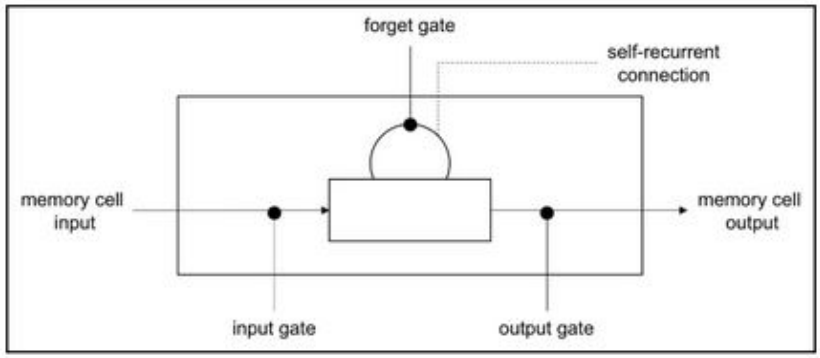


Figure 4.2: LSTM model.

For the construction and training of neural networks, the Deep Learning Keras library[2], a high-level API of TensorFlow [1], under the Python language[3], was used. To reduce training times, the Google Colab platform[4] is used, which provides an NVIDIA Tesla

2 Keras: https://keras.io
3 Python: https://www.python.org
4 Google Colab: https://colab.research.google.com

K80 GPU[5] as a hardware accelerator for processing. Pandas[6] and Scikit-learn libraries[7] are used for dataset management and data normalization.

We have built three variants of the proposed neural network:

- *M1*: it outputs the class prediction of a single angle;
- *M2*: it provides the classes of ϕ and ψ angles in a parallel way;
- *M3*: it foresees a pair of BLSTM layers for each input feature.

In Fig. 4.3 the structure of the most complex variant M3.

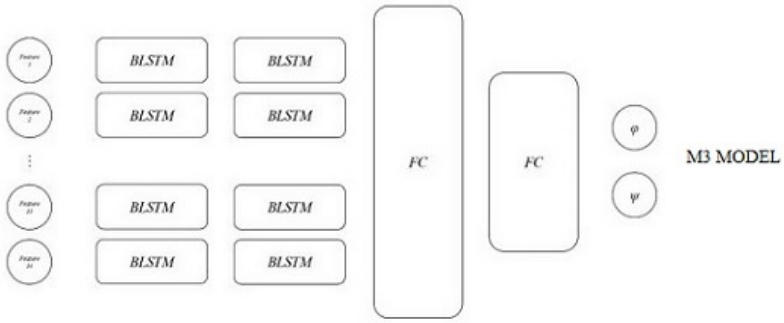


Figure 4.3: M3 variant.

In each variant, fully connected nodes use the *Rectified Linear Unit* (RELU) activation function, which allows working with large numbers as it maps the inputs x into the interval (0,x).

The output nodes use the *Softmax* function, which forces the network to output a range of values whose sum is 1. Therefore, the output values of the *Softmax* function can be considered as part of a probability distribution. The outputs represent the classes to which the angles ϕ and ψ belong. Each class provides the value of the angle amplitude with a maximum error of $\pm$ 2.5°.

For the prediction, the *Categorical Crossentropy Loss* function is used, which compares the distances between the outputs of the

5 GPU Tesla K80: https://www.nvidia.it/object/tesla-k80-dual-gpu
6 Pandas: https://pandas.pydata.org
7 Scikit-learn: https://scikit-learn.org

Softmax function and the values encoded with the *one-hot* scheme, following the Eq. 4.2:

$$Loss = -\sum_{i=1}^{outputsize} y_i \hat{y}_i \tag{4.2}$$

where $\hat{y}_i$ is the i-th value in the model output, y_i is the corresponding target value and *output size* is the number of values in the model output.

The model is trained with the *Adam optimizer* [97]. We searched for the optimal *batch size* able to best represent the average length of the protein sequences. The monitoring of the loss function on the validation data avoided training set overfitting. The M1 and M2 variants were trained with both *stateless* and *stateful* LSTM cells [101].

The ψ and ϕ angles prediction is treated as a classification problem, according to the study of [60]. The angle amplitudes are first divided into classes of 5° intervals. We've created a total of 73 classes, with 72 of them coding angles between -180° and 180° and a third class coding free angles. These angles can have any value in the range [-180°, 180°] at the start of a series. At the second scenario, we divided the angles into 37 classes by grouping them in 10-degree intervals.

In addition to training the networks with LSTM *stateful* and LSTM *stateless* cells, two types of normalization were tested: in the first we normalized the values of all features between [0, 1] and we have indicated this stage with `NORM1`; while in the second case we have normalized the PSSM as in Eq. 4.1, referred as `NORM2`. Later, we have introduced padding techniques (called here `PAD480`).

4.5 PERFORMANCE MEASURES

In order to compare the results, we have chosen the *mean absolute error* (MAE), whose formula is indicated in Eq. 4.3:

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} \tag{4.3}$$

where y_i is the expected value, x_i is the real value and n is the total number of observations.

We have also calculated the accuracy (ACC) as measurement criteria, as reported in Equation 2.1 in Subsection 2.4.3.

The ACC indicates the angle classes percentage correctly predicted by the model. Correct prediction implies an error of $\pm$ 2.5° on the continuous angle value. The MAE, instead, indicates the error in absolute terms on the continuous value of the angle amplitude between predicted angles and calculated angles experimentally.

4.6 PRELIMINARY CONSIDERATIONS

In Table 4.1, we have reported the various combinations of tested variables. *M1*, *M2* and *M3* represent the proposed neural networks variants. *S1-S10* indicate the performed experiments, with the relative two types of normalization used, the type of training (*stateful* or *stateless*), the number of introduced angular classes (37 o 73) and the padding addition.

When we compared the outcomes of the first and second studies (S1 and S2) using the same M1 model, we found that S2 performed better. As a result, we concentrated on S2 experiments rather than S1. Starting with the S3 experiment, the addition of padding has resulted in an average improvement in results. In parallel, grouping of angles into 37 classes (S4), we have noticed a clear improvement in the ACC. It was hypothesized that this was

Table 4.1: Details on conducted experiments

Experiment	Model	Angle class	Normal.	Training	Padding
S1	M1	73 classes	NORM1	Stateless	-
S2	M1	73 classes	NORM2	Stateless	-
S3	M1	73 classes	NORM2	Stateless	PAD480
S4	M1	37 classes	NORM2	Stateless	PAD480
S5	M1	73 DISTR	NORM2	Stateless	PAD480
S6	M2	37 classes	NORM2	Stateless	PAD480
S7	M2	73 classes	NORM2	Stateless	PAD480
S8	M2	37 classes	NORM2	Stateful	PAD480
S9	M2	73 classes	NORM2	Stateful	PAD480
S10	M3	73 classes	NORM2	Stateless	PAD480

Table 4.2: ACC, MAE e MAE variation for ϕ

Exp.	BS	With new features		Without new features		Var(°)
		ACC	MAE(°)	ACC(%)	MAE(°)	
S2	200	17.13	21.3	18.26	20.78	0.52
S3	240	17.76	21.27	17.8	21.37	-0.1
S3	480	17.63	21	17.9	21.61	-0.61
S7	120	17.85	21.33	17.96	21.26	0.07
S7	240	17.76	21.34	17.73	21.05	0.29
S7	480	18.21	20.68	18.11	20.82	-0.14
Average Variation						0.005

due to the reduction of the classes to be foreseen and the merging of several angles in a larger class. In the S5 trial, we have tested clusters which allowed for classes with a more evenly distributed average population (DISTR). Experiments S6 and S7 respectively resume experiments S3 and S4 using the M2 variant of the model. The multiple output of this model allows us to reduce training and forecasting times. In the S8 and S9 trials, a different approach was tried: adapting the model to the data. Iterative training was tested on stateful versions of both models. For this training, the data no longer requires a preliminary processing step to add separators between the various sequences, as data groups with heterogeneous lengths are allowed. However, the results obtained were equivalent, except for a performance degradation due to more time for training and forecasting. In the description of the subsequent results, we will exclude these experiments. In the experiment S10, the achieved ACC by the *M3* variant was found to be, in the testing phase, lower than that of the *M1* and *M2* models. In addition, training and forecasting times have grown exponentially. These reasons have led to discard the development of this variant. Following these considerations already encountered in the experimentation phase, we therefore relied on the following experiments: S2, S3 and S7.

Table 4.3: ACC, MAE e MAE variation for ψ

Exp.	BS	With new features		Without new features		Var(°)
		ACC	MAE(°)	ACC(%)	MAE(°)	
S2	200	15.66	35.18	15.78	36.95	-1.77
S3	240	16.01	34.75	16.06	36.13	-1.38
S3	480	15.58	34.68	15.7	36.69	-2.01
S7	120	16.03	35.04	15.48	36.61	-1.57
S7	240	16.13	34.87	15.46	37.45	-2.58
S7	480	15.58	34.91	15.53	37.24	-2.33
Average Variation						-1.94

Table 4.4: ACC e MAE for 37 and 73 angles classes

Class	Angle ϕ		Angle ψ	
	ACC(%)	MAE(°)	ACC(%)	MAE(°)
73	18.28	20.65	16.76	34.40
37	32.01	20.44	29.20	34.40

4.7 RESULTS

Tables 4.2 and 4.3 show the results of the experiments carried out. We have considered *batch size* (BS), type of experimentation, the calculation of the ACC and of the MAE with and without adding the new features and the variation with respect to the MAE for both angles ϕ and ψ.

Considering only the ACC, the contribution of the new features is negligible. However, some differences are found when examining the MAE. The addition of the new features involves an average reduction of the MAE relative to ψ angles of -1.94° (see Table 4.3). The same is not true for the angle ϕ. In this case, the addition of the new features is irrelevant, with an average MAE's variation of 0.005° (see Table 4.2).

Reducing the classes of angles to be predicted, from 73 to 37, there is a net increase in the ACC, almost double and a small improvement in the MAE, as shown in Table 4.4. However, with

37 classes, the expected output admits an error of $\pm$ 5° on the continuous angle values. For this reason, for the purpose of comparison with other studies, the results obtained with a finer-grained grouping in 73 classes were taken into account. There has not been a single model that has simultaneously achieved the maximum ACC and the minimum MAE, but the *M2* variant ensures better results on average and reduction of forecast time.

For the comparison with the results present in the literature, we have considered the MAE and the ACC, reported in Table 4.5 and with our best results declared in this work. The proposed M2 model provides an ACC of 18.3% for ϕ and 16.8% for ψ, not far from the results in the literature. For the MAE, we stand at 20.65° and 34.40° for ϕ and ψ, respectively. The introduction of the new features *RES.ASA*, *RES.VOL*, *FRAC.ASA*, *FRAC.VOL* improves the angle ψ prediction, reducing the MAE of about 2°.

4.8 VISUALIZATION OF RESULTS

Our predictor produces a pair of angles, each of which corresponds to a prediction for a single amino acid in the sequence. In order to make the reading of the data more immediate, we opted to incorporate two graphic representations to show the accurate and incorrect predictions of angles ψ and ϕ. For this purpose, we aligned two identical copies of each protein, based on their intrinsic nature to be represented by a string. In the first copy, residue character with the experimental ϕ and ψ pair values are represented. These experimental values were extracted from PDB files of the PDB site [169] for each individual protein in the dataset. Each second copy residue character reported predicted

Table 4.5: Comparison with other works

Study	Dataset size	ACC ϕ	ACC ψ	MAE ϕ	MAE ψ
Heffernan *et al.* [78]	5789	-	-	18.3°	27°
Gao *et al.* [60]	5789	19.6%	17.4%	-	-
Li *et al.* [112]	1652	-	-	20.49°	29.06°
HPAP (our method)	173	18.3%	16.8%	20.65°	34.40°

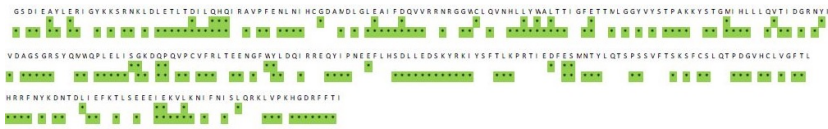


Figure 4.4: Pair-wise alignment.

values for the two angles. The maximum error for both torsional angles is $\pm$ 2.5° degrees, according to our preliminary experiment results. This threshold was used to distinguish between correct (in green) and incorrect predictions. We distinguish between correct prediction, incorrect prediction and disaster prediction. The disaster prediction is connected to the condition where the predicted angles have an inverse sign in terms of experimental value, and this condition disrupts the three-dimensional structure of the protein.

Figure 4.4 shows a partial pair-wise alignment. The protein sequence and *phi* and *psi* angles predictions are represented in the rows. The classes that the system accurately predicts are shown in green, with a maximum expected error of $\pm$ 2.5°.

We also created a Yanaconda macro [8] to plot the predictor output in 3D space so that the discrepancies between real and predicted angles can be seen in a more understandable and immediate visualization. The Yanaconda macro was executed by YASARA[9], a molecular modeling program [104]. Yanaconda allows us to extend the capabilities and behaviors of YASARA. We used YASARA in interactive mode to provide the user complete control over each step of the visualization process. The macro allows the user to load each structure into the 3D space (one at a time, from the test-set). It also gives the user the option of applying (or undoing) the projected angles for each structure residue (still one at a time). Finally, the macro can be paused to allow the user to view or manipulate the structure. Because the predictor's chemical limitations are not fully generalized, some predictions require user involvement with the viewer.

In Figure 4.5 we showed a series of three residues with predicted torsional angles, the third of which has a significant displacement in the opposite direction of the real value. As another

8 Yanaconda scripting language: http://www.yasara.org/yanaconda.htm
9 Yasara View: http://www.yasara.org/viewdl.htm

example, in Figure 4.6, the ϕ and ψ angles of a Proline residue are compared to the original residue on the left. On the right, the identical Proline when the predicted torsional angles is depicted. Because the molecular structure of the amino acid Proline involves a cycle, applying the expected angles causes the cycle design to fall apart. Due to our model's lack of generalization, this flaw can only be overcome by user intervention, which will reverse the application of predicted angles.

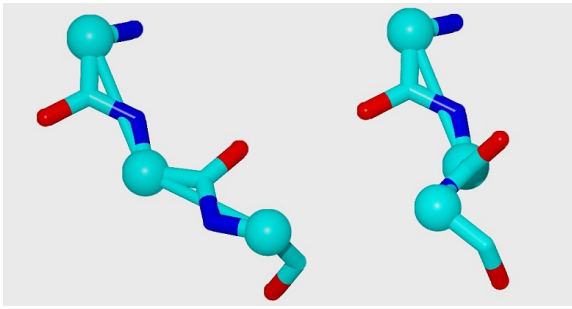


Figure 4.5: Comparison between predicted and original chains of three residues.

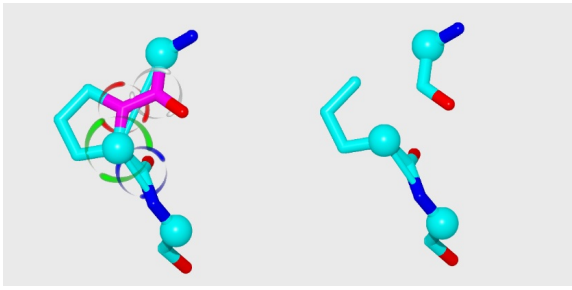


Figure 4.6: Comparison between original and predicted Proline residue.

4.9 CONCLUDING REMARKS

This chapter has presented an approach for prediction of torsional angles of proteins based on BLSTM neural network system, called *HPAP*, since trained only on human proteins. We have divided the angles into classes of 5° each, with a maximum expected error of $\pm$ 2.5°. In relation to the ASA and the Volume, four

additional features have been included. The dataset contains 173 human proteins (for a total of 6001 amino acid residues) and is 33 times smaller than those previously studied. Despite this, the obtained results, albeit lower, are not significantly different from those obtained by other literature investigations. In general, the usage of a BLSTM system for discrete class prediction has proven to be highly dynamic, as training on the supplied dataset only takes a few minutes. Furthermore, the addition of the new features allows us to lower the average absolute error on the ψ angle value prediction. In the future, expanding the dataset with new samples, including non-species-specific ones, could aid in the optimization of the model and improve accuracy for both torsional angles.

5

SNARER

In this Chapter, we present new molecular descriptors for a specific family of proteins, called SNAREs. We start with the introduction of related biological background (see Section 5.1) and this protein family (Section 5.2). Then, we discussed the studies already presented in literature in Section 5.3, the used classification algorithms (see section 5.4) and our contribution in terms of SNARE descriptors (Section 5.5). We described the methods for this work (Section 5.6), the experimental results (Section 5.7) and finally the discussion about our contribution and future works concludes the chapter (Section 5.8).

5.1 BACKGROUND

The process of determining the exact sequence of nucleotides that make up the whole genome of distinct living organisms is referred to as *genome sequencing*. Gene sequencing is becoming increasingly important, particularly as precision medicine develops. The latter highlights the prospect of increasingly personalized preventive, diagnosis and treatment protocols that are focused on the patients and are based on their genomic constitution [153].

Technology for genetic sequencing has advanced significantly in recent years. Parallel to this, the demand for new bioinformatic technologies to help in data acquisition, retention, analysis, and interpretation arose. These data to be analyzed range from the whole genes of an organism (genome) to the set of proteins produced (proteomics). Protein sequencing is one of the many biological disciplines where high-throughput sequencing techniques generate a lot of huge data [150]. For the identification of different genomic and protein regions, these massive amounts of data (up to petabytes) must be computationally evaluated using ever newer approaches. The current task is to contribute to this post-sequencing analysis and classification, as well as to assure

improved precision in the discriminating of accessible protein sequences.

In fact, the collection of protein sequences is constantly growing. There is a requirement for effective categorization methods that can characterize a protein's functionality based on its chemical-physical properties and label the sequence with greater precision. The more data we have on a protein, the better we will be able to fit it into a more sophisticated biological framework. This is clear and beneficial, especially when dealing with a protein whose function was initially unknown. The most common method involves determining whether the protein contains functional motifs and domains that allow it to be characterized from its amino acid sequence and determining whether it belongs to a protein family with members that have similar three-dimensional structures, functions and significant sequence similarities. To establish their role and mechanisms in a certain physiological and pathological biological path, knowledge of the protein family representatives is required.

5.2 SNARE PROTEINS

SNARE (*Soluble N–ethylmaleimide sensitive factor Attachment protein Receptor*) is a protein superfamily involved in the molecular trafficking between the different cellular compartments [176]. The evolutionary significance of the SNAREs superfamily is inextricably linked to their role in many cellular functions and pathological states, prompting researchers to further investigate their role in biological pathways [82, 123]. SNARE proteins consist of motifs of 60-70 amino acids containing hydrophobic heptad repeats, which form coiled-coil structures. The core of the SNARE complex is represented by 4 α helix bundle, as evidenced by the available crystallographic structures [170]. The center of the bundle contains 16 stacked layers which are all hydrophobic, except the central layer "0", which is called ionic layer and which contains 3 highly conserved glutamines (Q) and a conserved arginine (R) residue (see Figure 5.1).

Evolutionarily conserved members of this protein family can be found in yeasts through mammalian cells. Basic cellular functions such as the production of proteins and hormones, the

release of neurotransmitters, the immune system's phagocytosis of pathogens and the movement of molecules from one compartment of the cell to another rely on vesicle-mediated transport. Membrane receptors are involved in vesicular transport, which involves the identification of vesicles, the activation of membrane fusion and restructuring and the subsequent release of vesicular content into the extracellular environment (exocytosis) or inside the cell (endocytosis). SNARE complexes, in particular, facilitate membrane fusion during diffusion processes by forming a bridging connection between SNARE proteins on both membranes [29]. Initially, SNARE proteins were split into two categories: *vesicle* or v-SNARE proteins that are incorporated into vesicle membranes and *target* or t-SNARE proteins that are connected with target membranes. R-SNARE and Q-SNARE are two more recent subdivisions that are based on structural characteristics. R-SNARE proteins have an arginine residue (R) that aids in the formation of the complex, whereas Q-SNARE proteins have a glutamine residue (Q) and are categorized in order of their location in the bundle of four helices. They are classified in turn as Q_a, Q_b or Q_c [52]. Scientific investigations have indicated that SNARE proteins are involved in several brain diseases due to their critical involvement in neuronal and neurosensory release at the synaptic level [147]. The release of neurotransmitters is a highly regulated process that occurs thousands of times each minute in both time and space. The formation and disassembly of SNARE complexes is closely regulated in this situation. Impaired neurotransmitter release at any stage might cause dys-

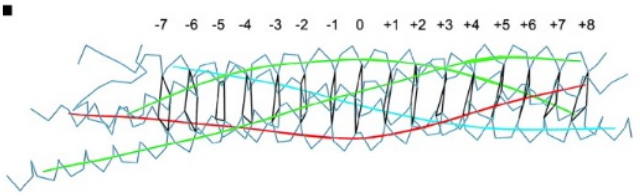


Figure 5.1: Visualization of the layers of the bundle of the fusion complex between the 4 parallel α-helices of the SNARE: 7 upstream layers (layers from -1 to -7) and 8 downstream layers (layers from +1 to +8) of the ionic layer (the layer 0) [52].

functions that jeopardize synaptic communication equilibrium. These compounds appear to play a role in the progression of neurodegenerative diseases (such as Alzheimer's and Parkinson's), neurodevelopment (autism) and psychiatric disorders, such as bipolar disorder and schizophrenia as well as depression. Different studies have shown the involvement of mutated or not properly regulated SNARE genes in the development of these disorders [50, 62, 72, 129, 165, 197].

5.3 RELATED WORKS

Since SNARE proteins are involved in numerous biological processes, studies have slightly increased in recent decades in order to identify and classify these proteins, but the papers dealing with this topic are still few. In the literature, there are documents that are based on different techniques, ranging from statistical models to the use of convolutional neural networks.

Kloepper *et al.* [99] have implemented a web-based interface which allows the new sequences submission to the Hidden Markov Models (HMM) for the four main groups of the SNARE family, in order to classify SNARE proteins based on sequence alignment and reconstruction of the phylogenetic tree. For their study, a set of ∼150 SNARE proteins is used in conjunction with the highly conserved motif, which is the sequence pattern signature representing the family of SNARE proteins. For SNARE proteins, this motif is an extended segment arranged in heptad repeats, a structural motif consisting of a seven-amino-acid repeating pattern. The extraction of HMM profiles, which allow identifying evolutionary changes in a set of correlated sequences, returns information on the occupancy and position-specific frequency of each amino acid in the alignment. Using this method, the authors are able to obtain a classification accuracy of at least 95% for nineteen of the twenty HMM profiles generated and to perform a cluster analysis based on functional subgroups.

Nguyen *et al.* [107] have disclosed a model with two-dimensional convolutional network and position-specific scoring matrix profiles for the SNARE proteins identification.The authors used multiple hidden layers for their models, in particular 2D sublayers such as zero padding, convolutional, max pooling and

fully-connected layers with different number of filters. Their model achieves a sensitivity of 76.6%, an accuracy of 89.7% and a specificity of 93.5%.

More recently, in 2020, *Guilin Li* [111] has proposed an hybrid model which combines the random forest algorithm with the oversampling filter and 188D feature extraction method. His work proposes different combinations of feature extraction methods, filtering methods and classification algorithms such as KNN, RF and AdaBoost for the classification of SNARE proteins. Since those results are shown only graphically, it is not possible to have a clear comparison with our results.

5.4 CLASSIFICATION ALGORITHMS

To see how accuracy varies with the use of SNARER descriptors, we employed the same three classification methods presented in [111], given the high performance reported. Thus, we have compared three different ML algorithms: AdaBoost (ADA) K-Nearest Neighbor classifier (KNN) and Random Forest (RF).

- AdaBoost is a machine learning meta-algorithm used in binary classification. AdaBoost is an adaptive algorithm which generates a model that is overall better than the single weak classifiers, adapting to the weak hypothesis accuracy and generating one weighted majority hypothesis in which the weight of each weak hypothesis is a function dependent of its accuracy. At each iteration, a new weak classifier is sequentially added, which corrects its predecessor until a final hypothesis with a low relative error is found [57].

- KNN is a supervised learning algorithm used for predictive classification and regression problems. The basis of the operation of this algorithm is to classify an object based on the similarity between the data, generally calculated by means of the Euclidean distance. In this way, the space is partitioned into regions according to the learning objects similarity. This algorithm identifies a collection of k objects in the training set that are the most similar to the test object. So, a parameter k, chosen arbitrarily, allows us to

identify the number of nearest neighbors, considering the k minimum distances. The prevalence of a certain class in this neighborhood becomes a forecast in order to assign a label to the object [194].

- RF is a supervised learning algorithm that combines many decision trees into one model by aggregation through bagging. The final result of the RF is represented by the class returned by the largest number of decision trees. In particular, the random forest algorithm learns from a random sample of data and trains on random characteristics subsets by splitting the nodes in each tree [79].

5.5 PROTEINS DESCRIPTORS

To use ML approaches to analyze data derived from protein sequences, each amino acid in the protein must have a numerical representation. As a result, a set of numerical parameters that operate as chemical-physical and structural descriptors of proteins are frequently used. The use of a diverse set of properly chosen descriptors improves classification efficiency [142] and allows functional protein families to be predicted [136].

In the literature, over the years, many indices and features of amino acids have been identified for classification methods, such as amino acid composition (AAC), auto-correlation functions [118] or pseudo amino acid composition (PseAAC) [33].

To compare our SNARER descriptors to those already utilized in the classification of SNARE proteins, we chose the four descriptors below.

- GAAC (*Grouped amino acid composition*) groups the 20 amino acids into five groups based on their chemical-physical properties and calculates the frequency for each of the five groups in a protein sequence. Specifically, the five groups are the following: positive charge (K, R, H), negative charge (D, E), aromatic group (F, Y, W), aliphatic group (A, G, I, L, M, V) and uncharge (C, N, P, Q, S, T) [30].

- CTDT (*Composition/Transition/Distribution*) represents the amino acid composition patterns distribution of a specific

chemical-physical or structural property in the protein sequence. The final T represents the transition between three types of patterns (neutral group, hydrophobic group and polar group) of which the percentage of occurrence frequency is calculated [30].

- CKSAAP are sequence-based features which, given a sequence, count all adjacent amino acid pairs, considering k-spaced amino acid pairs. Since there are 20 amino acids, for each value of k (from 0 to 5) there are 400 possible pairs of amino acids, for a total of 2400 features [28].

- 188D features constitute a features vector of which the first 20 represent the frequencies of each amino acid while eight types of chemical-physical properties (such as hydrophobicity, polarizability, polarity, surface tension, etc) allow us to calculate the remaining 168 features. In fact, for each type of property, 21 features are extracted [21].

Our proposed SNARE descriptors are 24 and 19 of them are selected by AAindex, i.e., the Amino Acid index database [95], on the basis of the chemical-physical, electrical and energy charge characteristics of the SNARE proteins. We chose features that consider the propensity of individual amino acids to create helixes, sheets and coils. Since there is mainly an helix structure in the SNARE proteins, we opted to evaluate features related to this structure. Others features are related to solvent accessibility, to the ability to interact with the surrounding environment and energy effects of amino acid residues in SNARE proteins. They are listed in Table 5.1.

The others (i.e., Steric parameter, polarizability, Volume, Isoelectric point, Helix probability, Sheet probability and Hydrophobicity) are the amino acid parameter sets defined by Fauchere *et al.* [51]. We used iFeature [30] for feature extraction of GAAC, CKSAAP and CTDT and MSFBinder [**liu2018model**] for 188D.

Table 5.1: The SNARER descriptors.

Code	Description	Source
ARGP820102	Signal sequence helical potential%	
CHAM830101	The Chou-Fasman parameter of the coil conformation	
CHAM830107	A parameter of charge transfer capability	
CHAM830108	A parameter of charge transfer donor capability	
CHOP780204	Normalized frequency of N-terminal helix	
CHOP780205	Normalized frequency of C-terminal helix	
EISD860101	Solvation free energy	
FAUJ880108	Localized electrical effect	AAindex [95]
FAUJ880111	Positive charge	
FAUJ880112	Negative charge	
GUYH850101	Partition energy	
JANJ780101	Average accessible surface area	
KRIW790101	Side chain interaction parameter	
ZIMJ680102	Bulkiness	
ONEK900102	Helix formation parameters (delta delta G)	
	Steric parameter	
	Polarizability	
	Volume	
	Isoelectric point	Fauchere *et al.* [51]
	Helix probability	
	Sheet probability	
	Hydrophobicity	

5.6 METHODS

5.6.1 *Data Preparation*

We have constructed two datasets, respectively named DUNI and D128. In order to prevent learning bias in classification training, both datasets were utilized to assess each classifier's robustness in both an imbalanced and balanced training environment. SNARE proteins were downloaded from UNIPROT[1]. For this purpose, we selected all the proteins with molecular function "SNAP receptor activity", identified with the unique GENE Ontology [38] alphanumeric code GO: 0005484. The dataset DUNI consists of 276 SNAREs and 806 non-SNAREs. On this unbalanced dataset, we applied the subsampling and ovesampling techniques used in [111]. The balanced dataset D128 is composed of 64 SNARE from UNIPROT and 64 non-SNARE protein sequences downloaded

1 https://www.uniprot.org/

from the PDB database[2]. All SNARE protein sequences in FASTA format[3] have been processed with the CD-HIT tool[4], which returns a set of non-redundant representative sequences as output, in order to create a balanced and non-redundant dataset and improve dataset quality. The incremental clustering approach is used by CD-HIT. It sorts the sequences in length order and constructs the first cluster in which the longest sequence is the representative one in the initial analysis. The sequences are then compared to the representatives of the clusters. The sequence will be grouped in that cluster if the similarity with a representative is greater than a specific threshold. Alternatively, a new cluster with that sequence as the representative can be constructed [113]. The criterion for similarity was set at 25The fraction of comparable residues between two sequences is used to determine sequence similarity. The smaller the sequence similarity, the more likely it is that the collection will contain representative proteins with no redundancy [137].

5.6.1.1 *Training and validation session*

All training sessions were conducted with Weka ML Platform (*Waikato Environment for Knowledge Analysis*), a software environment written in Java which allows the application of machine learning and data mining algorithms [183]. In order to speed-up analysis, an ad-hoc grid, based on the map/reduce paradigm, were used in order to distribute the work across multiple slaves [143]. Both data sets were used as the input for the training step for AdaBoost, KNN e RF classifiers. There were only two possible output classes: SNARE/ NON SNARE. Then, for each training session, we used the following cross-validation values: the range between 10 and 100 for k-fold and between 20 to 80% for hold out. As a result, the *ratio* of the samples in training and validation set is variable. Moreover, based on the parameters configured as in [111] in order to be able to compare with the results of the authors, we set k equal to 1 and Euclidean distance for the `distanceFunction` of KNN; for the AdaBoost algorithm,

2 https://www.rcsb.org/
3 https://zhanggroup.org/FASTA/
4 http://weizhongli-lab.org/cd-hit/

default values are `weightThreshold` = 100 and `numIterations` equal to 10, whilst for RF `numIterations` = 100.

The complete working set was composed of four logical parts: *i)* DUNI non-filtered; *ii)* DUNI oversampled; *iii)* DUNI subsampled; *iv)* D128 non-filtered. For each training session, we generated 10 k-fold variants and 7 hold out variants. Then, for each variant, we computed 100 training sessions of each of the three classifiers for each of the four descriptors. Thus, we distributed up to 836.000 training sessions among the distributed computing environment.

5.6.2 *Performance evaluation of classification algorithms*

We evaluated the ML models (Random Forest, AdaBoost and KNN) on the unbalanced dataset DUNI and on the balanced dataset D128. In order to estimate the prediction performance of the three ML algorithms, accuracy (ACC), sensitivity (SN) and specificity (SP) were used. The formulas of these equations have been shown in Subsection 2.4.3.

Sensitivity is the percentage of positive entities correctly identified. Specificity measures the proportion of negative entities that are correctly identified. In a biological sense, having a TP in our experiment means finding that a protein cataloged as a SNARE is recognized by the classifier as a SNARE.

The four sets of protein descriptors were initially evaluated separately (GAAC, CKSAAP, CTDT and 188D) on the datasets D128 and DUNI and subsequently these feature sets were extended with the SNARER descriptors addition disclosed in this work, here identified as extended classes *"ext"*.

5.7 EXPERIMENTAL RESULTS

On the unbalanced dataset DUNI and the balanced dataset D128, we employed the SNARER descriptors and three different ML methods.

We looked at four feature sets separately (GAAC, CTDT, CSKAAP, and 188D) before combining them with the SNARER descriptors class (designated by *".ext"*). Three criteria were used to assess categorization performance: average accuracy (ACC), average sensibility (SN) and average specificity (SP).

5.7.1 *Results on the unbalanced dataset DUNI*

The experimental results obtained using the DUNI dataset are presented below. The average ACCs for the ML algorithms for the four protein feature sets GAAC, CTDT, CKSAAP, and 188D are in the range of 76% to 94.9%. For the graphical comparison of the three ML approaches, histograms are employed in Figure 5.2.

Table 5.2: Performance of average ACC on the DUNI dataset.

	Accuracy		
	RF	KNN	ADA
GAAC	76.1%	85.1%	77.9%
GAAC.ext	**90.4**%	**91**%	**86.1**%
CTDT	76.1%	83.1%	76.7%
CTDT.ext	**91.5**%	**92.6**%	**81.6**%
CKSAAP	91.1%	90.04%	83.7%
CKSAAP.ext	**91.7**%	90.02%	**87.4**%
188D	93.9%	94.8%	88.1%
188D.ext	**94.1**%	**94.9**%	**88.7**%

In combination with all of the protein feature descriptors studied, the introduction of the SNARER class results in a significant improvement, as shown in Table 5.2. Overall, the KNN model, the 188D feature set, and the SNARER class combination produce the best average accuracy. The best average SP is obtained with this combined model, while the best average SN is obtained with the RF model trained independently utilizing both GAAC and CTDT features (see Table 5.3).

In the enlarged classes with the additional descriptors, SN lowers imperceptible for RF, but remains unchanged for the CKSAAP method. SP for RF, on the other hand, rises with the extended classes, particularly for the GAAC and CTDT descriptors.

In the enlarged classes referred to as GAAC and CTDT, the SN of KNN increases greatly, whereas CKSAAP and 188D stay largely unaltered. The SP of KNN follows the same pattern, with a modest improvement in the extended 188D class. We see an increase in SN for the AdaBoost algorithm, mainly for the

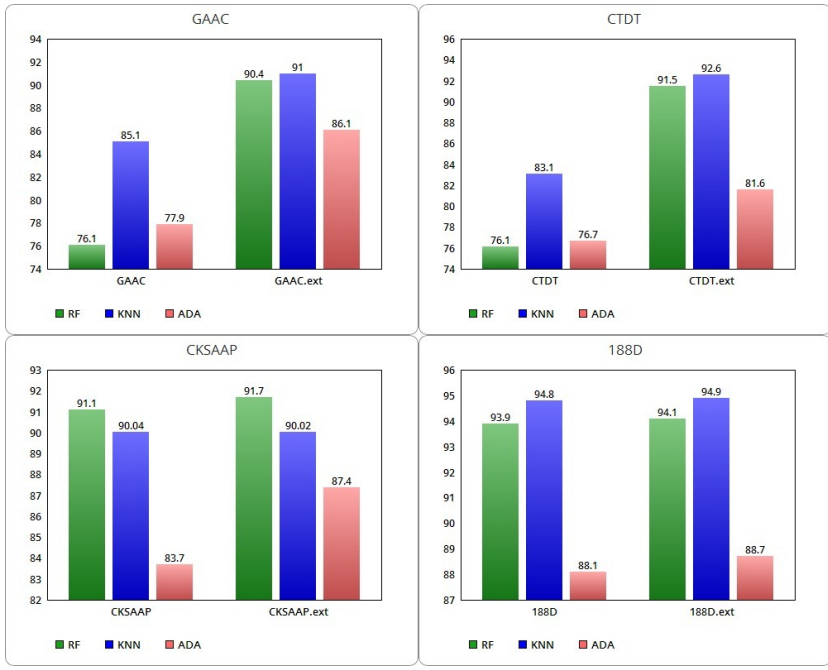


Figure 5.2: Comparison between GAAC, CTDT, CKSAAP and 188 D ACC with related extended classes with SNARER (on DUNI dataset).

Table 5.3: Performance for average SN and SP on the DUNI dataset.

	Sensitivity			**Specificity**		
	RF	KNN	ADA	RF	KNN	ADA
GAAC	**99.8**%	90.3%	83.6%	7%	7%	**61**%
GAAC.ext	97.2%	**94.5**%	**94.8**%	**70.7**%	**80.6**%	60.6%
CTDT	**99.8**%	89.1%	83.3%	7.1%	65.6%	**57.6**%
CTDT.ext	96.6%	**94.6**%	**91.1**%	**76.4**%	**87**%	54%
CKSAAP	97.8%	98%	89.9%	71.7%	66.7%	65.5%
CKSAAP.ext	97.8%	98%	**92**%	**74**%	66.7%	**74**%
188D	**97**%	**96.6**%	92%	85%	89.5%	76.7%
188D.ext	96.8%	96.5%	**92.4**%	**86.3**%	**90.1**%	**78**%

extended GAAC and CTDT classes, but a loss in SP. Instead, for the extended classes CKSAAP and 188D, the SP ADA rises.

The usage of extended classes using our SNARER descriptors improves accuracy for the GAAC, CTDT, CKSAAP and 188D classes of all three ML models on the unbalanced dataset, with the exception of KNN trained with CKSAAP. All selected ML algorithms achieve SN greater than 83%, with the best SN of 99.8% RF achieved by GAAC and CTDT without extension.

The SN settles in a region between 91.1% of the ADA algorithm with the CTDT class and 98% of the KNN algorithm with the extended CKSAAP class by incorporating the SNARER class for all four feature sets. Without the SNARER's descriptor extension, the SP ranges from a minimum of 7% of RF and KNN algorithms for the GAAC class to a maximum of 89.5% of KNN trained with the 188D feature set.

With the SNARER class addition, an SN of 54% of ADA with CTDT feature set is obtained at a maximum of 90.1% of KNN trained on the dataset with 188D feature set. More specifically, the KNN model using the 188D extended class with SNARER descriptors, achieves better performance in all metrics except for SN, where the RF model trained with the GAAC features obtains the highest value. In particular, the KNN model employing the 188D extended class with SNARER descriptors outperforms the RF model trained with the GAAC features in all metrics except SN, where the RF model outperforms the KNN model.

Finally, on the unbalanced DUNI dataset, the novel SNARER descriptors class assures an increase in terms of ACC when used in conjunction with all four evaluated feature sets, as well as a significant improvement in SN and SP for the tested ML methods.

5.7.1.1 *Results on the unbalanced dataset DUNI with oversampling and with subsampling*

We used subsampling and oversampling techniques because the dataset DUNI is imbalanced.

The SNARER class improves accuracy significantly with the oversampling strategy on the DUNI dataset, particularly so for the enlarged GAAC and CTDT classes for the three ML models RF, KNN and ADA, while the contribution to the CKSAPP and 188D feature sets remains mostly constant (as shown in Table 5.4). The average SN and average SP determined for RF,

Table 5.4: Performance of the average ACC on the DUNI dataset with oversampling and subsampling.

	Oversampling			**Subsampling**		
	RF	KNN	ADA	RF	KNN	ADA
GAAC	94.7%	96.3%	73.12%	75.2%	79.2%	72.6%
GAAC.ext	**98.03%**	**98.44%**	**85.02%**	**91.8%**	**86.4%**	**82.6%**
CTDT	93.9%	96.1%	70.4%	74.6%	78.1%	71.7%
CTDT.ext	**98%**	**98%**	**86.3%**	**90.6%**	**89.7%**	**86.4%**
CKSAAP	**99.07%**	**98.67%**	84%	**93.1%**	**84.4%**	83.5%
CKSAAP.ext	99.01%	98.6%	**89.1%**	79%	84.2%	**87.3%**
188D	98.5%	98.90%	89.5%	93.1%	**95%**	86.6%
188D.ext	98.5%	**98.95%**	**89.6%**	**93.5%**	94%	**89.3%**

Table 5.5: Performance for average SN and SP on the DUNI dataset with oversampling.

	Sensitivity			**Specificity**		
	RF	KNN	ADA	RF	KNN	ADA
GAAC	91.9%	95%	74.9%	97.5%	97.6%	71.3%
GAAC.ext	**96.6%**	**97.8%**	**88.4%**	**99.4%**	**99.1%**	**81.6%**
CTDT	88.9%	94%	68.8	98.8%	98.2%	72%
CTDT.ext	**96.4%**	**96.9%**	**78.4%**	**99.5%**	**99.2%**	**94.3%**
CKSAAP	**99%**	**99.2%**	80.8%	99.2%	98.2%	87.2%
CKSAAP.ext	98.7%	99.1%	**86.2%**	**99.3%**	98.2%	**92%**
188D	97.5%	98.3%	**90.2%**	**99.7%**	99.5%	88.8%
188D.ext	**97.7%**	98.3%	89.4%	99.3%	**99.7%**	**89.9%**

KNN and ADA show the same pattern (see Table 5.5). Applying the subsampling technique to the DUNI dataset, we observe the same trend for SN but with a slight decrease, around 2% -4%, of the values when considering the extended classes CKSAAP and 188D. The same decrease value is also present for the average SPs of the same classes (see Table 5.6).

Table 5.6: Performance for average SN and SP on the DUNI dataset with subsampling.

	Sensitivity			**Specificity**		
	RF	KNN	ADA	RF	KNN	ADA
GAAC	75.7%	76.1%	73.6%	74.6%	82.2%	71.7%
GAAC.ext	**88.8%**	**85.5%**	**80.8%**	**94.9%**	**87.3%**	**84.4%**
CTDT	78.3%	76.4%	73.9%	71%	79.7%	69.6%
CTDT.ext	**86.6%**	**88.4%**	**81.9%**	**94.6%**	**90.9%**	**90.9%**
CKSAAP	**90.9%**	**98.6%**	83.3%	**95.3%**	70.3%	83.7%
CKSAAP.ext	76.4%	98.2%	**83.7%**	81.5%	70.3%	**90.9%**
188D	90.9%	**95.3%**	88%	**95.3%**	94.6%	85.1%
188D.ext	**92%**	93.1%	**88.8%**	94.9%	**94.9%**	**89.9%**

5.7.2 *Results on the balanced dataset D128*

The classification results achieved on the balanced dataset D128, with and without the addition of the SNARER descriptors, are presented below. Table 5.7 shows the average accuracy of the ML algorithms in the balanced D128 dataset without taking into account the SNARER descriptors. In addition, histograms are depicted graphically in Figure 5.3: RF varies from a minimum of 71.1% with the use of the GAAC class to a maximum of 95.4% with the 188D class; KNN settles between a minimum of 64.2% with the use of GAAC to a maximum of 90% with the 188D class; ADA varies from a minimum of 70% with GAAC to a maximum of 90.2% trained on the 188D class. Extended classes with SNARER descriptors shift these average ACC rates. In particular, RF varies from a minimum of 84% using the extension with GAAC to a maximum of 95.3% with the 188D class. KNN starts from a minimum of 65.4% with the extended GAAC class and reaches a maximum of 88.6% with the extended 188D class. ADA varies in a range between 84% with the GAAC.ext class to a maximum of 90% with the combined class 188D. When comparing the evaluated average ACCs, the SNARER class addition enhances classification performance when compared to the GAAC, CKSAAP and CTDT feature sets, whereas the average

ACCs for the 188D class decrease slightly. To figure out why there has been a drop, more research should be done. The RF algorithm, in particular, produces the best classification results.

Table 5.7: Performance of average ACC for the D128 dataset.

	Accuracy		
	RF	KNN	ADA
GAAC	71.1%	64.2%	70%
GAAC.ext	**84%**	**65.4%**	**84%**
CTDT	73.4%	66.4%	70.3%
CTDT.ext	**88%**	**68.7%**	**84.1%**
CKSAAP	92.2%	72.4%	80.7%
CKSAAP.ext	**92.3%**	**74.1%**	**89.4%**
188D	**95.4%**	**90%**	**90.2%**
188D.ext	95.3%	88.6%	90%

Table 5.8: Performance for average SN and SP on the D128 dataset.

	Sensitivity			**Specificity**		
	RF	KNN	ADA	RF	KNN	ADA
GAAC	80.1%	**65.7%**	74.5%	62.2%	63%	65.4%
GAAC.ext	**84%**	62.2%	**88.6%**	**83.9%**	**69%**	**79.2%**
CTDT	74.7%	**70.4%**	70%	72.2%	62.3%	70.5%
CTDT.ext	**87.6%**	64.7%	**84.7%**	**88.3%**	**73%**	**83.4%**
CKSAAP	89.7%	55.4%	80.2%	95%	89.4%	81.3%
CKSAAP.ext	**90.1%**	**57%**	**89.5%**	95%	**91.2%**	**89.4%**
188D	**95.7%**	**89%**	88.5%	95.1%	**91%**	**92%**
188D.ext	95.5%	88%	**88.8%**	95.1%	89.2%	91.2%

With the combinated feature sets with SNARER descriptors , we can see that the SN of RF algorithm grows with GAAC and CTDT, while the SN of the other two descriptor classes remains essentially the same (see Table 5.8). SP grows as well, exhibiting the same pattern. For the KNN algorithm, SN decreases for the

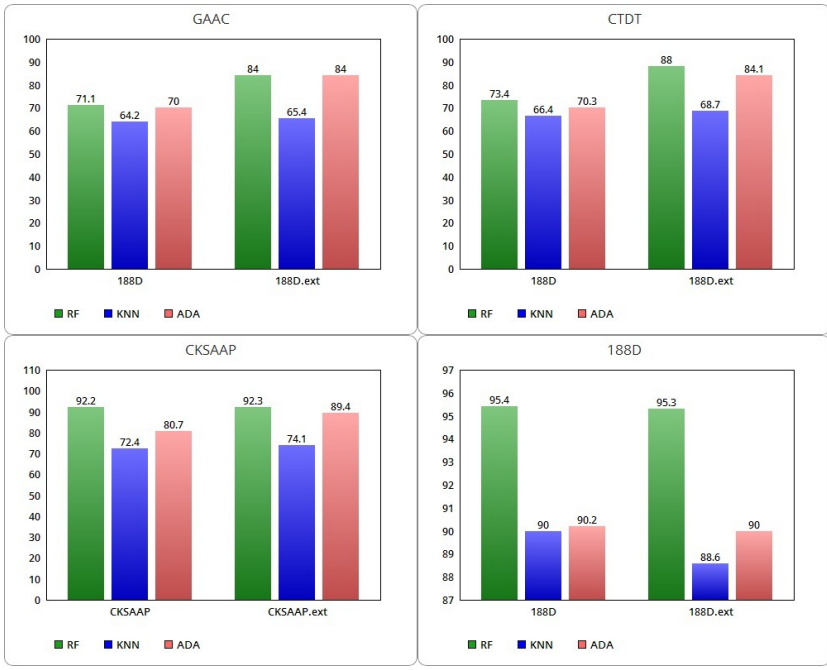


Figure 5.3: Comparison between GAAC, CTDT, CKSAAP and 188D ACC with related extended classes with SNARE (on D128 dataset).

GAAC and CTDT classes by 3% and 1% for the 188D class while it increases by 2% for the CKSAAP class. The SP of KNN instead increases for all classes except 188D, with a decrease of about 2%. ADA improves in terms of SN on all extended classes, while it decreases in SP by 0.8% when applied to the extended class 188D.

5.7.3 *Comparison between the DUNI and the D128 datasets*

Experiments on unbalanced or balanced datasets have an impact on the automated learning of different ML algorithms. In reality, it has been discovered that when tests are run on an unbalanced dataset, more accuracy is gained because each test sample is classified towards the majority class [190]. As a result, using a balanced dataset for training tests can result in better classification predictions. In the case of binary classifications, the coefficient of correlation between the true class and the expected

class can be calculated, dealing with them as two binary variables. Following the introduction of the SNARER descriptors, we used the Matthews Correlation Coefficient (MCC) [18] to compare the DUNI and D128 datasets because the ACC computation is susceptible to the imbalance class. In this context, we started from the hypothesis that the proportion of correct predictions (accuracy) are not useful when the two classes have different sizes. In this case, the use of MCC is useful. It represents a quality measure also in cases where the datasets have different sizes. MCC is a classification quality metric that ranges from $[-1;1]$. A perfect forecast is indicated by an MCC value of 1. A perfect negative correlation is represented by a value of -1, whereas a value of 0 indicates that the classifier produces just a forecast that is no better than a random one (see Equation 5.1). So, MCC considers all four values in the confusion matrix (TP, TN, FP and FN) and a high value (around 1) indicates that both classes are adequately covered, even if one is disproportionately under (or over) represented.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (5.1)$$

Table 5.9: Comparision of MCC for the DUNI and D128 datasets.

	Matthews correlation coefficient			
	Dataset	MCC RF	MCC KNN	MCC ADA
GAAC.ext	DUNI	**0.74**	**0.76**	0.61
	D128	0.69	0.32	**0.70**
CTDT.ext	DUNI	0.77	**0.81**	0.49
	D128	0.77	0.39	**0.70**
CKSAAP.ext	DUNI	0.77	0.73	0.69
	D128	**0.86**	0.53	**0.80**
188D.ext	DUNI	0.84	**0.87**	0.70
	D128	**0.91**	0.81	**0.81**

In Table 5.9, the MCC metrics for RF, KNN and ADA trained on the DUNI and D128 datasets with the extended descriptors

classes are compared. The MCC (see Figure 5.4) of RF improves on the balanced dataset, except for a decrease with the GAAC discriminant features and for no change on the CTDT class. The MCC of KNN is lowered for all combined descriptors, significantly for GAAC, CTDT and CKSAAP. In contrast, ADA's MCC is significantly improved in all four conditions. As a result, we can see how the values of MCC reflect the quality of the classifier input data. Only if the classifier successfully predicted the majority of positive data instances and the majority of negative data instances, MCC can generate a high score. In the presence of DUNI, which is a negatively imbalanced dataset, we have high values in terms of ACC, SN and SP compared to the balanced dataset. Since it ignores the proportion of positive and negative items, accuracy can produce misleading values for unbalanced datasets [31]. In Table 5.9, we showed how many MCC values are greater when we evaluate the algorithms on a balanced dataset with no positive and negative samples imbalance. In some circumstances, MCC values remain constant, owing to the classifier's ability to produce accurate predictions regardless of the *ratio* between classes. The MCC is lower in the case of the KNN algorithm, which reflects the worst performance measured by other measures.

In Table 5.10, we presented the comparison between our proposed method and the literature. The method by [99] is based on Hidden Markov Models (HMM), sequence alignment and phylogenetic tree reconstruction in order to classify SNARE proteins. Nguyen *et al.* [107] used a model with 2D-CNN and position-specific scoring matrix profiles, while the study of Guilin Li [111] has suggested a hybrid model that incorporates the random forest algorithm, oversampling filter and the 188D feature extraction approach. How can we see in Table 5.7, on the basis of the comparison between extended classes of the four descriptors, our best results are the combination of SNARER descriptors with CKSAAP feature on the dataset D128 with 92.3% of accuracy, 90.1% for sensitivity and 95% for specificity with the RF. Our highest performance on the D128 dataset with SNARE descriptors is achieved by the RF algorithm in combination with the 188D features. We did not consider this last result as a better result, since there is a slight decrease in the metrics used for

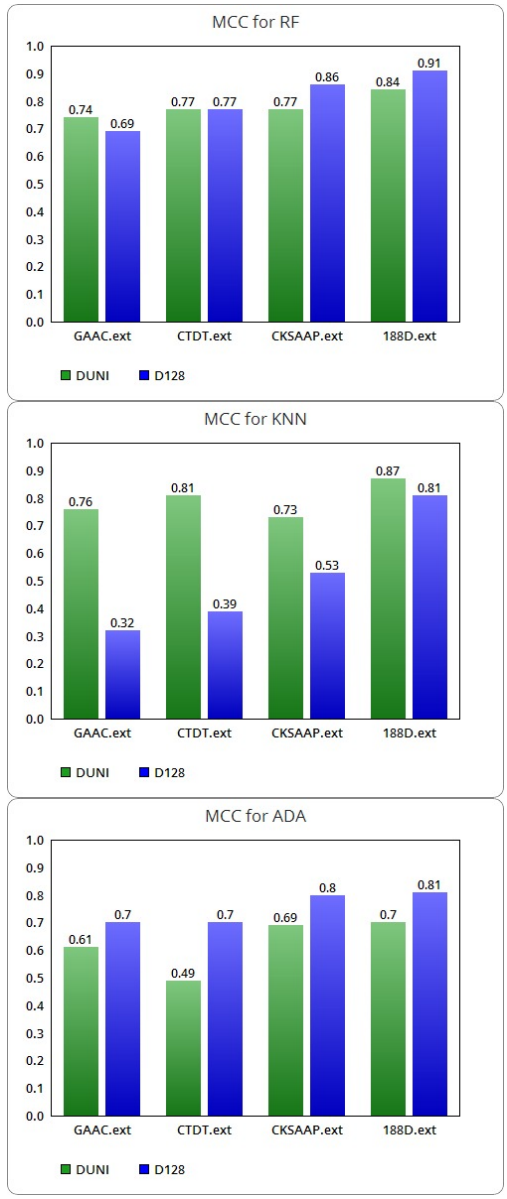


Figure 5.4: Graphic visualization of MCC for RF,KNN and ADA algorithms.

performance measures when we use the extended class 188D in comparison with not extended class 188D.

188D features include the 20 characteristics about frequencies of each amino acid and 168 features based on using eight types of

chemical-physical properties. These features probably strengthen the biological properties of the proteins, allowing to reach high levels of the tested classification algorithms. Further studies are needed to understand the intrinsic reasons for the improvement or decay of some parameters when using 188D features.

Table 5.10: Comparison with reference literature

Authors	**Methods**	**ACC**	**SP**	**SN**
Kloepper et al.	HMM	95%	-	-
Nguyen et al.	2D-CNN	89.7%	93.5%	76.6%
Guilin Li	188D-RF-oversample	90-95%	95-100%	75-80%
Our methods				
(highest value)	RF-188D.ext (D128)	95.3%	95.1%	95.5%
(best value)	RF-CKSAAP.ext (D128)	92.3%	95%	90.1%

5.8 RESULTS AND DISCUSSION

In order to investigate the role of balanced and unbalanced training in the classification of SNARE proteins, we examined four sets of protein descriptors with and without the addition of SNARER descriptors. As a result, we compared the performance of three machine learning algorithms (RF, KNN and ADA) on homogeneous and heterogeneous datasets. The ACC, SN and SP average values were used to evaluate the ML models. The performance of the ML algorithms improved on both datasets when the SNARER descriptors were extended to the feature sets employed, according to our findings. With the addition of SNARER descriptors to the 188D class, this increase is much bigger for RF, KNN and ADA algorithms. Our best results with the RF algorithm and the extended class *CKSAAP.ext* on the balanced and non-redundant dataset D128 are 92.3% of ACC, 90.1% for SN and 95% for SP. The ADA algorithm profited from improved performance on the balanced dataset when the MCC for RF, KNN and ADA was evaluated on both datasets trained with enlarged feature sets. KNN, on the other hand, has deteriorated in terms of performance, except the 188D class, attaining a higher value.

In particular, the algorithms trained on the balanced dataset yield a better MCC, particularly for RF and much more so for ADA, which recovers both in terms of ACC, SP and SN in all the tests studied. In comparison to the other algorithms evaluated, KNN appears to have lesser performance in terms of MCC.

6

GO TERMS VISUALIZATION

In this Chapter, we suggest a human-interaction system for viewing similarity data for proteins/genes of Alzheimer and Parkinson disease based on the three functions of the *Gene Ontology* (Cellular Component, Molecular Function and Biological Process). We start with an introduction about the importance of the representation of multilevel data and the Gene Ontology (see Section 6.1 and Section 6.2), then we present the related works in this field (Section 6.3). We discuss the methods in Section 6.4 and in particular the used similarity measures for the examined proteins in Section 6.4.2. Then, the proposed system of visualization follows in Section 6.6.

6.1 INTRODUCTION

The graphical depiction of information and data is known as data visualization. Data visualization tools make it easy to see and analyze trends, outliers and occurrences in the data by using visual elements like charts, graphs and maps. In the world of big data, where data visualization tools and technologies allow you to examine massive volumes of data, this is becoming increasingly crucial.

In recent years, having an omic vision has become increasingly important in order to characterize biological systems at ever-more-granular levels. The purpose of omic sciences is to produce relevant information that can be used to characterize and comprehend biological systems [178]. We refer to genomics, transcriptomics, proteomics and metabolomics as *omic sciences*, which encompass a wide spectrum of biomolecular disciplines distinguished by the suffix *-omics*. Biological information is multifaceted and extremely interconnected. The present challenge is to provide a more detailed integrative understanding of the dynamics of cellular processes in a cell or organism that is rich in biological and spatio-temporal data [180]. As O'Donoghue *et*

al. point out, the display of biological data has grown increasingly important in the Biosciences because it allows researchers to comprehend diverse data more quickly and easily [134]. One of the most pressing problems in omic data analysis right now is the inability to study links between multi-omic states in order to incorporate and combine higher-level expertise [196].

Due to interclass dissimilarities and inter-class similarity [9], protein similarity visualization that is not based on sequence alignment might be difficult. Clustering and Machine Learning approaches may be ineffective in extracting interdependencies across objects [70]. This fact frequently prevents us from creating a clear visual representation of the data.

When a typical clustering technique fails, we want to show how a human-assisted dynamic graph construction can help abstract functional links between proteins and provide a clear data representation.

6.2 GENE ONTOLOGY

The *Gene Ontology* (GO) is a bioinformatics project, used for gene enrichment analysis, that supports the standardization of biological information about attributes of genes and gene products through the use of ontology. It is organized as an acyclic oriented graph, with a word or strings and a unique alphanumeric code for each GO-term [65]. The Gene Ontology is based on two types of relationships between objects: *instances* and *part of*. All organisms have three biological domains that can be thought of as vocabularies that are structured and controlled, indicates as:

- *Biological Process* (BP) which refers to all the activities that occur within an organism as a result of a well-organized set of molecular processes;
- *Molecular Function* (MF) which describes the molecular processes that take place in an organism;
- *Cellular Component* (CC) relates to the position of the subject entity at the cellular and/or subcellular level.

6.3 STATE OF THE ART ABOUT PROTEIN INFORMATION VISUALIZATION

Several web interfaces, present in the scientific literature, may query the Gene Ontology terms.

QuickGO enables us to locate and display GO terms, as well as provide a list of correspondence results based on the query of the user. A directed acyclic graph (DAG) containing a single GO word and its associated terms and annotations is returned by this tool. JavaScript, Ajax and HTML were used to create it. On-the-fly statistics, including interactive graphs and views of term placement tables, are accessible, demonstrating which terms are commonly mentioned at the same time. The user can get a subset of annotations based on several factors (e.g., specific protein, Evidence Codes, Qualifier Data, Taxonomic Data, Go Terms) [16].

GOrilla [48] detects enriched GO words in ordered lists of genes using simple, clear and informative graphics, without needing the user to supply specific targets or backdrop sets. It is a GO analysis tool that uses a statistical approach with adjustable thresholds to find GO terms that are considerably overrepresented at the top of a gene list (very useful when genomic data can be represented as a classified list of genes). The findings of the study are given in a hierarchical framework, allowing for a clear view of the GO terms.

Blast2GO is an interactive tool that facilitates functional genomic research in non-model species. It is a data-sequence-based tool with a high level of user engagement that combines high-performance analysis algorithms and assessment statistics. On direct acyclic graphs, similarity searches yield results [37].

NaviGO [189] uses six different scores to assess semantic similarity and GO associations: Resnik, Lin, the relevant semantic Similarity score for semantic similarity, *Co-occurrence Association Score* (CAS), *PubMed Association Score* (PAS) and *Interaction Association Score* (IAS). There is also a `Funsim` score for functional similarity.

More recently, the open-source software *AEGIS* allows us to visually explore GO data in real-time, with the whole GO dataset as input. Any Go term can be used as the anchor, with a DAG

representing the root, leaf, or waypoint. Each source can contain all the descendants of the anchor term, the leaves will only have ancestors and the waypoint anchors will contain both ancestors and descendants.

6.4 METHODS

6.4.1 *Dataset*

For this work, we considered two diseases: *Alzheimer* and *Parkinson*, the two most common neurodegenerative pathologies.

Alzheimer's disease (AD) is a form of degenerative dementia that occurs after 65 years. The formation of senile plaques and the intracellular aggregation of *tau* protein are associated with the deposition of an Aβ peptide B in this disorder [46]. Parkinson's disease (PD) is the second most common neurodegenerative condition in the elderly, characterized by neuronal loss in the substantia nigra and the production of neuropathological α-synuclein aggregates [144]. These pathologies show similar neurodegeneration mechanisms supported by scientific evidence with genetic, biochemical and molecular studies. Pathological pathways involving α-synuclein and *tau* proteins, oxidative stress, mitochondrial dysfunction, iron pathway and *locus coeruleus* are among these findings [195]. They were chosen as an example for our search workflow because their pathogenic mechanisms are similar. Intra- and extra-class overlaps are introduced by this feature, which can fool traditional clustering techniques.

Protein datasets for AD and PD, belonging to *Homo Sapiens*, were downloaded from UNIPROT [177]. In order to remove all duplicates, data cleaning was performed. The reference gene for each UNIPROT ID has also been retrieved and linked to the STRING database. The STRING database enables us to consider any protein-protein interaction (PPI) based on a score derived from experimental evidences [172]. We found 216 genes for Alzheimer's disease and 137 genes for Parkinson's disease.

6.4.2 *Similarity Measures*

To compute pairwise semantic similarities, we used two types of metrics: *Lin* and *Wang*.

Lin's measure is based on *information content* (IC). IC stands for the negative log of a concept's probability. The ratio between the quantity of "common information" and the amount of "total information" in the descriptions of an item pair is computed using this method. The similarity of two items is represented by this ratio [117]. In this scenario, the similarity of the knowledge content of the GO keywords for each protein dataset, proteins of AD and proteins of PD, may be measured using this method. The estimation is based on the frequency of two GO words and their nearest common ancestor in a corpus of GO annotations. The term *Least Common Subsammer* (LCS) suggests the most basic definition that two concepts share as an ancestor.

For Lin similarity, we can consider the following Equation 6.1:

$$sim_{lin} = \frac{2 * IC(lcs(c_1, c_2))}{IC(c_1) + IC(c_2)} \tag{6.1}$$

where c_1 *and* c_2 are two concepts, *IC* is the information content and *lcs* is the function that computes the least common subsammer.

The concepts represented by the GO terms corresponding to the BP, CC and MF domains are represented by c_1 and c_2 in our experiment. For both AD and PD, the similarity is evaluated across all proteins in the pathological reference dataset.

The Wang method is based on a *graph-based* semantic similarity. By aggregating the terms of their ancestors in a GO graph, the GO terms are turned into a numeric value [185].

Given two GO terms, A and B, we can represent $DAG_A = (A, T_A, E_A)$ and $DAG_B = (B, T_B, E_B)$, where T_n is the set of GO terms including the term n and all of its ancestor terms in the GO graph while E_n are the semantic relations represented as edges between the GO terms. The semantic similarity between these two terms are calculated as in Equation 6.2:

$$S_{GO}(A, B) = \frac{\sum_{t \in T_A \cap T_B} S_A(t) + S_B(t)}{SV(A) + SV(B)} \tag{6.2}$$

where S_A(t) and S_B(t) denote the S-value of a GO term t related to term A and term B. Wang measures the semantic meaning of GO term n, SV(n), after obtaining the S-values for all terms in DAG_n with the Equation 6.3, represented below:

$$SV(n) = \sum_{t \in T_n} S_n(t) \tag{6.3}$$

We explored two methods for calculating semantic similarity.

In the first scenario, we calculated the similarity between proteins from Alzheimer's disease and proteins from Parkinson's disease for all three ontology gene domains (BP, MF and CC). For this purpose, we considered both Lin's similarities and Wang's method but, as an example, in this work we only show the results concerning the similarity of Lin. Subsequently, we clustered the data obtained for both similarity measures in BPs, CCs and MFs domains for AD and PD with the K-means algorithm, trying with n=3 and n=5 clusters.

In the second scenario, we estimated the similarity between the two sets of protein data of disorders about BPs, DCs and MFs domains using the Wang and Lin methods in order to compare these measures.

6.5 K-MEANS VISUALIZATION

K-means is a partitioning clustering technique that divides a set of objects into K groups depending on their attributes and it is one of the most extensively used [119]. A cluster is essentially a collection of data that has been grouped together based on simlarities. The division into K clusters is done *a priori*, based on the goal to be achieved or using heuristic techniques and the clusters represent the number of centroids required by the dataset. A centroid is a real or imaginary point that symbolizes the center of a cluster and it is modified with each algorithm iteration. The procedure is composed by four steps:

- *Step 1:* determine the value of K;
- *Step 2:* randomly select K points as initial centers of the clusters;

- *Step 3:* assign each new point to the cluster with the closest Euclidean distance to its center. Formally, if c_i is a centroid of the set of centroids C then each point x will be assigned to a cluster based on the following equation (Equation 6.4):

$$\arg\min_{c_i \in C} dist(c_i, x)^2 \tag{6.4}$$

  where *dist(.)* represents the Euclidean distance;

- *Step 4:* recalculate the updated cluster centers by averaging the points associated with each cluster (Equation 6.5):

$$c_i = \frac{1}{|S_i|} \sum_{x_i \in S_i} x_i \tag{6.5}$$

  where *Si* is the cluster's set of points.

Steps 3 and 4 are repeated until a convergence is attained. The method enables fast execution while allowing the data to group and move around freely. Due to the goal of this research, we limited the max number of clusters to five. No PCA techniques were used. This constraint is connected to the primary premise that a lesser number of clusters can be beneficial for biological scope when the concept of similarity associated with the GO is addressed. When K is lower, the K-means allows us to save the information but not to view it intuitively. The end user would be unable to appropriately evaluate the results without a clear display of the data. In order to convert such data into knowledge, it must be represented as clearly as possible.

6.5.1 *Results with k-means*

Figures 6.1-6.4 report how the GO objects are partitioned regarding the BP and MF features for AD and PD, with K equal to 3 and 5. The axis reports the distance between each item to its centroid.

Clustering with the K-means algorithm causes visually misleading and uninformative overlaps, according to our findings. This is related to cluster density, which involves extremely short intra-cluster distances.

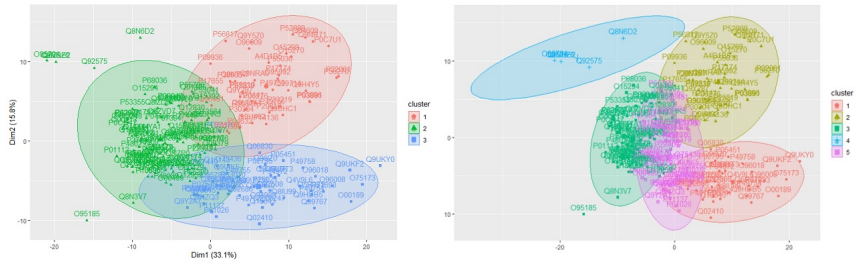


Figure 6.1: K-means for BP for AD with Lin's measure (K=3 on the left and K=5 on the right).

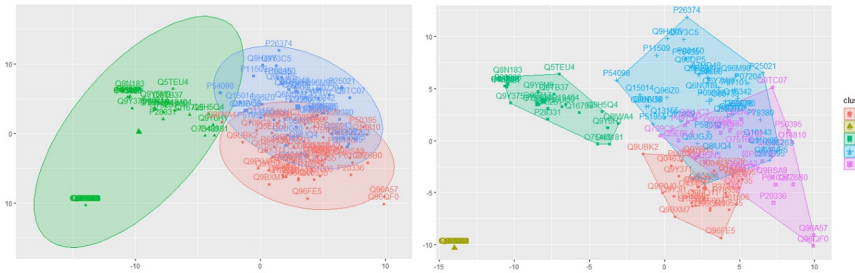


Figure 6.2: K-means for BP for PD with Lin's measure (K=3 on the left and K=5 on the right).

6.6 ALTERNATIVE APPROACH TO VISUALIZE GENE ONTOLOGY TERMS

We propose a *dynamic build cyclic distance graph* (DCDG) to visualize and convey knowledge about GO terms in order to address the problems of overlaps in visualization. Our goal is to visualize the GO links more clearly than previous visualization approaches such as clustering or partitioning. To allow the user to explore this interconnectedness, we created a web-based workspace using Javascript and SigmaJS, a JavaScript library dedicated to graph drawing[1].

The work environment is intended to be as clean as possible. It begins as a blank web app with a single callable overlay menu in the upper left corner that allows users to search datasets for the entry point protein. The input datasets were the BP, CC and MF distance matrices based on similarities of Lin and Wang.

1 SigmaJS: https://sigmajs.org

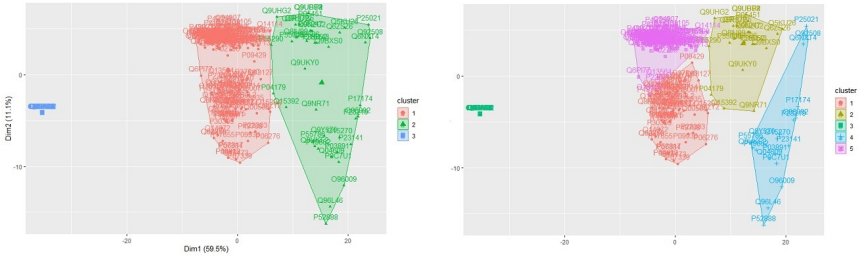


Figure 6.3: K-means for MF for AD with Lin's measure (K=3 on the left and K=5 on the right).

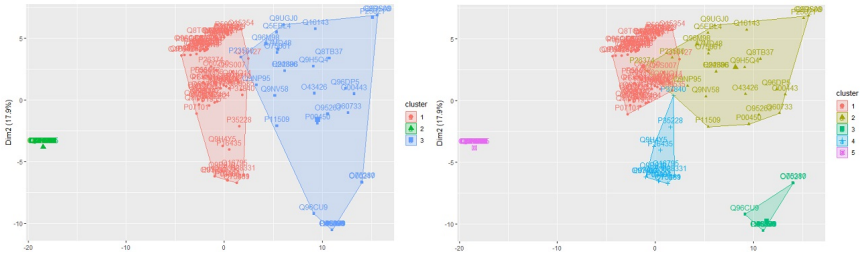


Figure 6.4: K-means for MF for PD with Lin's measure (K=3 on the left and K=5 on the right).

When the entry protein is chosen, it becomes the root of the graph. Users can right-click on any graph node to bring up a context menu (as shown in Figure 6.5) where they can choose an extension (explosion) action for the node.

For this contribution, we established three types of extensions, each of which is tied to a single dataset: BP, CC and MF, whose definitions are those specified by the three vocabularies of the GO. On the arcs between each node pair, the distance between them is written. The reading key for displaying protein through the dynamic build cyclic distance graph is this value, which establishes the similarity measure. These values connect proteins, allowing us to explore the network while considering the similarity values between biological processes, molecular functions, and cellular components. The distance value can also be used to divide nodes into spaces. The ForceAtlas2 algorithm is used to avoid overlapping between near nodes. In particular, we used ForceAtlas2 embedded into SimgaJS [88].

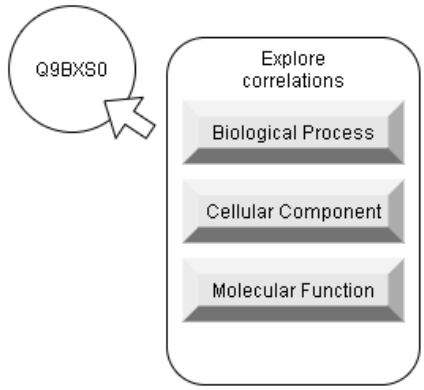


Figure 6.5: The contextual menu is available for each node.

For force-directed graphs, we used the layout ForceAtlas2 algorithm. Using the distances between nodes as edge weights, this approach allows us to place each node in relation to the others. Because of this, the position of a node must always be compared to the positions of other nodes. Because the structural proximity present in the original datasets is transformed to visual proximity, the primary advantage of adopting ForceAtlas2 for the representation of protein graphs is an easier view of the structure.

In order to better empathize the functionality distance between GO, we defined a spatial distance SD with the following equation (Equation 6.6). Given two nodes, A and B and their own distance d:

$$SD = log_e(d) \tag{6.6}$$

where d is the distance and the log_e is the natural logarithm with the number of Nepero as base. SD is solely utilized in the rendering processes for graphical purposes. Figure 6.6 shows no linear proportionality into edge lengths: see the distance between (Q8IZY2, Q9BS0) and (Q93045, Q9BS0). Still, for graphical purposes, we defined a threshold th_i as the mean of all the distances into the dataset i used for node expansion. As an example, given the node Q9BXS0 (see Figure 6.6), the threshold for the protein Q9BXS0 is the mean of the edge's weight between Q9BXS0 and

the related nodes. When the distance *SD* between two nodes A and B is greater than *th_i*, then node A and B are considered belonging to a different cluster. A dotted line renders each class separation. For the first prototype of the proposed method, see the Prototype Page[2].

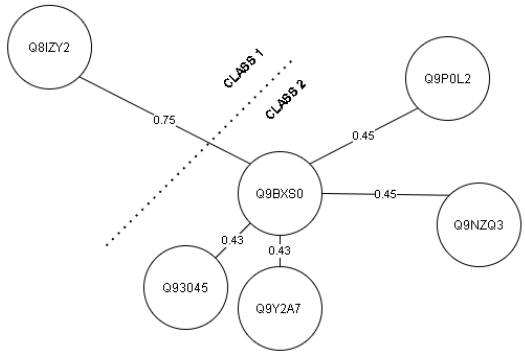


Figure 6.6: The result of Q9BX80 expansion by BP dataset.

A symmetry (or distance) matrix in tab-separeted values (TSV) format is required as an input. After clustering, the table of coordinates between the individual proteins, displayed graphically as dynamic dots, can be downloaded. The prototype is constantly being modified to ensure that the user has complete control over the visualization process.

6.6.1 *Results with dynamic build cyclic distance graph*

Protein data based on Lin's computed similarity were used to test our approaches. To construct our view of node expansion, we used the similarity matrices related to the G9BXS0 protein and we identify the proteins in its neighborhood. Figure 6.6 shows the BP expansion with the DCDG view for the node G9BXS0, a protein produced by *COL25A1* gene for *Homo Sapiens* organism. This protein inhibits the fibrillization of β-amyloid peptide, which constitutes amyloid plaques present in Alzheimer's disease. It also assembles the amyloid fibrils in aggregates which are resistant to the demerger mechanisms [55]. The DCDG view

2 https://smcovid19.org/simtest

enables the user to see and understand proteins belonging to two distinct BP classes: **CLASS 1**, which is concerned with the organization of fibrils, microtubules and cytoskeleton structures and **CLASS 2**, which is concerned with many biological processes such as signaling pathways and positive and negative regulation of cellular and chemical complexes.

Figure 6.7 shows the successive expansion of Q8IZY2 and Q9P0L2 proteins.

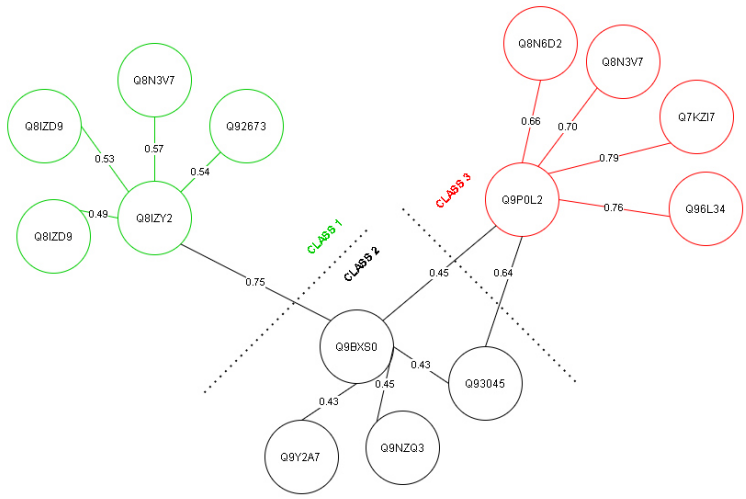


Figure 6.7: The result of Q8IZY2 and Q9P0L2 expansion by BP dataset.

Due to distances, a new class was identified by the system (**CLASS 3**). In terms of biological significance, the visualization clearly illustrates that, in comparison to prior classes, the extra third class stresses greater participation of proteins indicated in various biological processes. This class of proteins has a role in broader biological regulating processes such as energy balance and cell cycle control.

6.7 SIMILARITY BETWEEN AD AND PD

We considered the similarity between Parkinson's and Alzheimer's diseases based on the three domains of the GO. So, we can calculate the molecular function (MF) similarity, the biological process similarity (BP) and the cellular component similarity (CC).

We employed Wang's method to compute semantic similarity between the two sets of Alzheimer's and Parkinson's proteins, which takes advantage of the graph structure topology for the GO. We also estimated Lin's similarity between AD and PD, based on the IC of the three GO domains, to examine the differences between these two techniques, as shown in Table 6.1. We can note as the values are similar for both similarity measure, except for a 5% waste for BP.

Measure	**BP similarity**	**MF similarity**	**CC similarity**
Wang	88.3%	91.3%	96.7%
Lin	93%	92%	96.6%

Table 6.1: Similarity values for AD and PD.

In Table 6.2, we listed the shared proteins between the two disorders, along with their UNIPROT IDs and descriptions. We can construct a protein network for each of the three domains under examination based on the similarities of BP, MF and CC. This could be in response to a request from a user for similar proteins to be found in the function, biological process or cellular location of a group of pathologies.

As an example, in Figure 6.8 and Figure 6.9, the BP and MF domains of the P03886 protein, which is seen in AD and PD, are demonstrated to be comparable. The threshold chosen for the representation is 80%. The protein in question is highlighted in the chord graph.

With the same threshold, we recovered the similarities between proteins in PD and AD in Figure 6.10 and Figure 6.11 for BP and MF.

6.8 CONCLUSION

In many domains, graphs are the most natural approach to model interactions between entities. The naturally dynamic nature of such data leads to dynamic graph representations [53]. In this work, we looked at a different technique to visualize the links between GO terms based on their information content graphically. We have presented a human interaction-based viewing system

UNIPROT ID	Description
P03886	NADH-ubiquinone oxidoreductase chain 1
P05067	Amyloid-beta precursor protein
P09936	Ubiquitin carboxyl-terminal hydrolase isozyme L1
P10636	Microtubule-associated protein tau
P25021	Histamine H2 receptor
P37840	Alpha-synuclein
P49754	Vacuolar protein sorting-associated protein 41 homolog
P61026	Ras-related protein Rab-10
P68036	Ubiquitin-conjugating enzyme E2 L3
P78380	Oxidized low-density lipoprotein receptor 1
Q5S007	Leucine-rich repeat serine/threonine-protein kinase 2
Q9H4Y5	Glutathione S-transferase omega-2
Q96IZ0	PRKC apoptosis WT1 regulator protein
Q00535	Cyclin-dependent-like kinase 5
Q13127	RE1-silencing transcription factor
Q13501	Sequestosome-1
Q16143	Beta-synuclein
Q92508	Piezo-type mechanosensitive ion channel component 1
Q92876	Kallikrein-6

Table 6.2: Common proteins in AD and PD.

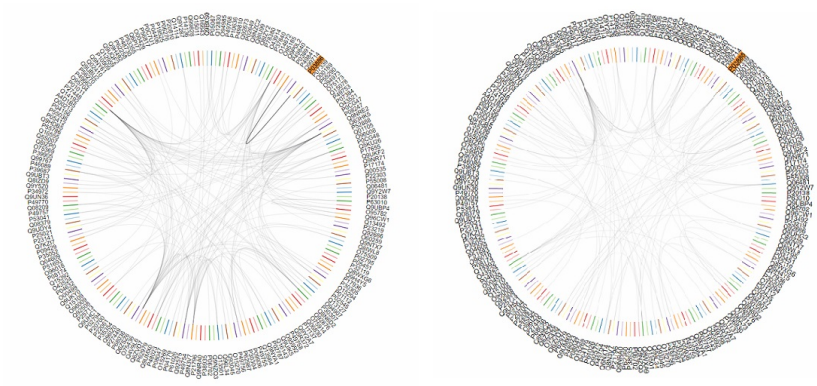


Figure 6.8: Similarity of BP (on left) and MF (on right) for the protein P03886 in AD.

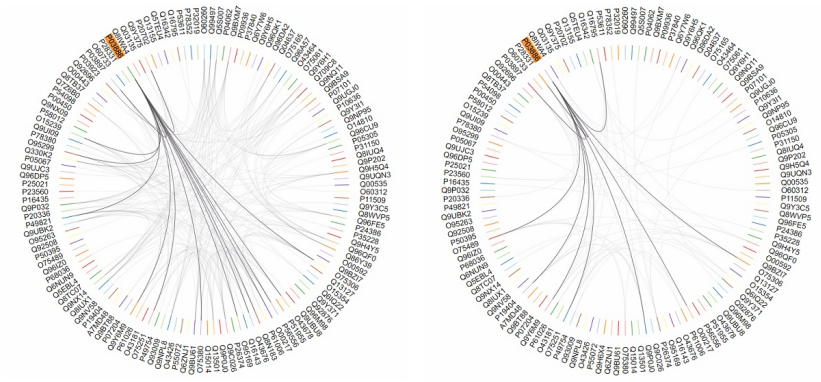


Figure 6.9: Similarity of BP (on left) and MF (on right) for the protein P03886 in PD.

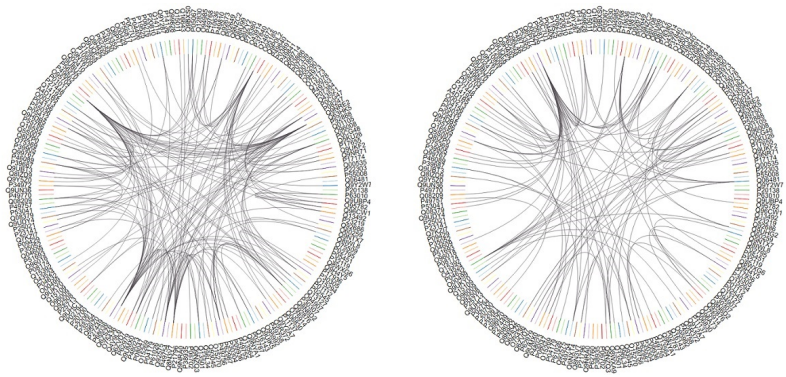


Figure 6.10: Similarity of BP (on left) and MF (on right) in AD.

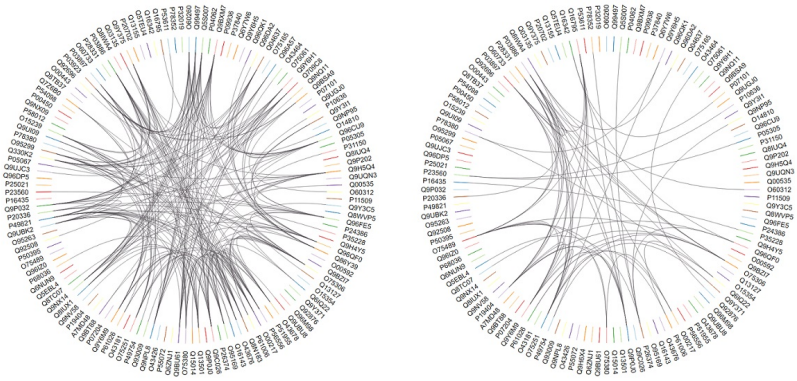


Figure 6.11: Similarity of BP (on left) and MF (on right) in PD.

in particular, which allows users to have a complete omic vision of data. For this purpose, as a GO terms visualization strategy, we presented a distance cyclic distance graph (DCDG) to immediately illustrate interconnection between elements. Using the SigmaJS framework, the prototype was created as a web app. We also explored the differences between the standard cluster view and the proposed DCDG view. Because of the difficulty of overlapping cluster elements, it was not possible to visually recover the information using a traditional clustering display. The display with DCDG, on the other hand, allows for a quicker understanding of the relationships that exist between the proteins based on the similarity representative of the three GO vocabularies (Biological Process, Cellular Component and Molecular Function). One of the goals of our research is to determine whether a system has well-known protein clusters, as this is a crucial topological property for understanding the full network of connections. This subdivision allows us to see the current protein links and provide a tool to detect and explain why particular structural elements are grouped at different degrees of in-depth (cellular, biological, and molecular).

7

CONCLUSIONS AND FUTURE WORKS

In this Chapter, we end this thesis discussing in a summary our contributions of the research work in Section 7.1 and then we provide some insights for future works (in Section 7.2).

7.1 SUMMARY

Over the last few years, there has been a progressive and exponential increase in the amount of biomedical data of various forms and origins. The goal of analyzing all of this data is to better their decoding and our understanding of the biological system. This enables us to determine the implications of their use in the biomedical area in order to determine their relationship to specific disorders. Simultaneously, new IT tools have been developed that allow us to analyze biological Big Data in order to produce novel therapies, diagnoses that are more accurate and extract new knowledge. The ability to combine the calculating power and analysis of IT tools with the use and interaction between the numerous available biobanks is critical to the success of research to discovery of diseases, pathological and functional biological pathways, new drugs and new therapeutic applications.

In this thesis, we address two main topics in the field of biomedical big data analysis: the first concerns melanoma and the second concerns proteins. In particular, we started from the challenges still open in both topics, studying the different approaches and the techniques used currently.

In this contribution, we propose a scalable three-levels architecture (Cloud, Fog and Edge) for a system with the aim of addressing the problem of conservation, training (and re-qualification) and the problem of the distribution of models for the classification of melanoma. Users are able to automatically create and insert new classification models, without the need to change the architecture. Accredited users are able to modify the training, validation, and testing phase databases automatically. We also

tested three deep neural networks: AlexNet, GoogleNet and Inception V3. Our findings also reveal that AlexNet is the most stable network in terms of Transfer Learning. Furthermore, without segmentation or data augmentation, all the CNN networks employed improved their average accuracy. These results encourage further developments, especially, to reduce the number of false positives and increase sensitivity. For the classification of melanoma, we also explored the combination of neural networks and genetic algorithms. These algorithms allow the parallelization of elaboration and the achievement of results close to the excellent in reasonable times (see Chapter 3). Despite the fact that this strategy converges to an acceptable solution, a clearer specification of the initial parameters of the algorithms and related genetic functions (selection, cross over and mutation) is still required.

In the case of protein classification, it is critical to solve the challenge of predicting tertiary structure or assigning chemical-physical identifying features to specific proteins based on their amino acid sequence. As a result, we presented new molecular features to improve the quality of performance of the classification system. Starting with this aspect, we added new features to both the determination of the torsional angles of proteins (Chapter 4) and the classification of proteins in the SNARE family (in Chapter 5). In particular, memory-based deep neural networks as Long short-term memory (LSTM) is used in order to investigate its performance for angles classification. In this context, with the addition of new descriptors and the use of this network on a reduced-size human protein dataset, we have obtained an improvement in the mean absolute error (MAE) on the prediction of the angle ψ. The results show that a gated recurrent unit requires many instances and larger datasets can contribute to further improvements also about the accuracy.

The realization that a biological system is more than the sum of its components, and that its functioning cannot be mirrored by the function of a single component sparked the development of holistic research methods. This has aided the advancement of the omic field, which studies the many classes of biological components in their whole. In the approach to the study of biological systems, in the Chapter 6 we have developed a system

of integrating ontological data of some proteins involved in Parkinson and Alzheimer diseases. In this example, our goal was to improve our understanding and knowledge of biological domains in the face of a significant amount of heterogeneous data by displaying multilayer data in a more immediate and interactive way.

The contributions and major achievements of this thesis can be summarized as follows:

- a system in which the classification of the melanoma data uses combined AI techniques typical of Computer Vision for managing, integration, pre-processing and classification of melanoma data. This system is based on three architectural levels, in which the cloud level manages the data centrally, the fog level performs the services offered by the network and the edge level performs computations at the local level, improving resource management, interoperability, and computing power;
- preliminary results obtained from the combination of genetic algorithms and the neural network AlexNet for the classification of melanoma;
- presentation of new molecular descriptors for the prediction of the torsional angles of proteins (ϕ and ψ) and for the classification of SNARE proteins;
- a new web server which allows us to introduce a more immediate and interactive display mode of the ontology-related protein data by similarity between the terms of the three ontological domains of the *Gene Ontology* (biological process, cellular component and molecular function).

7.2 FUTURE WORKS

In a variety of fields, deep learning algorithms have achieved exceptional classification performance. Parallel to the exponential increase of Big Data, the development of mining algorithms, as well as the creation of more performant platforms and architecture for these analyses, must be accelerated. One potential direction for further research for melanoma classification is to

increase the accuracy of classification systems based on neural networks. In our work we used images of melanoma readily available, thanks to the use of smart devices of common use, bypassing the problem of having more sophisticated tools. However, even in this dermatological field, it is still necessary to have a broad perspective that considers all relevant data in order to provide an accurate diagnosis. Anamnestic data, related to the patients and their family history, could be used to improve melanoma classification performance. Additional clinical features can also be extrapolated from images at the same time. As a future work in this direction, we intend to further investigate the possible correlations between different data, expanding analysis to other data sources at clinical level. In this context, the heterogeneity of multi-domain text sources can be taken into consideration and exploited with new methods of transfer learning. Protein classification based on related activities, structural patterns, extended gene annotations and any other multi-class task are all possible future research topics. Other machine learning methods can be tested to achieve this goal. Specifically, it is useful to find new molecular descriptors that allow us to distinguish the great protein amount produced by sequencing techniques much more effectively. The representation and probable linkages between numerous biological domains can be improved in order to improve the omic vision of biological systems. Methods for various algorithms of clustering, for example, may be encountered, as well as the building of biological networks from the starting data.

BIBLIOGRAPHY

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. "Tensorflow: A system for large-scale machine learning." In: *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*. 2016, pp. 265–283.

[2] Naheed R Abbasi, Helen M Shaw, Darrell S Rigel, Robert J Friedman, William H McCarthy, Iman Osman, Alfred W Kopf, and David Polsky. "Early diagnosis of cutaneous melanoma: revisiting the ABCD criteria." In: *JAMA Dermatology* 292.22 (2004), pp. 2771–2776.

[3] Adekanmi Adegun and Serestina Viriri. "Deep learning techniques for skin lesion analysis and melanoma cancer detection: a survey of state-of-the-art." In: *Artificial Intelligence Review* 54.2 (2021), pp. 811–841.

[4] Hamdan O Alanazi, Abdul Hanan Abdullah, and Kashif Naseer Qureshi. "A critical review for developing accurate and dynamic predictive models using machine learning methods in medicine and health care." In: *Journal of medical systems* 41.4 (2017), p. 69.

[5] Marwan Ali Albahar. "Skin lesion classification using convolutional neural network with novel regularizer." In: *IEEE Access* 7 (2019), pp. 38306–38313.

[6] Vitoria Diana Mateus de Almeida Gonçalves, Marcelo Ferrari de Almeida Camargo Filho, Tânia Zaleski, Rogério Rodrigues Vilas Boas, Elaine Rossi Ribeiro, Rogério Saad Vaz, and Francelise Bridi Cavassin. "Chemotherapy in focus: a meta-analysis confronts immunotherapy in the treatment of advanced melanoma." In: *Critical Reviews in Oncology/Hematology* (2021), p. 103304.

[7] Md Zahangir Alom, Tarek M Taha, Chris Yakopcic, Stefan Westberg, Paheding Sidike, Mst Shamima Nasrin, Mahmudul Hasan, Brian C Van Essen, Abdul AS Awwal, and Vijayan K Asari. "A state-of-the-art survey on deep learning theory and architectures." In: *Electronics* 8.3 (2019), p. 292.

[8] Stephen F Altshull, Thomas L Madden, Alejandro A Schäffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J Lipman. "Gapped blast and psiblast: a new generation of protein database searche programs." In: *Nucleic Acids Res* 25.17 (1997), p. 3.

[9] Muhammad Arif. "Similarity-dissimilarity plot for visualization of high dimensional data in biomedical pattern classification." In: *Journal of Medical Systems* 36.3 (2012), pp. 1173–1181.

[10] Mohamed Attia, Mohamed Hossny, Saeid Nahavandi, and Anousha Yazdabadi. "Skin melanoma segmentation using recurrent and convolutional neural networks." In: *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*. IEEE. 2017, pp. 292–296.

[11] Binamrata Baral, Sandeep Gonnade, and Toran Verma. "Lesion segmentation in dermoscopic images using decision based neuro fuzzy model." In: *International Journal of Computer Science and Information Technologies* 5.2 (2014), pp. 2546–2552.

[12] Siddharth S Bass, Gary H Lyman, Christa R McCann, Ni Ni Ku, Claudia Berman, Kara Durand, Monica Bolano, Sarah Cox, Christopher Salud, Douglas S Reintgen, et al. "Lymphatic mapping and sentinel lymph node biopsy." In: *The breast journal* 5.5 (1999), pp. 288–295.

[13] Mitra Basu. "Gaussian-based edge-detection methods-a survey." In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 32.3 (2002), pp. 252–260.

[14] Andreas D Baxevanis, Gary D Bader, and David S Wishart. *Bioinformatics*. John Wiley & Sons, 2020.

[15] Prachi Bhave, Lalit Pallan, Georgina V Long, Alexander M Menzies, Victoria Atkinson, Justine V Cohen, Ryan J Sullivan, Vanna Chiarion-Sileni, Marta Nyakas, Katharina Kahler, et al. "Melanoma recurrence patterns and management after adjuvant targeted therapy: a multicentre analysis." In: *British journal of cancer* 124.3 (2021), pp. 574–580.

[16] David Binns, Emily Dimmer, Rachael Huntley, Daniel Barrell, Claire O'donovan, and Rolf Apweiler. "QuickGO: A web-based tool for Gene Ontology searching." In: *Bioinformatics* 25.22 (2009), pp. 3045–3046.

[17] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*. Vol. 4. 4. Springer, 2006.

[18] Sabri Boughorbel, Fethi Jarray, and Mohammed El-Anbari. "Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric." In: *PloS one* 12.6 (2017), e0177678.

[19] Carl Ivar Branden and John Tooze. *Introduction to protein structure*. Garland Science, 2012.

[20] Ralph Peter Braun, Harold S Rabinovitz, Margaret Oliviero, Alfred W Kopf, and Jean-Hilaire Saurat. "Dermoscopy of pigmented skin lesions." In: *Journal of the American Academy of Dermatology* 52.1 (2005), pp. 109–121.

[21] CZ Cai, LY Han, Zhi Liang Ji, X Chen, and Yu Zong Chen. "SVM-Prot: web-based support vector machine software for functional classification of a protein from its primary sequence." In: *Nucleic acids research* 31.13 (2003), pp. 3692–3697.

[22] Hung Cao, Monica Wachowicz, Chiara Renso, and Emanuele Carlini. "An edge-fog-cloud platform for anticipatory learning process designed for internet of mobile things." In: *arXiv preprint arXiv:1711.09745* (2017).

[23] Oliviero Carugo and Kristina Djinović-Carugo. "A proteomic Ramachandran plot (PRplot)." In: *Amino acids* 44.2 (2013), pp. 781–790.

[24] Andrea Cavalli, Xavier Salvatella, Christopher M Dobson, and Michele Vendruscolo. "Protein structure determination from NMR chemical shifts." In: *Proceedings of the National Academy of Sciences* 104.23 (2007), pp. 9615–9620.

[25] M Emre Celebi, Hitoshi Iyatomi, Gerald Schaefer, and William V Stoecker. "Lesion border detection in dermoscopy images." In: *Computerized medical imaging and graphics* 33.2 (2009), pp. 148–153.

[26] Saptarshi Chatterjee, Debangshu Dey, Sugata Munshi, and Surajit Gorai. "Dermatological expert system implementing the ABCD rule of dermoscopy for skin disease identification." In: *Expert Systems with Applications* 167 (2021), p. 114204.

[27] Hsinchun Chen, Roger HL Chiang, and Veda C Storey. "Business intelligence and analytics: From big data to big impact." In: *MIS quarterly* (2012), pp. 1165–1188.

[28] Ke Chen, Lukasz A Kurgan, and Jishou Ruan. "Prediction of flexible/rigid regions from protein sequences using k-spaced amino acid pairs." In: *BMC structural biology* 7.1 (2007), p. 25.

[29] Yu A Chen and Richard H Scheller. "SNARE-mediated membrane fusion." In: *Nature reviews Molecular cell biology* 2.2 (2001), pp. 98–106.

[30] Zhen Chen, Pei Zhao, Fuyi Li, André Leier, Tatiana T Marquez-Lago, Yanan Wang, Geoffrey I Webb, A Ian Smith, Roger J Daly, Kuo-Chen Chou, et al. "iFeature: a python package and web server for features extraction and selection from protein and peptide sequences." In: *Bioinformatics* 34.14 (2018), pp. 2499–2502.

[31] Davide Chicco and Giuseppe Jurman. "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation." In: *BMC genomics* 21.1 (2020), pp. 1–13.

[32] Cyrus Chothia. "Hydrophobic bonding and accessible surface area in proteins." In: *Nature* 248.5446 (1974), pp. 338–339.

[33] Kuo-Chen Chou. "Prediction of protein cellular attributes using pseudo-amino acid composition." In: *Proteins: Structure, Function, and Bioinformatics* 43.3 (2001), pp. 246–255.

[34] Kuo-Chen Chou. "Pseudo amino acid composition and its applications in bioinformatics, proteomics and system biology." In: *Current Proteomics* 6.4 (2009), pp. 262–274.

[35] Noel CF Codella, Q-B Nguyen, Sharath Pankanti, David A Gutman, Brian Helba, Allan C Halpern, and John R Smith. "Deep learning ensembles for melanoma recognition in dermoscopy images." In: *IBM Journal of Research and Development* 61.4/5 (2017), pp. 5–1.

[36] Noel Codella, Junjie Cai, Mani Abedini, Rahil Garnavi, Alan Halpern, and John R Smith. "Deep learning, sparse coding, and SVM for melanoma recognition in dermoscopy images." In: *International workshop on machine learning in medical imaging*. Springer. 2015, pp. 118–126.

[37] Ana Conesa, Stefan Götz, Juan Miguel García-Gómez, Javier Terol, Manuel Talón, and Montserrat Robles. "Blast2GO: A universal tool for annotation, visualization and analysis in functional genomics research." In: *Bioinformatics* 21.18 (2005), pp. 3674–3676.

[38] Gene Ontology Consortium. "Gene ontology consortium: going forward." In: *Nucleic acids research* 43.D1 (2015), pp. D1049–D1056.

[39] Thomas E Creighton. "Protein folding." In: *Biochemical journal* 270.1 (1990), p. 1.

[40] Yuri Demchenko, Paola Grosso, Cees De Laat, and Peter Membrey. "Addressing big data issues in scientific data infrastructure." In: *2013 International conference on collaboration technologies and systems (CTS)*. IEEE. 2013, pp. 48–55.

[41] Luigi Dibiasi, Michele Risi, Genoveffa Tortora, and Alessia Auriemma Citarella. "A Cloud Approach for Melanoma Detection based on Deep Learning Networks." In: *IEEE Journal of Biomedical and Health Informatics* (2021).

[42] YR Ding, YJ Cai, PD Sun, and B Chen. "The use of combined neural networks and genetic algorithms for prediction of river water quality." In: *Journal of Applied Research and Technology* 12.3 (2014), pp. 493–499.

[43] Jacqueline Dinnes, Lavinia Ferrante di Ruffano, Yemisi Takwoingi, Seau Tak Cheung, Paul Nathan, Rubeta N Matin, Naomi Chuchu, Sue Ann Chan, Alana Durack, Susan E Bayliss, et al. "Ultrasound, CT, MRI, or PET-CT for staging and re-staging of adults with cutaneous melanoma." In: *Cochrane Database of Systematic Reviews* 7 (2019).

[44] Ivo D Dinov. "Volume and value of big healthcare data." In: *Journal of medical statistics and informatics* 4 (2016).

[45] Lyn M Duncan, James Deeds, Frank E Cronin, Michael Donovan, Arthur J Sober, Michael Kauffman, and Jeanette J McCarthy. "Melastatin expression and prognosis in cutaneous malignant melanoma." In: *Journal of clinical oncology* 19.2 (2001), pp. 568–576.

[46] Charles Duyckaerts, Benoit Delatour, and Marie-Claude Potier. "Classification and basic pathology of Alzheimer disease." In: *Acta Neuropathologica* 118.1 (2009), pp. 5–36.

[47] Sean Eddy. "HMMER user's guide. biological sequence analysis using profile hidden Markov models." In: (2003).

[48] Eran Eden, Roy Navon, Israel Steinfeld, Doron Lipson, and Zohar Yakhini. "GOrilla: A tool for discovery and visualization of enriched GO terms in ranked gene lists." In: *BMC Bioinformatics* 10.1 (2009), pp. 1–7.

[49] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. "Dermatologist-level classification of skin cancer with deep neural networks." In: *Nature* 542.7639 (2017), pp. 115–118.

[50] B. Etain, A. Dumaine, F. Mathieu, F. Chevalier, C. Henry, J. Kahn, J. Deshommes, F. Bellivier, M. Leboyer, and S. Jamain. "A SNAP25 promoter variant is associated with early-onset bipolar disorder and a high expression level in brain." In: *Molecular Psychiatry* 15.7 (2010), pp. 748–755.

[51] JEAN-LUC FAUCHÈRE, Marvin Charton, Lemont B Kier, Arie Verloop, and Vladimir Pliska. "Amino acid side chain parameters for correlation studies in biology and pharmacology." In: *International journal of peptide and protein research* 32.4 (1988), pp. 269–278.

[52] Dirk Fasshauer, R Bryan Sutton, Axel T Brunger, and Reinhard Jahn. "Conserved structural features of the synaptic fusion complex: SNARE proteins reclassified as Q-and R-SNAREs." In: *Proceedings of the national academy of sciences* 95.26 (1998), pp. 15781–15786.

[53] Daniel J Fenn, Mason A Porter, Peter J Mucha, Mark McDonald, Stacy Williams, Neil F Johnson, and Nick S Jones. "Dynamical clustering of exchange rates." In: *Quantitative Finance* 12.10 (2012), pp. 1493–1520.

[54] Hannes Flöckner, Michael Braxenthaler, Peter Lackner, Markus Jaritz, Maria Ortner, and Manfred J Sippl. "Progress in fold recognition." In: *Proteins: Structure, Function, and Bioinformatics* 23.3 (1995), pp. 376–386.

[55] Charlotte Forsell, Behnosh Fakhri Björk, Lena Lilius, Karin Axelman, Susanne Froelich Fabre, Laura Fratiglioni, Bengt Winblad, and Caroline Graff. "Genetic association to the amyloid plaque associated protein gene COL25A1 in Alzheimer's disease." In: *Neurobiology of aging* 31.3 (2010), pp. 409–415.

[56] Maria Frasca, Michele Nappi, Michele Risi, Genoveffa Tortora, and Alessia Auriemma Citarella. "A comparison of neural network approaches for melanoma classification." In: *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE. 2021, pp. 2110–2117.

[57] Yoav Freund and Robert E Schapire. "A decision-theoretic generalization of on-line learning and an application to boosting." In: *Journal of computer and system sciences* 55.1 (1997), pp. 119–139.

[58] Y Fujisawa, Y Otomo, Y Ogata, Y Nakamura, R Fujita, Y Ishitsuka, R Watanabe, N Okiyama, K Ohara, and M Fujimoto. "Deep-learning-based, computer-aided classifier developed with a small dataset of clinical images

surpasses board-certified dermatologists in skin tumour diagnosis." In: *British Journal of Dermatology* 180.2 (2019), pp. 373–381.

[59] Robert D Galliers, Sue Newell, G Shanks, and Heikki Topi. "Datification and its human, organizational and societal effects." In: *The Journal of Strategic Information Systems* 26.3 (2017), pp. 185–190.

[60] Jianzhao Gao, Yuedong Yang, and Yaoqi Zhou. "Grid-based prediction of torsion angle probabilities of protein backbone and its application to discrimination of protein intrinsic disorder regions and selection of model structures." In: *BMC bioinformatics* 19.1 (2018), pp. 1–8.

[61] Yujuan Gao, Sheng Wang, Minghua Deng, and Jinbo Xu. "RaptorX-Angle: real-value prediction of protein backbone dihedral angles through a hybrid method of clustering and deep learning." In: *BMC bioinformatics* 19.4 (2018), pp. 73–84.

[62] Pablo Garcia-Reitböck, Oleg Anichtchik, Arianna Bellucci, Mariangela Iovino, Chiara Ballini, Elena Fineberg, Bernardino Ghetti, Laura Della Corte, PierFranco Spano, George K Tofaris, et al. "SNARE protein redistribution and synaptic failure in a transgenic mouse model of Parkinson's disease." In: *Brain* 133.7 (2010), pp. 2032–2044.

[63] MR Garey, David Johnson, and Hans Witsenhausen. "The complexity of the generalized Lloyd-max problem (corresp.)" In: *IEEE Transactions on Information Theory* 28.2 (1982), pp. 255–256.

[64] Peter Géczy. "Big data characteristics." In: *The Macrotheme Review* 3.6 (2014), pp. 94–104.

[65] Gene Ontology Consortium. "The gene ontology project." In: *Nucleic Acids Research* 36.suppl_1 (2008), pp. D440–D444.

[66] Jeffrey E Gershenwald, Richard A Scolyer, Kenneth R Hess, Vernon K Sondak, Georgina V Long, Merrick I Ross, Alexander J Lazar, Mark B Faries, John M Kirkwood, Grant A McArthur, et al. "Melanoma staging: evidence-based changes in the American Joint Committee on Can-

cer eighth edition cancer staging manual." In: *CA: a cancer journal for clinicians* 67.6 (2017), pp. 472–492.

[67] Stephen Gilmore, Rainer Hofmann-Wellenhof, and H Peter Soyer. "A support vector machine for decision support in melanoma recognition." In: *Experimental Dermatology* 19.9 (2010), pp. 830–835.

[68] Ioannis Giotis, Nynke Molders, Sander Land, Michael Biehl, Marcel F Jonkman, and Nicolai Petkov. "MED-NODE: A computer-assisted melanoma diagnosis system using non-dermoscopic images." In: *Expert Systems with Applications* 42.19 (2015), pp. 6578–6585.

[69] Michael Golden, Eduardo García-Portugués, Michael Sørensen, Kanti V Mardia, Thomas Hamelryck, and Jotun Hein. "A generative angular model of protein structure evolution." In: *Molecular biology and evolution* 34.8 (2017), pp. 2085–2100.

[70] Manu Goyal, Thomas Knackstedt, Shaofeng Yan, and Saeed Hassanpour. "Artificial intelligence-based image classification for diagnosis of skin cancer: Challenges and opportunities." In: *Computers in Biology and Medicine* (2020), p. 104065.

[71] Manu Goyal, Thomas Knackstedt, Shaofeng Yan, and Saeed Hassanpour. "Artificial intelligence-based image classification methods for diagnosis of skin cancer: Challenges and opportunities." In: *Computers in Biology and Medicine* 127 (2020), p. 104065. ISSN: 0010-4825. DOI: https://doi.org/10.1016/j.compbiomed.2020.104065. URL: https://www.sciencedirect.com/science/article/pii/S0010482520303966.

[72] F. R. Guerini, E. Bolognesi, M. Chiappedi, S. Manca, A. Ghezzo, C. Agliardi, S. Sotgiu, S. Usai, M. Matteoli, and M. Clerici. "SNAP-25 single nucleotide polymorphisms are associated with hyperactivity in autism spectrum disorders." In: *Pharmacological Research* 64.3 (2011), pp. 283–288.

[73] Manoj Kumar Gupta, Gayatri Gouda, S Sabarinathan, Ravindra Donde, Pallabi Pati, Sushil Kumar Rathore, Ramakrishna Vadde, and Lambodar Behera. "Structural Proteomics." In: *Bioinformatics in Rice Research*. Springer, 2021, pp. 239–256.

[74] Holger Andreas Haenssle, Christine Fink, Ferdinand Toberer, Julia Winkler, Wilhelm Stolz, Teresa Deinlein, Rainer Hofmann-Wellenhof, Aimilios Lallas, Steffen Emmert, Timo Buhl, et al. "Man against machine reloaded: performance of a market-approved convolutional neural network in classifying a broad spectrum of skin lesions in comparison with 96 dermatologists working under less artificial conditions." In: *Annals of Oncology* 31.1 (2020), pp. 137–143.

[75] Thomas Hamelryck. "An amino acid has two sides: a new 2D measure provides a different view of solvent exposure." In: *Proteins: Structure, Function, and Bioinformatics* 59.1 (2005), pp. 38–48.

[76] James M Heather and Benjamin Chain. "The sequence of sequencers: The history of sequencing DNA." In: *Genomics* 107.1 (2016), pp. 1–8.

[77] Rhys Heffernan, Kuldip Paliwal, James Lyons, Abdollah Dehzangi, Alok Sharma, Jihua Wang, Abdul Sattar, Yuedong Yang, and Yaoqi Zhou. "Improving prediction of secondary structure, local backbone angles and solvent accessible surface area of proteins by iterative deep learning." In: *Scientific reports* 5.1 (2015), pp. 1–11.

[78] Rhys Heffernan, Yuedong Yang, Kuldip Paliwal, and Yaoqi Zhou. "Capturing non-local interactions by long short-term memory bidirectional recurrent neural networks for improving prediction of protein secondary structure, backbone angles, contact numbers and solvent accessibility." In: *Bioinformatics* 33.18 (2017), pp. 2842–2849.

[79] Tin Kam Ho. "Random decision forests." In: *Proceedings of 3rd international conference on document analysis and recognition*. Vol. 1. IEEE. 1995, pp. 278–282.

[80] Sepp Hochreiter. "The vanishing gradient problem during learning recurrent neural nets and problem solutions." In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 6.02 (1998), pp. 107–116.

[81] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory." In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[82] William G Honer, Peter Falkai, Thomas A Bayer, Jane Xie, Lily Hu, Hong-Ying Li, Victoria Arango, J John Mann, Andrew J Dwork, and William S Trimble. "Abnormalities of SNARE mechanism proteins in anterior frontal cortex in severe mental illness." In: *Cerebral Cortex* 12.4 (2002), pp. 349–356.

[83] John J Hopfield. "Artificial neural networks." In: *IEEE Circuits and Devices Magazine* 4.5 (1988), pp. 3–10.

[84] Khalid M Hosny, Mohamed A Kassem, and Mohamed M Foaud. "Classification of skin lesions using transfer learning and augmentation with Alex-net." In: *PloS one* 14.5 (2019), e0217293.

[85] Zilong Hu, Jinshan Tang, Ziming Wang, Kai Zhang, Ling Zhang, and Qingling Sun. "Deep learning for image-based cancer detection and diagnosis- A survey." In: *Pattern Recognition* 83 (2018), pp. 134–149.

[86] Atsushi Ikai. "Thermostability and aliphatic index of globular proteins." In: *The Journal of Biochemistry* 88.6 (1980), pp. 1895–1898.

[87] Andrea Ilari and Carmelinda Savino. "Protein structure determination by x-ray crystallography." In: *Bioinformatics* (2008), pp. 63–87.

[88] Mathieu Jacomy, Tommaso Venturini, Sebastien Heymann, and Mathieu Bastian. "ForceAtlas2, a continuous graph layout algorithm for handy network visualization designed for the Gephi software." In: *PloS one* 9.6 (2014), e98679.

[89] Shunichi Jinnai, Naoya Yamazaki, Yuichiro Hirano, Yohei Sugawara, Yuichiro Ohe, and Ryuji Hamamoto. "The development of a skin cancer classification system for pigmented skin lesions using deep learning." In: *Biomolecules* 10.8 (2020), p. 1123.

[90] David T Jones. "Protein secondary structure prediction based on position-specific scoring matrices." In: *Journal of molecular biology* 292.2 (1999), pp. 195–202.

[91] Wolfgang Kabsch and Christian Sander. "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features." In: *Biopolymers: Original Research on Biomolecules* 22.12 (1983), pp. 2577–2637.

[92] Avita Katal, Mohammad Wazid, and Rayan H Goudar. "Big data: issues, challenges, tools and good practices." In: *2013 Sixth international conference on contemporary computing (IC3)*. IEEE. 2013, pp. 404–409.

[93] Sourabh Katoch, Sumit Singh Chauhan, and Vijay Kumar. "A review on genetic algorithm: past, present, and future." In: *Multimedia Tools and Applications* 80.5 (2021), pp. 8091–8126.

[94] Jeremy Kawahara, Sara Daneshvar, Giuseppe Argenziano, and Ghassan Hamarneh. "Seven-point checklist and skin lesion classification using multitask multimodal neural nets." In: *IEEE journal of biomedical and health informatics* 23.2 (2018), pp. 538–546.

[95] Shuichi Kawashima and Minoru Kanehisa. "AAindex: amino acid index database." In: *Nucleic acids research* 28.1 (2000), pp. 374–374.

[96] James M Keller, Michael R Gray, and James A Givens. "A fuzzy k-nearest neighbor algorithm." In: *IEEE transactions on systems, man, and cybernetics* 4 (1985), pp. 580–585.

[97] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization." In: *arXiv preprint arXiv:1412.6980* (2014).

[98] Gerard J Kleywegt and T Alwyn Jones. "Phi/psi-chology: Ramachandran revisited." In: *Structure* 4.12 (1996), pp. 1395–1400.

[99] Tobias H Kloepper, C Nickias Kienle, and Dirk Fasshauer. "An elaborate classification of SNARE proteins sheds light on the conservation of the eukaryotic endomembrane system." In: *Molecular biology of the cell* 18.9 (2007), pp. 3463–3471.

[100] Janusz Kolbusz, Pawel Rozycki, and Bogdan M Wilamowski. "The study of architecture MLP with linear neurons in order to eliminate the "vanishing gradient" problem." In: *International Conference on Artificial Intelligence and Soft Computing*. Springer. 2017, pp. 97–106.

[101] Koffi Mawuna Koudjonou and Minakhi Rout. "A stateless deep learning framework to predict net asset value." In: *Neural Computing and Applications* (2019), pp. 1–19.

[102] Oliver Kramer. "Genetic algorithms." In: *Genetic Algorithm Essentials*. Springer, 2017, pp. 11–19.

[103] Elmar Krieger, Sander B Nabuurs, and Gert Vriend. "Homology modeling." In: *Methods of biochemical analysis* 44 (2003), pp. 509–524.

[104] Elmar Krieger and Gert Vriend. "YASARA View—molecular graphics for all devices—from smartphones to workstations." In: *Bioinformatics* 30.20 (2014), pp. 2981–2982.

[105] Puneet Kumar, Shalini Batra, and Balasubramanian Raman. "Deep neural network hyper-parameter tuning through twofold genetic approach." In: *Soft Computing* 25.13 (2021), pp. 8747–8771.

[106] Kun Lan, Dan-tong Wang, Simon Fong, Lian-sheng Liu, Kelvin KL Wong, and Nilanjan Dey. "A survey of data mining and deep learning in bioinformatics." In: *Journal of medical systems* 42.8 (2018), pp. 1–20.

[107] Nguyen Quoc Khanh Le and Van-Nui Nguyen. "SNARE-CNN: a 2D convolutional neural network architecture to identify SNARE proteins from high-throughput sequencing data." In: *PeerJ Computer Science* 5 (2019), e177.

[108] Yann LeCun, Yoshua Bengio, et al. "Convolutional networks for images, speech, and time series." In: *The handbook of brain theory and neural networks* 3361.10 (1995), p. 1995.

[109] Tim Lee, Vincent Ng, Richard Gallagher, Andrew Coldman, and David McLean. "Dullrazor®: A software approach to hair removal from images." In: *Computers in Biology and Medicine* 27.6 (1997), pp. 533–543.

[110] Ulrike Leiter and Claus Garbe. "Epidemiology of melanoma and nonmelanoma skin cancer—the role of sunlight." In: *Sunlight, vitamin D and skin cancer* (2008), pp. 89–103.

[111] Guilin Li. "Identification of SNARE proteins through a novel hybrid model." In: *IEEE Access* 8 (2020), pp. 117877–117887.

[112] Haiou Li, Jie Hou, Badri Adhikari, Qiang Lyu, and Jianlin Cheng. "Deep learning methods for protein torsion angle prediction." In: *BMC bioinformatics* 18.1 (2017), pp. 1–13.

[113] Weizhong Li and Adam Godzik. "Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences." In: *Bioinformatics* 22.13 (2006), pp. 1658–1659.

[114] Xiangbo Li, Mohsen Amini Salehi, Yamini Joshi, Mahmoud K Darwich, Brad Landreneau, and Magdy Bayoumi. "Performance analysis and modeling of video transcoding using heterogeneous cloud services." In: *IEEE Transactions on Parallel and Distributed Systems* 30.4 (2018), pp. 910–922.

[115] Yixue Li and Luonan Chen. "Big biological data: challenges and opportunities." In: *Genomics, proteomics & bioinformatics* 12.5 (2014), p. 187.

[116] Yuexiang Li and Linlin Shen. "Skin lesion analysis towards melanoma detection using deep learning network." In: *Sensors* 18.2 (2018), p. 556.

[117] Dekang Lin. "Extracting collocations from text corpora." In: *Proceedings of the First Workshop on Computational Terminology*. 1998, pp. 57–63.

[118] Rui-yan Luo, Zhi-ping Feng, and Jia-kun Liu. "Prediction of protein structural class by amino acid and polypeptide composition." In: *European Journal of Biochemistry* 269.17 (2002), pp. 4219–4225.

[119] James MacQueen et al. "Some methods for classification and analysis of multivariate observations." In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. Vol. 1. 14. 1967, pp. 281–297.

[120] Stephan Mandt, Matthew D Hoffman, and David M Blei. "Stochastic gradient descent as approximate bayesian inference." In: *arXiv preprint arXiv:1704.04289* (2017).

[121] Jens Meiler, Michael Müller, Anita Zeidler, and Felix Schmäschke. "Generation and evaluation of dimension-reduced amino acid parameter representations by artificial neural networks." In: *Molecular modeling annual* 7.9 (2001), pp. 360–369.

[122] Rudy Melli, Costantino Grana, and Rita Cucchiara. "Comparison of color clustering algorithms for segmentation of dermatological images." In: *Medical Imaging 2006: Image Processing*. Vol. 6144. International Society for Optics and Photonics. 2006, 61443S.

[123] Jianghui Meng and Jiafu Wang. "Role of SNARE proteins in tumourigenesis and their potential as targets for novel anti-cancer therapeutics." In: *Biochimica et Biophysica Acta (BBA)-Reviews on Cancer* 1856.1 (2015), pp. 1–12.

[124] Scott W Menzies. "d Menzies method." In: *An Atlas of Dermoscopy*. CRC Press, 2012, pp. 128–133.

[125] Kajsa Møllersen, Herbert Kirchesch, Maciel Zortea, Thomas R Schopf, Kristian Hindberg, and Fred Godtliebsen. "Computer-aided decision support for melanoma detection applied on melanocytic and nonmelanocytic skin lesions: a comparison of two systems based on automatic analysis of dermoscopic images." In: *BioMed research international* 2015 (2015).

[126] William Montagna. *The structure and function of skin*. Elsevier, 2012.

[127] Anthony J Myles, Robert N Feudale, Yang Liu, Nathaniel A Woody, and Steven D Brown. "An introduction to decision tree modeling." In: *Journal of Chemometrics: A Journal of the Chemometrics Society* 18.6 (2004), pp. 275–285.

[128] Piyu Parth Naik. "Cutaneous Malignant Melanoma: A Review of Early Diagnosis and Management." In: *World Journal of Oncology* 12.1 (2021), p. 7.

[129] Kazuhiko Nakamura, Ayyappan Anitha, Kazuo Yamada, Masatsugu Tsujii, Yoshimi Iwayama, Eiji Hattori, Tomoko Toyota, Shiro Suda, Noriyoshi Takei, Yasuhide Iwata, et al. "Genetic and expression analyses reveal elevated expression of syntaxin 1A (STX1A) in high functioning autism." In: *International Journal of Neuropsychopharmacology* 11.8 (2008), pp. 1073–1084.

[130] Ebrahim Nasr-Esfahani, Shadrokh Samavi, Nader Karimi, S Mohamad R Soroushmehr, Mohammad H Jafari, Kevin Ward, and Kayvan Najarian. "Melanoma detection by analysis of clinical images using convolutional neural network." In: *Proceedings of the 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE. 2016, pp. 1373–1376.

[131] S Needleman and C Wunsch. "A General Method Applicable to the Search for Similarities in the Amino Acid Sequences of Two Proteins." In: *Molecular Biology: A Selection of Papers* 48 (2012), p. 453.

[132] David L Nelson, Albert L Lehninger, and Michael M Cox. *Lehninger principles of biochemistry*. Macmillan, 2008.

[133] William S Noble. "What is a support vector machine?" In: *Nature biotechnology* 24.12 (2006), pp. 1565–1567.

[134] Seán I O'Donoghue, Anne-Claude Gavin, Nils Gehlenborg, David S Goodsell, Jean-Karim Hériché, Cydney B Nielsen, Chris North, Arthur J Olson, James B Procter, David W Shattuck, et al. "Visualizing biological data—now and in the future." In: *Nature Methods* 7.3 (2010), S2–S4.

[135] Erdem Okur and Mehmet Turkan. "A survey on automated melanoma detection." In: *Engineering Applications of Artificial Intelligence* 73 (2018), pp. 50–67.

[136] Serene AK Ong, Hong Huang Lin, Yu Zong Chen, Ze Rong Li, and Zhiwei Cao. "Efficacy of different protein descriptors in predicting protein functional families." In: *Bmc Bioinformatics* 8.1 (2007), p. 300.

[137] Christine A Orengo, David T Jones, and Janet M Thornton. "Protein superfamilles and domain superfolds." In: *Nature* 372.6507 (1994), pp. 631–634.

[138] Christine A Orengo, Annabel E Todd, and Janet M Thornton. "From protein structure to function." In: *Current opinion in structural biology* 9.3 (1999), pp. 374–382.

[139] Nobuyuki Otsu. "A threshold selection method from gray-level histograms." In: *IEEE Transactions on Systems, Man, and Cybernetics* 9.1 (1979), pp. 62–66.

[140] Debnath Pal and David Eisenberg. "Inference of protein function from protein structure." In: *Structure* 13.1 (2005), pp. 121–130.

[141] Sinno Jialin Pan and Qiang Yang. "A survey on transfer learning." In: *IEEE Transactions on Knowledge and Data Engineering* 22.10 (2009), pp. 1345–1359.

[142] Komal Patil and Usha Chouhan. "Relevance of Machine Learning Techniques and Various Protein Features in Protein Fold Classification: A Review." In: *Current Bioinformatics* 14.8 (2019), pp. 688–697.

[143] Stefano Piotto, Luigi Di Biasi, Simona Concilio, Aniello Castiglione, and Giuseppe Cattaneo. "GRIMD: distributed computing for chemists and biologists." In: *Bioinformation* 10.1 (2014), p. 43.

[144] Werner Poewe, Klaus Seppi, Caroline M Tanner, Glenda M Halliday, Patrik Brundin, Jens Volkmann, Anette-Eleonore Schrag, and Anthony E Lang. "Parkinson disease." In: *Nature Reviews Disease Primers* 3.1 (2017), pp. 1–21.

[145] Dawid Połap, Gautam Srivastava, and Marcin Woźniak. "Multi-agent Architecture for Internet of Medical Things." In: *Proceedings of the International Conference on Artificial Intelligence and Soft Computing*. Springer. 2020, pp. 49–58.

[146] Li-na Qi, Bo Zhang, and Zhan-kai Wang. "Application of the Otsu method in image processing." In: *Radio Engineering of China* 7.009 (2006).

[147] Neeliyath A Ramakrishnan, Marian J Drescher, and Dennis G Drescher. "The SNARE complex in neuronal and sensory cells." In: *Molecular and Cellular Neuroscience* 50.1 (2012), pp. 58–69.

[148] Marco Rastrelli, Saveria Tropea, Carlo Riccardo Rossi, and Mauro Alaibac. "Melanoma: epidemiology, risk factors, pathogenesis, diagnosis and classification." In: *In vivo* 28.6 (2014), pp. 1005–1011.

[149] Michael Remmert, Andreas Biegert, Andreas Hauser, and Johannes Söding. "HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment." In: *Nature methods* 9.2 (2012), pp. 173–175.

[150] Jason A Reuter, Damek V Spacek, and Michael P Snyder. "High-throughput sequencing technologies." In: *Molecular cell* 58.4 (2015), pp. 586–597.

[151] Timothy J Richmond. "Solvent accessible surface area and excluded volume in proteins: Analytical equations for overlapping spheres and implications for the hydrophobic effect." In: *Journal of molecular biology* 178.1 (1984), pp. 63–89.

[152] Darrell S Rigel, Robert J Friedman, Alfred W Kopf, and David Polsky. "ABCDE—an evolving concept in the early detection of melanoma." In: *Archives of dermatology* 141.8 (2005), pp. 1032–1034.

[153] Richard Rosenquist, Edwin Cuppen, Reinhard Buettner, Carlos Caldas, Helene Dreau, Olivier Elemento, Geert Frederix, Sean Grimmond, Torsten Haferlach, Vaidehi Jobanputra, et al. "Clinical utility of whole-genome sequencing in precision oncology." In: *Seminars in cancer biology*. Elsevier. 2021.

[154] Burkhard Rost, Reinhard Schneider, and Chris Sander. "Protein fold recognition by prediction-based threading." In: *Journal of molecular biology* 270.3 (1997), pp. 471–480.

[155] Daniel Ruiz, Vicente Berenguer, Antonio Soriano, and Belén Sánchez. "A decision support system for the diagnosis of melanoma: A comparative approach." In: *Expert Systems with Applications* 38.12 (2011), pp. 15217–15223.

[156] Hasim Sak, Andrew W Senior, and Françoise Beaufays. "Long short-term memory recurrent neural network architectures for large scale acoustic modeling." In: *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)* (2014).

[157] Martina Sanlorenzo, Igor Vujic, Christian Posch, Akshay Dajee, Adam Yen, Sarasa Kim, Michelle Ashworth, Michael D Rosenblum, Alain Algazi, Simona Osella-Abate, et al. "Melanoma immunotherapy." In: *Cancer biology & therapy* 15.6 (2014), pp. 665–674.

[158] Rahul Sarkar, Chandra Churh Chatterjee, and Animesh Hazra. "Diagnosis of melanoma from dermoscopic images using a deep depthwise separable residual convolutional network." In: *IET Image Processing* 13.12 (2019), pp. 2130–2142.

[159] Heiko Schöder, Steven M Larson, and Henry WD Yeung. "PET/CT in oncology: integration into clinical management of lymphoma, melanoma, and gastrointestinal malignancies." In: *Journal of Nuclear Medicine* 45.1 suppl (2004), 72S–81S.

[160] V. Shah, P. Autee, and P. Sonawane. "Detection of Melanoma from Skin Lesion Images using Deep Learning Techniques." In: *Proceedings of the International Conference on Data Science and Engineering (ICDSE)*. 2020, pp. 1–8. DOI: 10.1109/ICDSE50459.2020.9310131.

[161] Rajalingappaa Shanmugamani. *Deep Learning for Computer Vision: Expert techniques to train advanced neural networks using TensorFlow and Keras*. Packt Publishing Ltd, 2018.

[162] Connor Shorten and Taghi M Khoshgoftaar. "A survey on image data augmentation for deep learning." In: *Journal of Big Data* 6.1 (2019), pp. 1–48.

[163] Karen Simonyan and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." In: *arXiv preprint arXiv:1409.1556* (2014).

[164] Radomir M Slominski, Michal A Zmijewski, and Andrzej T Slominski. "The role of melanin pigment in melanoma." In: *Experimental dermatology* 24.4 (2015), p. 258.

[165] Ruben Smith, Pontus Klein, Yeliz Koc-Schmitz, Henry J Waldvogel, Richard LM Faull, Patrik Brundin, Markus Plomann, and Jia-Yi Li. "Loss of SNAP-25 and rabphilin 3a in sensory-motor cortex in Huntington's disease." In: *Journal of neurochemistry* 103.1 (2007), pp. 115–123.

[166] David J Spiegelhalter, A Philip Dawid, Steffen L Lauritzen, and Robert G Cowell. "Bayesian analysis in expert systems." In: *Statistical science* (1993), pp. 219–247.

[167] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. "Dropout: a simple way to prevent neural networks from overfitting." In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[168] Nils Strodthoff, Patrick Wagner, Markus Wenzel, and Wojciech Samek. "UDSMProt: universal deep sequence models for protein classification." In: *Bioinformatics* 36.8 (2020), pp. 2401–2409.

[169] Joel L Sussman, Dawei Lin, Jiansheng Jiang, Nancy O Manning, Jaime Prilusky, Otto Ritter, and Enrique E Abola. "Protein Data Bank (PDB): database of three-dimensional structural information of biological macromolecules." In: *Acta Crystallographica Section D: Biological Crystallography* 54.6 (1998), pp. 1078–1084.

[170] R Bryan Sutton, Dirk Fasshauer, Reinhard Jahn, and Axel T Brunger. "Crystal structure of a SNARE complex involved in synaptic exocytosis at 2.4 Å resolution." In: *Nature* 395.6700 (1998), pp. 347–353.

[171] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1–9.

[172] Damian Szklarczyk, John H Morris, Helen Cook, Michael Kuhn, Stefan Wyder, Milan Simonovic, Alberto Santos, Nadezhda T Doncheva, Alexander Roth, Peer Bork, et al. "The STRING database in 2017: Quality-controlled protein–

protein association networks, made broadly accessible." In: *Nucleic Acids Research* (2016), gkw937.

[173] Arthur Tenenhaus, Alex Nkengne, Jean-François Horn, Camille Serruys, Alain Giron, and Bernard Fertil. "Detection of melanoma from dermoscopic images of naevi acquired under uncontrolled conditions." In: *Skin Research and Technology* 16.1 (2010), pp. 85–97.

[174] Maya Topf and Andrej Sali. "Combining electron microscopy and comparative protein structure modeling." In: *Current opinion in structural biology* 15.5 (2005), pp. 578–585.

[175] Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions." In: *Scientific data* 5.1 (2018), pp. 1–9.

[176] Daniel Ungar and Frederick M Hughson. "SNARE protein structure and function." In: *Annual review of cell and developmental biology* 19.1 (2003), pp. 493–517.

[177] UniProt Consortium. "UniProt: A hub for protein information." In: *Nucleic Acids Research* 43.D1 (2015), pp. D204–D212.

[178] Mario Vailati-Riboni, Valentino Palombo, and Juan J Loor. "What are Omics Sciences?" In: *Periparturient Diseases of Dairy Cows*. Springer, 2017, pp. 1–7.

[179] Wil Van Der Aalst. "Data science in action." In: *Process mining*. Springer, 2016, pp. 3–23.

[180] Timothy D Veenstra. "Omics in systems biology: Current progress and future outlook." In: *Proteomics* 21.3-4 (2021), p. 2000235.

[181] Jakob Vesterstrøm. "Heuristic algorithms in bioinformatics." PhD thesis. Citeseer, 2005.

[182] Eugenio Vocaturo, Ester Zumpano, and Pierangelo Veltri. "Image pre-processing in computer vision systems for melanoma detection." In: *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE. 2018, pp. 2117–2124.

[183] SG WEKA. "The Waikato Environment for Knowledge Analysis." In: *University of Waikato, Hamilton, New Zealand: University of Waikato* (1995).

[184] Fiona M Walter, A Toby Prevost, Joana Vasconcelos, Per N Hall, Nigel P Burrows, Helen C Morris, Ann Louise Kinmonth, and Jon D Emery. "Using the 7-point checklist as a diagnostic aid for pigmented skin lesions in general practice: a diagnostic validation study." In: *British Journal of General Practice* 63.610 (2013), e345–e353.

[185] James Z Wang, Zhidian Du, Rapeeporn Payattakool, Philip S Yu, and Chin-Fu Chen. "A new method to measure the semantic similarity of GO terms." In: *Bioinformatics* 23.10 (2007), pp. 1274–1281.

[186] Charles M Washington and Dennis T Leaver. *Principles and Practice of Radiation Therapy-E-Book*. Elsevier Health Sciences, 2015.

[187] Rolf H Weber and Romana Weber. *Internet of things*. Vol. 12. Springer, 2010.

[188] Leyi Wei, PengWei Xing, Ran Su, Gaotao Shi, Zhanshan Sam Ma, and Quan Zou. "CPPred-RF: a sequence-based predictor for identifying cell-penetrating peptides and their uptake efficiency." In: *Journal of Proteome Research* 16.5 (2017), pp. 2044–2053.

[189] Qing Wei, Ishita K Khan, Ziyun Ding, Satwica Yerneni, and Daisuke Kihara. "NaviGO: Interactive tool for visualization and functional similarity and coherence analysis with gene ontology." In: *Bmc Bioinformatics* 18.1 (2017), pp. 1–13.

[190] Qiong Wei and Roland L Dunbrack Jr. "The role of balanced training and testing data sets for binary classifiers in bioinformatics." In: *PloS one* 8.7 (2013), e67863.

[191] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. "A survey of transfer learning." In: *Journal of Big data* 3.1 (2016), pp. 1–40.

[192] Darrell Whitley, Timothy Starkweather, and Christopher Bogart. "Genetic algorithms and neural networks: Optimizing connections and connectivity." In: *Parallel Computing* 14.3 (1990), pp. 347–361.

[193] Leigh Willard, Anuj Ranjan, Haiyan Zhang, Hassan Monzavi, Robert F Boyko, Brian D Sykes, and David S Wishart. "VADAR: a web server for quantitative evaluation of protein structure quality." In: *Nucleic acids research* 31.13 (2003), pp. 3316–3319.

[194] Xindong Wu, Vipin Kumar, J Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, Geoffrey J McLachlan, Angus Ng, Bing Liu, S Yu Philip, et al. "Top 10 algorithms in data mining." In: *Knowledge and information systems* 14.1 (2008), pp. 1–37.

[195] Anmu Xie, Jing Gao, Lin Xu, and Dongmei Meng. "Shared mechanisms of neurodegeneration in Alzheimer's disease and Parkinson's disease." In: *BioMed Research International* (2014).

[196] Jingwen Yan, Shannon L Risacher, Li Shen, and Andrew J Saykin. "Network approaches to systems biology analysis of complex disease: Integrative methods for multi-omics data." In: *Briefings in Bioinformatics* 19.6 (2018), pp. 1370–1381.

[197] Xiaofei Yang, Yea Jin Kaeser-Woo, Zhiping P Pang, Wei Xu, and Thomas C Südhof. "Complexin clamps asynchronous release by blocking a secondary Ca2+ sensor via its accessory α helix." In: *Neuron* 68.5 (2010), pp. 907–920.

[198] Lequan Yu, Hao Chen, Qi Dou, Jing Qin, and Pheng-Ann Heng. "Automated melanoma recognition in dermoscopy images via very deep residual networks." In: *IEEE transactions on medical imaging* 36.4 (2016), pp. 994–1004.

[199] Zhen Yu, Xudong Jiang, Feng Zhou, Jing Qin, Dong Ni, Siping Chen, Baiying Lei, and Tianfu Wang. "Melanoma recognition in dermoscopy images via aggregated deep convolutional features." In: *IEEE Transactions on Biomedical Engineering* 66.4 (2018), pp. 1006–1016.

[200] Xiaojing Yuan, Ning Situ, and George Zouridakis. "A narrow band graph partitioning method for skin lesion segmentation." In: *Pattern Recognition* 42.6 (2009), pp. 1017–1028.

[201] Jianpeng Zhang, Yutong Xie, Yong Xia, and Chunhua Shen. "Attention residual learning for skin lesion classification." In: *IEEE transactions on medical imaging* 38.9 (2019), pp. 2092–2103.

[202] Xiaoqing Zhang. "Melanoma segmentation based on deep learning." In: *Computer Assisted Surgery* 22.sup1 (2017), pp. 267–277.

[203] Yu Zhou and Zhuoyi Song. "Binary decision trees for melanoma diagnosis." In: *Proceedings of the International Workshop on Multiple Classifier Systems*. Springer. 2013, pp. 374–385.