Democratic and Popular Republic of Algeria Ministry of
Higher Education and Scientific ResearchUniversity
Mohamed Khider of Biskra

Faculty of Exact Sciences and Science of Nature and Life
Department of Computer Science

Ref : ……..

Thesis Presented to obtain the degree of

Doctorate 3rd CYCLE IN COMPUTER SCIENCE

Option: Artificial intelligence and image processing

Entitled:

Deep Learning technique and parallel optimization algorithm for intelligent pattern recognition

Presented by:

Hakima Rym RAHAL

Publicly defended on:

19/02/2025

In front of the Jury committee composed of:

Mr. Rezeg Khaled	Professor	University of Biskra	President
Mme. Slatnia Sihem	Professor	University of Biskra	Supervisor
Mr. Kazar Okba	Professor	University of Charjah	Co-Supervisor
Mr. BenHarzallah Saber	Professor	University of Batna2	Examiner
Mr. Lejdel Brahim	Professor	University of Eloued	Examiner
Mr. Belouar Hocine	MCA	University of Biskra	Examiner

Acknowledgements

First, I want to thank almighty ALLAH for giving me the will, patience and health to develop this work. I would like to express my profound gratitude to my supervisors Pr. Sihem SLATNIA, Pr. Okba KAZAR, Pr. Ezedin BARKA, and Pr. Saad HAROUS for their involvement in this research work and for the support they have given me, their patience, their availability and the relevance of their advice that have been of invaluable assistance throughout this work. Also, I would like to thank all my Professors in the L'INFI Laboratory, and the Computer Science Department. I extend my sincere thanks to the members of the jury, for having accepted to judge this work. Finally, I thank my Mother, and My Aunties and Uncles for supporting me, and the rest of my family and friends for encouraging me during these years, and to have always been available when I needed it.

Abstract

Misdiagnosis poses a significant challenge within the healthcare sector, carrying potentially severe consequences for patients, including delayed or inappropriate treatment, unnecessary medical procedures, emotional distress, financial burdens, and legal repercussions. To address this issue, we propose the utilization of deep learning algorithms to enhance the precision of medical diagnoses. However, the development of accurate deep learning models for medical purposes necessitates substantial quantities of top-quality data, a resource that can be challenging for individual healthcare entities to acquire. Consequently, there is a need to aggregate data from various sources to create a diverse dataset suitable for effective model training. Nevertheless, the sharing of medical data across different healthcare sectors is fraught with security concerns due to the sensitive nature of the information and stringent privacy regulations. To tackle these complex challenges, we advocate for the adoption of Blockchain technology, which offers a secure, decentralized, and privacy-centric approach to sharing locally trained deep learning models, thereby obviating the need to exchange raw data. Our proposed technique, known as model ensembling, combines the strengths of multiple local deep learning models by aggregating their weights to construct a unified global model. This global model enables accurate diagnosis of intricate medical conditions across various locations while safeguarding patient privacy and data integrity. Our research serves as a testament to the efficacy of this approach, achieving high accuracy rates in the diagnosis of three diseases (accuracy of 97.44 % for the Breast Cancer, 97.14 % for the Diabetes, and 98.51 % for the Lung Cancer) that surpass those of individual local models. Furthermore, we have successfully developed a multi-diagnosis application as an outcome of this innovative methodology.

Keywords : ***Blockchain, Medical Big data, Deep Learning (DL) methods, Pattern Recognition.***

Résumé

Le diagnostic erroné représente un défi significatif au sein du secteur de la santé, avec des conséquences potentiellement graves pour les patients, notamment un traitement retardé ou inapproprié, des procédures médicales inutiles, une détresse émotionnelle, des charges financières et des conséquences juridiques. Pour faire face à ce problème, nous proposons l'utilisation d'algorithmes d'apprentissage profond pour améliorer la précision des diagnostics médicaux. Cependant, le développement de modèles d'apprentissage profond précis à des fins médicales nécessite des quantités substantielles de données de haute qualité, une ressource qui peut être difficile à acquérir pour les entités de soins de santé individuelles. Par conséquent, il est nécessaire de regrouper des données provenant de diverses sources pour créer un ensemble de données diversifié adapté à une formation efficace des modèles. Néanmoins, le partage de données médicales entre différents secteurs de la santé est entaché de préoccupations en matière de sécurité en raison de la nature sensible des informations et des réglementations strictes en matière de confidentialité. Pour relever ces défis complexes, nous préconisons l'adoption de la technologie Blockchain, qui offre une approche sécurisée, décentralisée et axée sur la confidentialité pour le partage de modèles d'apprentissage profond formés localement, éliminant ainsi le besoin d'échanger des données brutes. Notre technique proposée, appelée "model ensembling", combine les forces de plusieurs modèles d'apprentissage profond locaux en agrégeant leurs poids pour construire un modèle global unifié. Ce modèle global permet un diagnostic précis de conditions médicales complexes dans divers endroits tout en préservant la confidentialité des patients et l'intégrité des données. Notre recherche témoigne de l'efficacité de cette approche, atteignant des taux de précision élevés dans le diagnostic de trois maladies (accuracy of 97.44 % for the Breast Cancer, 97.14 % for the Diabetes, and 98.51 % for the Lung Cancer) dépassant ceux des modèles locaux individuels. De plus, nous avons développé avec succès une application de diagnostic multiple en résultat de cette méthodologie innovante.

Mots-clés : ***Blockchain, Données médicales volumineuses, Méthodes d'apprentissage profond (DL), Reconnaissance de motifs.***

ملخص

يشكل التشخيص الخاطئ تحديًا كبيرًا في قطاع الرعاية الصحية، ويحمل عواقب وخيمة محتملة على المرضى، بما في ذلك العلاج المتأخر أو غير المناسب، والإجراءات الطبية غير الضرورية، والاضطرابات العاطفية، والأعباء المالية، والعواقب القانونية. لمعالجة هذه المشكلة، نقترح الاستفادة من خوارزميات التعلم العميق لتعزيز دقة التشخيصات الطبية. ومع ذلك، فإن تطوير نماذج التعلم العميق الدقيقة للأغراض الطبية يتطلب كميات كبيرة من البيانات عالية الجودة، وهو مورد قد يكون من الصعب على كيانات الرعاية الصحية الفردية الحصول عليه. وبالتالي، هناك حاجة إلى تجميع البيانات من مصادر مختلفة لإنشاء مجموعة بيانات متنوعة مناسبة للتدريب الفعال على النماذج. ومع ذلك، فإن تبادل البيانات الطبية عبر قطاعات الرعاية الصحية المختلفة محفوف بالمخاوف الأمنية بسبب الطبيعة الحساسة للمعلومات واللوائح الصارمة المتعلقة بالخصوصية. لمعالجة هذه التحديات المعقدة، ندعو إلى اعتماد تقنية بلوكتشاين، التي توفر نهجًا آمنًا ولامركزيًا ومركزًا على الخصوصية لمشاركة نماذج التعلم العميق المدربة محليًا، وبالتالي تجنب الحاجة إلى تبادل البيانات الخام. تجمع تقنيتنا المقترحة، المعروفة باسم تجميع النماذج، بين نقاط القوة في نماذج التعلم العميق المحلية المتعددة من خلال تجميع أوزانها لبناء نموذج عالمي موحد. يتيح هذا النموذج العالمي التشخيص الدقيق للحالات الطبية المعقدة عبر مواقع مختلفة مع حماية خصوصية المريض وسلامة البيانات. يُعد بحثنا بمثابة شهادة على فعالية هذا النهج، حيث حقق معدلات دقة عالية في تشخيص ثلاثة أمراض (دقة 97.44٪ لسرطان الثدي، و97.14٪ لمرض السكري، و98.51٪ لسرطان الرئة) والتي تتجاوز تلك الخاصة بالنماذج المحلية الفردية. علاوة على ذلك، نجحنا في تطوير تطبيق تشخيص متعدد كنتيجة لهذه المنهجية المبتكرة.

الكلمات المفتاحية: بلوكتشين، بيانات طبية ضخمة، أساليب التعلم العميق (*DL*)، التعرف على الأنماط.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Healthcare, an integral aspect of our lives, is intrinsically linked to the pervasive and intricate issue of misdiagnosis. This concern, while prevalent, serves as a poignant reminder of the multifaceted nature of modern medical practice and its formidable challenges. Misdiagnosis, characterized by the inaccurate identification of a medical condition, extends its influence across a wide spectrum, impacting patients and healthcare systems in profound ways. The repercussions encompass potential harm to patients, an escalation of healthcare costs, and a substantial strain on medical resources, all of which contribute to a complex web of consequences. At its core, the issue of misdiagnosis carries profound significance as it delves into matters of patient safety, the quality of care, and the overall efficiency of healthcare delivery systems.

Misdiagnosis is a critical medical error characterized by the incorrect identification or classification of a patient's medical condition, leading to inaccurate treatment, delayed intervention, or the prescription of inappropriate medications. This failure to provide an accurate diagnosis can result from various factors, including errors in clinical judgment, misinterpretation of medical tests, insufficient patient medical history, or inadequate communication among healthcare providers. Misdiagnosis poses substantial risks to patient safety and can have far-reaching consequences, including physical harm, emotional distress, and increased healthcare costs, emphasizing the importance of improving diagnostic accuracy in healthcare systems.

Addressing the multifaceted challenge of misdiagnosis requires a multifaceted approach. Healthcare systems around the world are increasingly turning to advanced technologies such as Artificial Intelligence (AI) and Machine Learning (ML) to assist healthcare professionals in achieving more accurate diagnoses. These technologies have the potential to analyze vast datasets of medical information, detect subtle patterns, and offer evidence-based recommendations to clinicians, thus augmenting their decision-making capabilities. Furthermore, initiatives aimed at improving healthcare education and fostering a culture of continuous learning among healthcare professionals are crucial steps in reducing misdiagnosis rates. Continuous medical education, interdisciplinary collaboration, and the sharing of best practices all contribute to mitigating the risks associated with misdiagnosis.

Creating accurate deep learning models for medical diagnosis in a hospital can be a formidable challenge, particularly when the institution does not have access to a sufficient volume of data. The accuracy and reliability of deep learning models hinge on the availability of diverse and extensive datasets that encompass a wide spectrum of patient demographics, medical conditions, and variables. In cases where a hospital's data pool is limited, the models may struggle to capture the full complexity and variability of real-world medical scenarios. This limitation can result in models that are prone to biases, overfitting, or generalization errors, ultimately compromising their diagnostic accuracy.

Furthermore, insufficient data can hinder the model's ability to recognize rare diseases or conditions, as it may not have encountered enough instances to learn the relevant patterns and nuances. The lack of comprehensive data may also impede the model's capacity to adapt to evolving medical knowledge and practices, as it may not have access to a broad range of cases and outcomes.

To address the complexities associated with limited data resources, hospitals often engage in proactive measures and strategic collaborations. One prominent strategy involves fostering collaborative efforts with other healthcare institutions and research centers. By pooling their data resources and collective expertise, hospitals can create more robust and comprehensive datasets that encompass a wider array of medical conditions and patient profiles. These collaborative endeavors not only enhance the quality of the data available for deep learning model training but also promote a culture of knowledge sharing and interdisciplinary cooperation within the healthcare community.

While collaborative efforts can significantly enhance the quality of healthcare research and diagnostic capabilities, they also introduce complexities related to the protection of sensitive patient information. One prominent concern revolves around data privacy, as healthcare institutions are entrusted with vast amounts of personal and confidential patient data. The sharing of such data necessitates stringent safeguards to ensure that individuals' privacy rights are upheld, and their sensitive medical information remains secure. Healthcare organizations must navigate a labyrinth of regulatory requirements, such as the Health Insurance Portability and Accountability Act (HIPAA) in the United States or the General Data Protection Regulation (GDPR) in Europe, which impose strict guidelines on data sharing and security.

Another security limitation pertains to the potential for data breaches or unauthorized access when healthcare organizations share data. Even with stringent security measures in place, the interconnected nature of collaborative efforts may introduce vulnerabilities that malicious actors can exploit. The risk of data leakage or cyberattacks is a constant concern, necessitating robust encryption protocols, access controls, and auditing mechanisms to monitor and mitigate potential security threats.

Blockchain technology emerges as a promising solution to address the security limitations inherent in collaborative data sharing between healthcare organizations. One of its primary advantages lies in its decentralized and immutable nature. Blockchain enables healthcare institutions to create a secure, transparent, and tamper-proof ledger where data transactions are recorded in a series of blocks, with each block linked to the previous one, forming a chain. This decentralized structure eliminates the need for a central authority, reducing the risk of data breaches or unauthorized access by malicious entities. Data stored on the blockchain is encrypted and can only be accessed by authorized parties with the corresponding private keys, ensuring robust access controls and data security. But the risk of data leakage remains.

In response to this critical challenge, we have devised an innovative approach that leverages the potent technique of model ensembling. This method harnesses the collective strengths of numerous localy trained deep learning models, uniting them to form a formidable and resilient global deep learning model. The fundamental premise underlying this approach is the combination of the final weights from multiple local models, each meticulously trained on data originating from specific locations or sources. This fusion culminates in the creation of a single global model, endowed with the capacity to provide accurate predictions across a diverse spectrum of data inputs. The strength of this approach lies not only in its capability to enhance the accuracy but also in its unwavering commitment to data security and privacy.

We've put this innovative technique into action by creating a practical multi-disease diagnosis application that could accurately diagnose three different diseases: Breast cancer, Lung cancer, and Diabetes. And this application was created using local deep learning models created by three hospitals collaboration. Two of these hospitals focused on building local artificial neural network (ANN) models for each disease, using their own private data. They then shared these models with the third hospital on blockchain. This third hospital had the challenging task of combining these local models into three strong global deep learning models, one for each disease. These global models, created by merging the expertise from different sources, were further trained using the third hospital's private data. The end result was a single, comprehensive diagnostic system that embodies collaborative innovation. And, to ensure data security, we used blockchain technology to safely share this tool among the participating hospitals.

The results of this collaborative effort show the effectiveness of our approach. The global model for Breast cancer achieved an outstanding accuracy rate of 97.44%, outperforming the local models by a remarkable margin of 2.7% for the first model and 2.71% for the second. Similarly, the Lung cancer global model exhibited exceptional accuracy, soaring to 98.51%, surpassing the local models by 2.99% for the first model and a substantial 5.97% for the second. Finally, the Diabetes global model posted an impressive accuracy rate of 97.14%, effectively outstripping the local models by 11.43% for the first model and 2.85% for the second. These remarkable outcomes underscore the effectiveness of our approach in healthcare diagnostics.

The rest of the Thesis is organized as follows: Chapter 2 is a study on the previous works focused on the same topic as this thesis with a comparison between them. Chapter 3 holds an overview on the problem of Misdiagnosis in the healthcare sector, in addition to the effectiveness and basics of the Deep Learning techniques. Chapter 4 presents the importance of the security and privacy in the healthcare field, also the privacy techniques used to safeguard patient data. Chapter 5 represents the contribution and methods developed in this project with the results obtained.

Chapter 2

State of the Art

2.1 Introduction

In this chapter, we explore research related to our thesis to understand what others have studied. Our goal is to evaluate the strengths and weaknesses of these earlier studies and get a complete view of what they've contributed to the field. By doing this, we place our research in the context of the larger academic landscape. As we examine these related works, we look for similarities and differences, helping us see where our research is unique. This detailed exploration also helps us see where our work aligns with or differs from previous studies. Additionally, we introduce the specific diseases we'll focus on in our research. This chapter sets the stage for our work, building on past research while addressing its limitations to move the field forward.

2.2 Previous Works

2.2.1 A Predictive Tool for Ovarian Cancer based on Blockchain technology

In this study [11], M. Abraham et al. have developed a new tool to predict ovarian cancer. This tool relies on advanced technology called a CNN Siamese network, which can identify genetic mutations linked to cancer by analyzing microscopic images of proteins. These images were obtained from the 'Human Protein Atlas' database.

What's particularly exciting is that this system not only helps predict cancer but also ensures the safe sharing of healthcare data, patient records, and cancer predictions among different organizations and research labs. They achieve this secure sharing through a technology called blockchain.

The tool's performance has been tested and it showed an accuracy rate of 86%. However, it's essential to note that it has some limitations, such as being tested on a small number of classes and making predictions based on unseen data, as only the model's weights were shared. This means there is still room for improvement and further research in this important area of cancer prediction.

2.2.2 Framework for Diagnosing COVID-19 using X-ray Images

The swift rise in COVID-19 cases globally has posed a significant threat. To address this, a novel deep learning-based detection framework has been introduced to aid in the early diagnosis of COVID-19 from chest X-ray images in this study [12] by H. Nasiri et al. This framework utilizes the powerful DenseNet169 network to extract image features. To improve efficiency and accuracy, the features are carefully selected using the analysis of variance (ANOVA) method, followed by classification with eXtreme Gradient Boosting (XGBoost).

The model was trained on the ChestX-ray8 dataset and achieved remarkable accuracy rates of 98.72% for distinguishing between COVID-19 and No-findings, as well as 92% for classifying COVID-19, No-findings, and Pneumonia. Notably, this approach outperforms existing methods in the field. However, it is essential to acknowledge that data privacy considerations were not addressed in this study, highlighting the need for future research to address this crucial aspect of healthcare data management.

2.2.3 A Blockchain-Powered Expert System for Healthcare Emergencies

This article [13] presents the "Emotional Medical System Administrator," a predictive system that uses audiovisual emotion patterns to identify healthcare emergencies by R. C. Aguilera et al. Emotion recognition helps determine if a patient needs immediate attention, reducing costs. The system employs Convolutional Neural Networks (CNN) and Kalman filters for pattern recognition, achieving up to 84.1% accuracy. Patient emotions are captured using a device with a camera, microphone, and oscilloscope. Data is stored in a centralized database for regular and intense emotions, with blockchain used for data security. Expanding the database can enhance accuracy but may slow learning. This system enhances healthcare decision-making while prioritizing data security.

2.2.4 A Blockchain-based Deep Learning Platform for Myopia

In recent years, myopia has seen a rapid global proliferation, emerging as a substantial public health concern. This surge has led to an increasing prevalence of high myopia among individuals, putting them at risk of severe vision-related complications, such as myopic macular degeneration. In response, T. E. Tan et al. have developed a deep learning-based platform designed for myopia diagnosis in [14], capable of detecting high myopia and myopic macular degeneration. The model comprises three deep learning algorithms rooted in ResNet-101 (CNN), with two focused on high myopia identification and the third dedicated to diagnosing myopic macular degeneration.

To train and test the model effectively, diverse datasets from Singapore, India, Taiwan, Russia, China, and the United Kingdom were employed. A key challenge in employing deep learning algorithms on a global scale lies in securely exchanging data and models across various hospitals and centers in different countries. To address this concern, blockchain technology was integrated into the platform to enhance security and foster trust among collaborators.

Remarkably, the model exhibited impressive accuracy rates, reaching up to 91.3% for high myopia and 96.9% for myopic macular degeneration. Comparative tests pitted the deep learning algorithms against six human experts tasked with assessing a random set of 400 images from external datasets. The results were telling, with the deep learning algorithms outperforming all six experts, achieving 97.8% accuracy for myopic macular degeneration and 97.3% accuracy for high myopia.

Nevertheless, certain limitations warrant consideration. The algorithm can solely detect myopic macular degeneration as a binary outcome (present or absent) and cannot ascertain its underlying causes. Altering definitions would necessitate retraining the deep learning algorithms. Additionally, the algorithm may struggle with very low-resolution images. Data privacy concerns persist, as the dataset remains susceptible to sharing by collaborators.

2.2.5 A Blockchain-based Smartphone Platform for Diagnosing Malaria

In remote rural regions with limited resources, where infectious diseases often impose a substantial burden, there exists a significant challenge in promptly communicating test results to healthcare professionals during the disease detection process. In response to this challenge, this article [15] introduces an innovative smartphone-based platform designed for multiplexed DNA malaria diagnostics by X. Guo et al. The approach involves the utilization of an affordable paper-based microfluidic diagnostic test and harnesses the power of a Convolutional Neural Network (CNN) to support local decision-making. Furthermore, blockchain technology plays a pivotal role in data management and secure data connectivity.

Field testing in remote areas of Uganda has demonstrated the platform's remarkable efficacy, successfully detecting over 98% of cases. Beyond ensuring the secure geotagging of diagnostic data, this technology also enables the integration of infectious disease data into surveillance systems. To train the deep learning decision support system, a dataset comprising five categories and 92 test images obtained from loop-mediated isothermal amplification (LAMP) diagnostic tests was utilized.

Nonetheless, it is essential to acknowledge a security concern associated with the temporary retention of data on the smartphone for subsequent transmission to the cloud. This aspect warrants careful consideration to mitigate potential risks associated with data handling and privacy.

2.2.6 Utilizing blockchain for secure data sharing within healthcare systems

This study by R. Kumar et al. [16] introduces a novel data sharing framework named PBDL, which combines Deep Learning (DL) techniques with Private Blockchain and smart contracts to ensure security and efficiency. Initially, PBDL employs blockchain for registration, verification (utilizing zero-knowledge proof), and authentication of collaborating parties before implementing a smart contract-based consensus mechanism.

Furthermore, a pioneering DL approach is proposed, integrating Stacked Sparse Variational AutoEncoder (SSVAE) with Self Attention-based Bidirectional Long Short Term Memory (SA-BiLSTM), using the verified data. In this system, SA-BiLSTM plays a crucial role in identifying and enhancing attack detection methods, while SSVAE transforms healthcare data into a new format. Security analysis and experimentation, involving the IoT-Botnet and ToN-IoT datasets, underscore the superior performance of the PBDL system, achieving an accuracy rate of up to 99.89% for ToN-IoT and 99.98% for IoT-Botnet.

However, to further assess its performance and scalability, it is recommended to explore a software-defined network version of this approach.

2.2.7 A Personal Health Record (PHR) Application built on Blockchain Technology

Personal health records (PHRs) play a pivotal role in enhancing healthcare and patient safety by providing essential health information like allergies, medication dosages, and test results. They prove especially valuable in emergency scenarios where patients may be incapacitated or suffer memory loss, thus avoiding redundant and time-consuming tests. Despite their utility, PHRs are susceptible to security vulnerabilities that raise concerns about data integrity and patient privacy. To address these concerns, this study [17] by J. W. Kim et al, introduces an innovative blockchain-based PHR application designed to securely store and share personal medical data. The application securely retrieves and encrypts patients' personal information, storing it on the blockchain to mitigate the risk of data tampering and fraud. Furthermore, the study emphasizes user-centricity by developing a mobile PHR application leveraging blockchain technology. It's important to note that the study's scope was constrained by limited participant numbers and time, preventing a comprehensive evaluation of the application's usability and practicality.

2.2.8 A deep Learning Model for Detecting and Evaluating Sports Injuries

To effectively detect and assess the risk of sports-related medical conditions, this research [18] by H. Song et al. employs an optimized convolutional neural network (OCNN) designed for categorizing sports medical data. Additionally, it incorporates the Self-Adjustment Resizing algorithm (SAR) enhanced by the self-coding method (SCM) within the convolutional neural network. This approach enables multi-dimensional analysis of sports medicine data and the establishment of an advanced medical data network for sports medicine using a cloud-based loop model. The study primarily utilizes medical data provided by a professional sports laboratory. The OCNN demonstrates an accuracy of approximately 80%. However, future research enhancements should concentrate on achieving improved evaluations of sports injury data, particularly by refining neural networks for time series data analysis.

2.2.9 A CNN model based on blockchain for assessing lung cancer's food quality and well-being aspects

This study [19] by M. A. Aboamer aimed to dissect the influence of various factors, including features, filters, resolution, kernel size, epoch values, and padding value, on the accuracy of a lung cancer prediction model that integrates CNN and Blockchain. The research also ventured into the realm of food quality assessment.

Key findings indicated that efficient feature and filter utilization, coupled with image augmentation and a substantial dataset, significantly improved CNN accuracy in lung cancer prediction and food safety assessment. Notably, the study identified an optimal range of 10-12 epochs for achieving an impressive 99% accuracy, with accuracy declining when surpassing this threshold. Additionally, a correlation between image resolution and accuracy was observed, where reduced resolution corresponded with enhanced accuracy. In summary, the model attained an impressive overall accuracy rate of 92.5

However, it's essential to acknowledge potential limitations, leading the researcher to conduct supplementary investigations that yielded varied data, highlighting the intricate nature of this research domain.

2.2.10 Leveraging a blockchain-powered digital pathology system to enhance diagnostic processes

The application of artificial intelligence algorithms in machine-assisted disease detection is facing growing challenges due to the increasing complexity of data collection, diagnosis, and concerns related to storage, transmission, and security. In response, H. Subramanian et al. the authors of this paper [20] have developed and prototyped a decentralized, secure, and privacy-respecting digital pathology system. This system is built upon Ethereum-based smart contracts, the nonfungible token (NFT) standard, and the Interplanetary File System for data storage. The proposed solution, when integrated into data systems, has the potential to accelerate the diagnostic process, reduce processing time, enhance service quality, and facilitate access to specialized pathological diagnostics. Moreover, this approach can be extended to other medical specialties requiring high-fidelity imaging and data storage.

2.2.11 Heart Disease Prediction Algorithm

A new algorithm called Sine Cosine Weighted K-Nearest Neighbor (SCA-WKNN), developed by H. Hasanova et al. [21], employs machine learning to predict heart disease. This algorithm relies on blockchain technology for secure and tamper-resistant storage of patient data. To assess its performance, SCA-WKNN was compared to other algorithms in terms of accuracy, precision, recall, F-score, and root mean square error. The results demonstrated that SCA-WKNN outperformed both W K-NN and KNN algorithms, achieving a maximum accuracy rate difference of 4.59% and 15.61%, respectively. This innovation holds promise for enhancing disease prediction and healthcare efficiency, but further research is necessary to fully unlock its potential.

2.2.12 Blockchain-Based Self-Defined Access Control for Data Security

E. A. Mantey and colleagues have introduced a fresh approach to access control, granting users the autonomy to establish their access policies independently, devoid of the need for data owner authorization. This innovation hinges on the implementation of an Access Control Repository (ACR) as an integral facet of an identity-based access control framework. Furthermore, the ACR seamlessly integrates with smart contracts, which are then distributed across a Blockchain network by data owners. This multifaceted system is meticulously crafted to fortify the security of personal data and erect an impenetrable barrier against the potential leakage of sensitive information. Additionally, the article delves into a method for the detection of COVID-19 within X-ray images. Leveraging the prowess of Deep Learning, Keras, and TensorFlow, this technique has yielded remarkable outcomes, boasting an impressive accuracy range of 90-95% [22].

As we can see, some of the works didn't implement any protection for the healthcare data they worked on, and some of them did use Blockchain for either store or share the data with other healthcare organization, but still the risk that one of the collaborators may leak the data intentionally or unintentionally outside the system. This limit is dangerous and can't be ignored.

2.3 The Main Focus Diseases in Our Implementation

We have selected three distinct diseases as our focal points to rigorously validate the effectiveness of our methodology and to showcase its versatility and applicability across a range of medical conditions. These diseases serve as comprehensive test cases, allowing us to thoroughly examine the performance, accuracy, and generalizability of our approach:

2.3.1 Breast Cancer

Breast cancer is a form of cancer that originates in the cells of the breast, primarily affecting women. If left untreated, can infiltrate nearby tissues and potentially spread to other parts of the body, a process called metastasis. While the exact cause of breast cancer remains partially understood, various risk factors, including genetic, hormonal, and environmental influences, contribute to its development. Advances in medical research and treatment, coupled with early detection methods such as mammography, clinical breast exams, and biopsies, have significantly improved the prognosis. Treatment approaches vary but often encompass surgery, radiation therapy, chemotherapy, hormone therapy, or targeted therapy, tailored to the individual characteristics of the cancer and the patient's health. Regular breast cancer screenings and self-examinations are vital for timely identification and successful management [23].

Diagnosing whether a breast cancer is malignant or benign can be a complex and challenging task due to several factors. One of the primary difficulties lies in the often subtle and overlapping characteristics of malignant and benign breast tumors, both in clinical presentation and medical imaging. Distinguishing between the two requires a highly accurate assessment, and even experienced clinicians may encounter difficulties in certain cases. Additionally, there is a need to minimize the possibility of misdiagnosis, as misclassifying a malignant tumor as benign can have life-threatening consequences, while inaccurately categorizing a benign tumor as malignant can lead to unnecessary and invasive treatments. Therefore, accurate diagnosis often requires a combination of clinical expertise, advanced medical imaging techniques, and, increasingly, the utilization of artificial intelligence and deep learning models to aid in precise and reliable classification [24].

2.3.2 Lung Cancer

Lung cancer is a malignant and life-threatening disease that originates in the tissues of the lungs, primarily in the cells lining the air passages. It is one of the most common types of cancer worldwide and is often associated with long-term exposure to tobacco smoke, as well as other environmental and genetic factors. The symptoms of lung cancer may include persistent coughing, chest pain, shortness of breath, coughing up blood, and unexplained weight loss. Early diagnosis and appropriate treatment are critical for improving the chances of survival, making lung cancer a subject of significant research and medical advancements [25].

Diagnosing lung cancer is a complex task primarily due to the disease's asymptomatic early stages, non-specific symptoms that overlap with other conditions, and the existence of multiple lung cancer types with similar presentations. To establish a lung cancer diagnosis, a combination of diagnostic tests, including imaging procedures and tissue biopsies, is often necessary, and these tests may carry associated risks. Additionally, challenges in lung cancer screening, particularly for individuals not considered high-risk, can hinder early detection. Moreover, determining the precise stage of lung cancer, which is vital for treatment planning, is a multifaceted process that necessitates various tests. Consequently, ongoing research aims to enhance early detection and diagnostic accuracy in lung cancer, addressing these intricate diagnostic challenges [26].

2.3.3 Diabetes

Diabetes is a chronic medical condition that affects how the body processes glucose (sugar), a crucial source of energy. There are two main types of diabetes: Type 1 and Type 2. In Type 1 diabetes, the immune system mistakenly attacks and destroys insulin-producing cells in the pancreas, leading to a lack of insulin. Type 1 diabetes typically begins in childhood or adolescence and requires lifelong insulin therapy for blood sugar control. Type 2 diabetes is more common and usually develops in adulthood. In this form of diabetes, the body's cells become resistant to the effects of insulin, and the pancreas may not produce enough insulin to compensate. This results in elevated blood sugar levels. Type 2 diabetes can often be managed with a combination of lifestyle changes, dietary modifications, and, in some cases, medication or insulin. Uncontrolled diabetes can lead to various complications, including heart disease, kidney problems, nerve damage, and vision issues. Therefore, proper management and monitoring of blood sugar levels are essential for individuals with diabetes [27].

Diagnosing diabetes can be challenging due to several factors. One significant difficulty arises from the fact that diabetes, especially in its early stages, may not always exhibit noticeable symptoms. As a result, individuals with diabetes can remain undiagnosed for an extended period. Additionally, the symptoms of diabetes, when present, can overlap with those of other health conditions, leading to misdiagnosis or delayed diagnosis. Moreover, there isn't a single definitive test for diabetes, and diagnosis often requires a combination of blood tests to measure blood sugar levels, such as fasting glucose tests, oral glucose tolerance tests, and HbA1c tests. Physicians must carefully interpret these results while considering a patient's medical history and risk factors. As a result, diagnosing diabetes necessitates a high degree of clinical suspicion, vigilance, and an awareness of the diverse ways it can manifest, which can be a challenge for healthcare providers [28].

2.4 Conclusion

In this chapter, we have delved into an extensive review of various research endeavors aimed at addressing critical challenges within the healthcare industry. Our exploration has offered valuable insights into the strengths and weaknesses of these studies. Notably, we have observed that some research papers primarily focused on solving intricate healthcare issues but often overlooked the crucial aspect of data security. Conversely, others recognized the significance of safeguarding sensitive medical data and employed blockchain technology as a means to store and share information among different healthcare organizations. While these blockchain-based solutions did provide a degree of security for medical data, it's essential to acknowledge that the risk of data leakage still looms large.

In light of these findings, it becomes evident that there exists an unmet need for comprehensive and foolproof data security measures in the healthcare domain. The current landscape underscores the necessity for innovative solutions that not only harness the power of cutting-edge technologies like blockchain but also ensure the utmost protection of patients' confidential information. As we move forward, it is imperative that future research endeavors place an increased emphasis on data security, employing robust mechanisms that leave no room for potential breaches. Only through such dedicated efforts we can aspire to create a healthcare ecosystem that prioritizes patient privacy and data integrity above all else, ultimately ushering in a new era of trust and confidence in medical data management.

Also in this chapter, we have not only conducted a comprehensive overview of the diseases central to our study but have also delved into the intricate complexities and challenges associated with diagnosing each of these medical conditions. By providing an extensive examination of the diseases, we aim to offer a holistic understanding of the multifaceted nature of these health issues and the diagnostic obstacles they present.

In the upcoming chapter, we will delve into the critical issue of misdiagnosis in healthcare, examining the potential consequences it holds for patients and the healthcare system. We'll also explore the fundamental concepts of deep learning, a powerful branch of artificial intelligence, and how it can play a role in addressing this challenge.

Chapter 3

Misdiagnosis and Deep Learning

3.1 Introduction

Misdiagnosis in healthcare is a critical issue, often leading to incorrect treatments, delayed interventions, and patient harm. Deep learning offers promise in mitigating this risk by leveraging its ability to analyze vast datasets and identify subtle patterns in medical information. Through advanced algorithms, deep learning models can enhance diagnostic accuracy, aiding healthcare professionals in making more informed decisions. These models can process complex medical data, such as images, genetic profiles, and patient histories, to identify diseases at earlier stages, predict patient outcomes, and recommend tailored treatment plans. By harnessing the power of deep learning, healthcare can substantially reduce the risk of misdiagnosis, ultimately improving patient care and outcomes.

In this chapter we are going to present an overview on the risks of Misdiagnosis and how can deep learning help reduce this problem, in addtion to the basic concepts of deep learning.

3.2 Misdiagnosis in Healthcare

Diagnostic mistakes happen when medical practitioners fail to accurately and promptly describe a patient's health problems or fail to adequately communicate that explanation to the patient. These mistakes are described as missed chances to accurately and quickly identify a patient's illness while taking into account the information at hand at the moment. Diagnostic mistakes, in general, refer to a variety of circumstances in which medical professionals may misinterpret signs and symptoms, misread test results, or fail to notice important details. Such mistakes could result in inaccurate or delayed diagnoses, which could affect the patient's recovery and general well-being. These missed chances for timely or accurate patient diagnosis can have serious repercussions, hurting the patient's physical health as well as their emotional well-being and faith in the healthcare system [29].

3.2.1 Statistics in the European Union

Current figures in the European Union show an unsettling reality: 23% of its citizens have personally or within their family personally experienced the effects of a medical misdiagnosis. In this group, a sizeable 18% reported being a victim of a serious diagnostic error that happened in a hospital setting, and another 11% said they had received the wrong prescription for a medication.

The degree of vulnerability to diagnostic errors varies among the European Union's member states, according to a detailed examination. With rates of 32%, 29%, and 28%, respectively, Latvia, Denmark, and Poland stand out as countries with a greater chance of hospital mishaps. Similar to Estonia and Malta, Latvia and Denmark are noted as nations with higher rates of medication errors, with rates of 23%, 21%, and 18%, respectively [30].

3.2.2 Statistics in the U.S.A

The number of misdiagnoses that occur each year among Americans seeking outpatient medical care services is close to 12 million, or almost 1 in every 20 patients. According to studies, of the 12 million incidents of misdiagnosis, between 10% and 20% involve individuals with serious medical disorders, with 44% of those patients having cancer kinds that were misdiagnosed. The situation is made more serious by the fact that breast, thyroid, and prostate cancer are the cancers that are misdiagnosed the most frequently. Furthermore, 28% of these misdiagnosed instances are considered to be life-threatening or life-altering, which results in unneeded treatments, higher healthcare expenses, and stress on the afflicted people's physical and mental health. Unfortunately, in the worst-case instances, a misdiagnosis may potentially be fatal. Additionally, getting a second opinion in the U.S. medical system reveals startling statistics. Incredibly, 66% of patients who get a second opinion have their original diagnosis changed, illustrating the prevalence of misdiagnosis. 21% of people who seek a second opinion have their diagnosis completely changed, underscoring the degree of diagnostic uncertainty present in the healthcare system. Only 12% of patients are fortunate enough to have their initial diagnosis confirmed, highlighting the importance of getting a second opinion as a vital preventative measure against misdiagnosis. This last causes a new type of harm to the patient, not only physically and emotionally, but also financially[30].

3.2.3 Deep Learning Effectiveness

In the effort to prevent misdiagnosis, the utilization of dependable and accurate diagnostic methods becomes paramount, particularly in intricate and challenging cases. Deep learning, as a revolutionary technology in the field of artificial intelligence, holds immense potential to significantly mitigate the occurrence of misdiagnosis. By processing vast volumes of medical data and discerning intricate patterns and correlations that may elude human specialists, deep learning algorithms can play a pivotal role in revolutionizing the diagnostic process. The integration of deep learning in healthcare can yield substantial benefits, particularly in complex scenarios where traditional diagnostic approaches may fall short. Through its ability to analyze diverse patient data, including medical images, clinical records, and genetic information, deep learning can uncover unusual patterns indicative of specific diseases or conditions. As a result, medical professionals can obtain more accurate and precise diagnoses, significantly reducing the risk of misdiagnosis and its associated negative consequences [31].

3.3 Deep Learning Basics

Deep neural networks draw inspiration from the hierarchical organization of human brains, where they gradually learn simpler features and then process them into more abstract representations [32, 33]. A typical deep neural network, known as a feedforward network, comprises an input layer, multiple hidden layers, and an output layer. In the simplest form of deep architecture, the multi-layer perceptron (MLP) network, the output is computed directly by passing the input data through consecutive layers of the model. Within each neuron of the hidden layers, a weighted sum of the previous layer's outputs is subjected to a nonlinear activation function, producing the neuron's output. The basic architecture of a deep neural network is represented in Fig.1. The hierarchical nature of representation learning in deep learning enables it to discover meaningful but abstract correlations and patterns in large datasets [34]. Figure 2.1 represents the basic architecture of a Neural Network.

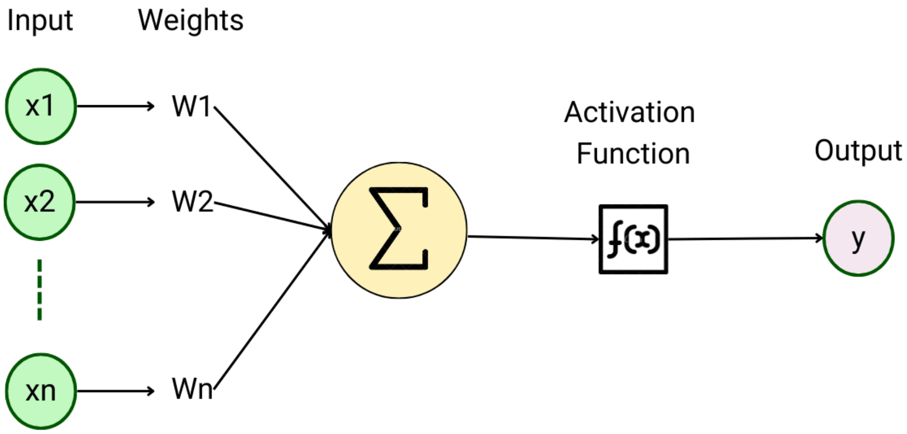


Figure 3.1 – Basic Architecture of a Neural Network [1]

Using input data, a deep learning system performs a specified task without the need for explicit programming or hard-coding for a predetermined result. Instead, because these algorithms repeatedly modify and adapt their underlying design, they are referred to as "soft-coded" algorithms. During this ongoing modification, referred to as training, the algorithm is given samples of the input data together with the expected results. The algorithm adjusts its configuration throughout the training phase to deliver the intended results accurately with the training data in addition to generalizing its understanding and performing well with new, previously unknown information [35]. Effective generalization is a key component of deep learning processes. The "learning" component of deep learning is analogous to the training phase. It can involve continual progress and "lifelong" learning, similar to how humans learn from novel situations and failures, and is not restricted to a finite initial adaption. Deep learning algorithms essentially go through an iterative and flexible learning process, improving their performance and skills over time by being exposed to a variety of data. Due to their versatility and capacity to adjust to shifting settings, they are extremely useful tools in contemporary artificial intelligence applications [36].

3.3.1 Deep Neural Network Architectures

3.3.1.1 Artificial Neural Networks (ANNs)

Artificial Neural Networks (ANNs) form the fundamental building blocks of deep learning structures [37]. A standard feedforward neural network comprises an input layer, several hidden layers, and an output layer. Within the hidden layers, each neuron receives inputs weighted accordingly, applies an activation function, and then transmits the outcome to the subsequent layer [38, 39, 40].

In mathematical terms, in the case of a basic feedforward ANN featuring a solitary hidden layer, the output of each neuron can be expressed as follows:

$$Output_i = \sigma\left(\sum_{j=1}^{n} w_{ij} \cdot Input_j + b_i\right)$$

Where:
— $Output_i$ is the output of the i^{th} neuron.
— σ is the activation function.
— w_ij is the weight between the i^{th} neuron's input j and the j^{th} neuron's output.
— $Input_j$ is the output of the j^{th} neuronin the previous layer.
— b_i is the bias term of the i^{th} neuron.

3.3.1.2 Recurrent Neural Networks (Unidirectional RNN)

Unidirectional Recurrent Neural Networks (RNNs) are specifically crafted to handle sequential data, with each neuron in the hidden layer receiving inputs not only from the preceding layer but also from its own output in the preceding time step [41]. This unique architecture allows the network to retain temporal information and capture patterns in sequential data [42].

The hidden state h_t of a unidirectional RNN at time step t can be represented by the following equation:

$$h_t = RNN\_Cell(Input_t, h_{t-1})$$

Where:
— RNN_Cell is the RNN cell that combines the current $Input_t$ and the previous hidden state h_{t-1} to compute the new hidden state h_t.

3.3.1.3 Recurrent Neural Networks (Bidirectional RNN)

Bidirectional Recurrent Neural Networks (RNNs) are an advanced version of unidirectional RNNs designed to handle sequential data in two directions, both forward and backward [43]. By doing so, they capture information from both previous and future time steps, enhancing their ability to understand temporal patterns in the data [44].

The hidden state h_t of a bidirectional RNN at time step t can be represented as:

$$h_t = RNN\_Cell\_Forward(Input_t, h_{t-1}) + RNN\_Cell\_Backward(Input_t, h_{t+1})$$

Where:
— $RNN\_Cell\_Forward$ is the RNN cell for the forward pass.
— $RNN\_Cell\_Backward$ is the RNN cell for the backward pass.
— h_{t-1} is the previous hidden state in the forward pass.
— h_{t+1} is the next hidden state in the backward pass.

3.3.1.4 Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) are specifically tailored for processing image and spatial data. These networks employ convolutional layers, which utilize filters to extract distinctive features from the input data [45, 46]. Subsequently, pooling layers are applied to decrease spatial dimensions, and fully connected layers are utilized for tasks like classification or regression [47].

The output of a CNN's convolutional layer can be represented as:

$$Output_i = \sigma\left(\sum_{j=1}^{n} w_{ij} * Input_j + b_i\right)$$

Where:
— $Output_i$ is the output of the i^{th} feature map in the convolutional layer.
— σ is the activation function.
— w_ij is the convolutional filter's weight.
— $Input_j$ is the input data or a feature map from a previous layer.
— b_i is the bias term of the i^{th} feature map.

3.3.1.5 Autoencoders (AEs)

The Autoencoder (AE) is a model designed to facilitate feature learning and reduce dimensionality. It employs an encoder to convert input data into a lower-dimensional representation and a decoder to reconstruct the original input based on this compressed representation [48, 49].

The encoder function f and the decoder function g can be represented as follows:

Encoder:

$$h = f(Input)$$

Decoder:

$$Reconstructed\_Input = g(h)$$

Where:
— h is the lower-dimensional representation (latent space).
— f is the encoder function.
— g is the decoder function.

3.3.1.6 Stacked Autoencoders

A stacked autoencoder is an advanced version of the AE, comprising several layers of encoders and decoders, which results in the formation of a deep architecture [50].

The output of a stacked autoencoder's encoder function f can be represented as:

$$h = f^{(L)}\left(f^{(L-1)}\left(\dots\left(f^{(1)}\left(Input\right)\right)\right)\right)$$

Where:
— $f^{(L)}$ is the encoder function for the L^{th} layer.

3.3.1.7 Variational Autoencoders (VAEs)

A Variational Autoencoder (VAE) is an enhanced version of the AE that incorporates probabilistic encoding into the latent space. This integration enables the model to perform sampling within the latent space, facilitating the generation of new data [51].

The encoder function f and decoder function g of a VAE can be represented as:

Encoder:

$$h, \mu, \sigma = f(Input)$$

Decoder:

$$Reconstructed\_Input = g(Sample(\mu, \sigma))$$

Where:
— h is the lower-dimensional representation (latent space).
— μ is the mean of the latent space distribution.
— σ is the standard deviation of the latent space distribution.
— f is the encoder function.
— g is the decoder function.
Figure 2.2 represents the architectures of each model explained earlier.

3.3.2 Overfitting and Underfitting

When training machine learning models, overfitting and underfitting are two frequent issues. They relate to how successfully a model extrapolates from the training set of data to fresh or unknown data.

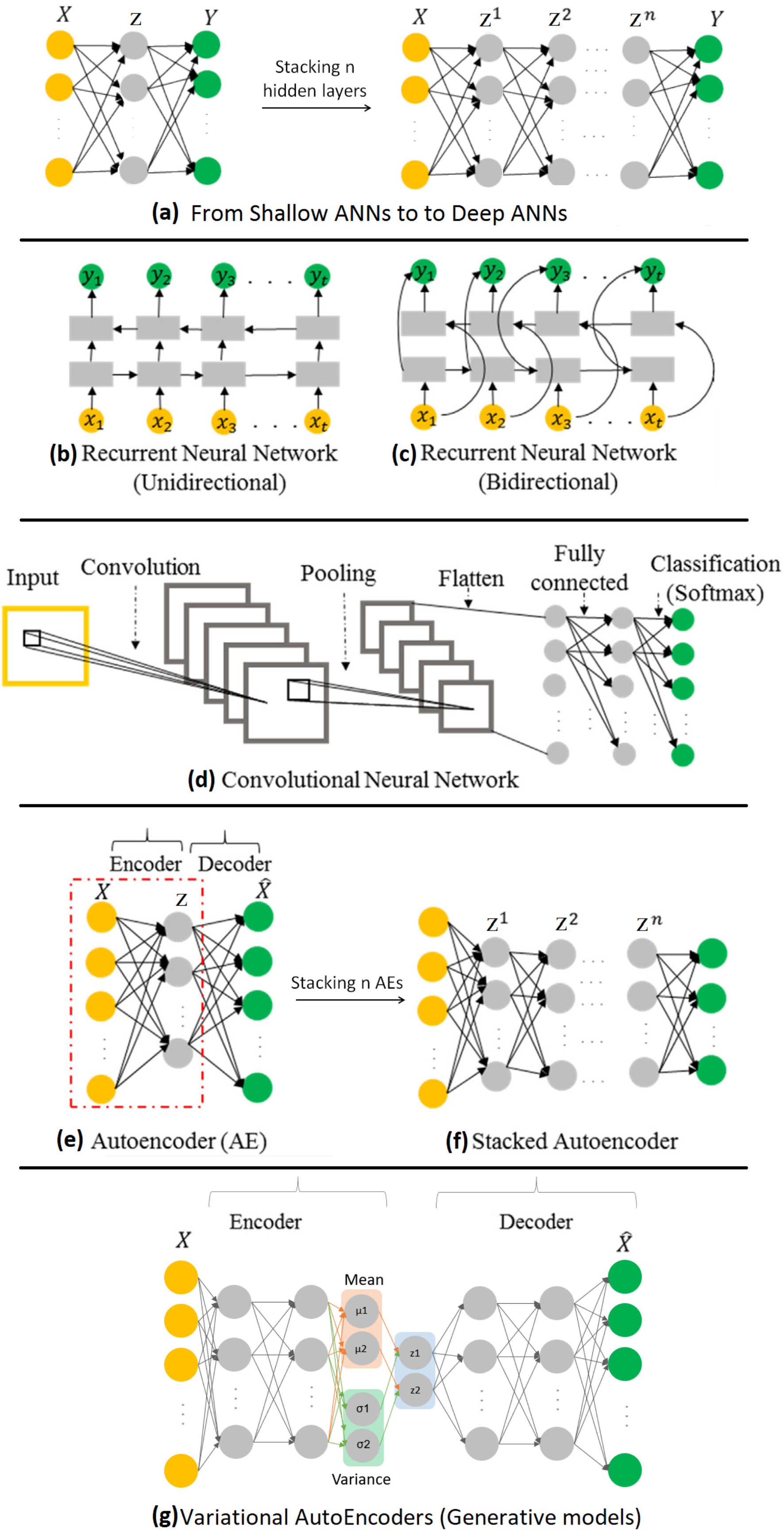
3.3.2.1 Overfitting

When a deep learning model learns the training data too thoroughly, overfitting happens. This causes the model to collect noise and random fluctuations rather than the underlying patterns or relationships in the data. Because the model has effectively memorized the training data instead of learning to make generalizations, it performs very well on the training data but poorly on fresh, unknown data. very complicated models with several parameters, a low training error but a large validation or test error, and models that are very sensitive to slight perturbations in the training data are all indications of overfitting [52].

3.3.2.2 Underfitting

On the other side, underfitting happens when a machine learning model is too basic to accurately represent the underlying patterns in the training data. Because the model hasn't learnt enough about the structure of the data, it performs poorly on both the training data and new, unseen data. High training error and high validation or test error are indicators of underfitting [53].

Figure 2.3 represents the types of fitting in a deep learning model.



(a) From Shallow ANNs to to Deep ANNs

(b) Recurrent Neural Network (Unidirectional)

(c) Recurrent Neural Network (Bidirectional)

(d) Convolutional Neural Network

(e) Autoencoder (AE)

(f) Stacked Autoencoder

(g)Variational AutoEncoders (Generative models)

Figure 3.2 – Deep Neural Network Architectures [2]

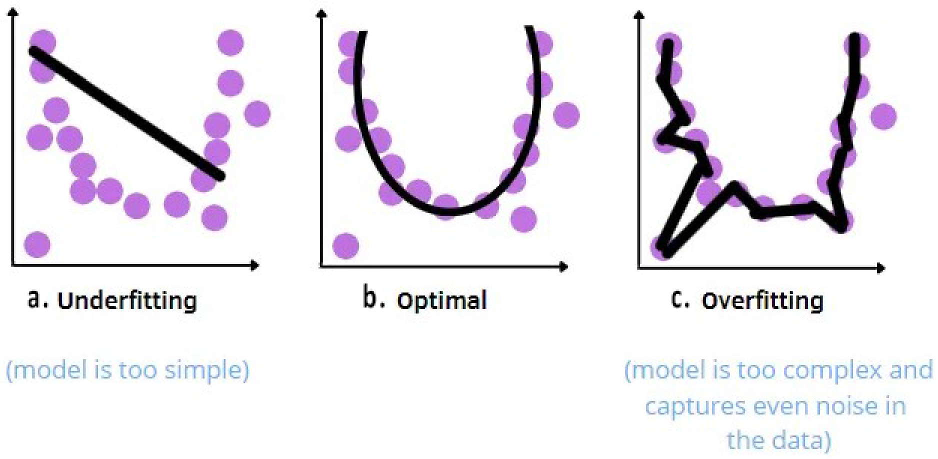


Figure 3.3 – Overfitting and Underfitting [3]

3.3.3 Regularizations in Deep Learning

Regularization is a crucial deep learning approach that enhances neural network robustness, prevents overfitting, and improves generalization. When a model learns to fit the training data too closely, overfitting happens. This captures noise and leads to subpar outcomes on unobserved data. Regularization techniques impose restrictions or penalties on the model during training in order to promote more basic models and lessen the chance of overfitting [54, 55].

3.3.3.1 L1 Regularization (Lasso Regularization)

L1 regularization, a widely employed technique in the realm of deep learning, plays a pivotal role in enhancing the generalization capabilities of neural networks by incorporating an additional term into the loss function during the training process. The primary objective of this regularization method is to combat the issue of overfitting, which often arises when the model becomes excessively complex and fits the training data too closely, resulting in poor performance on unseen data. The distinctive characteristic of L1 regularization lies in the penalty term it introduces, which is intricately linked to the absolute values of the model's weights. Specifically, it imposes a constraint on these weights by adding a penalty proportional to the absolute magnitudes of the weights. This penalty term effectively encourages certain weights to become precisely zero, essentially driving a selection process among the model's features. Consequently, it fosters sparsity within the model, allowing it to focus on the most relevant features while disregarding less significant ones [56].

3.3.3.2 L2 Regularization

L2 regularization stands as a fundamental and widely adopted technique within the domain of deep learning, wielding the power to enhance the performance and generalization capabilities of neural networks. Its overarching goal is to tackle the notorious issue of overfitting, the introduction of a penalty term that is intricately linked to the squares of the model's weights. This penalty term plays a crucial role during the training process, as it imposes a constraint on the magnitude of the model's weights. By proportionately penalizing the square values of these weights, L2 regularization encourages the model to favor small weight values. In doing so, it effectively deters the model from becoming overly reliant on any single feature or becoming overly complex, thereby mitigating the risks associated with overfitting [57].

3.3.3.3 Dropout

The dropout technique is a remarkable innovation in the realm of neural network training that has revolutionized the way we approach deep learning. Implemented during each forward and backward pass of the training process, dropout introduces an element of controlled randomness by selectively omitting a random subset of neurons within the neural network. This strategy has profound implications for the model's performance, as it effectively disrupts the intricate co-dependencies and intricate interactions that can develop among neurons during training. The primary motivation behind dropout is twofold: to combat overfitting and to deter the co-adaptation of neurons. By randomly deactivating neurons, dropout introduces an element of uncertainty into the training process, forcing the model to be more robust and adaptive. This, in turn, mitigates overfitting by ensuring that the model doesn't rely too heavily on any specific set of neurons or features [58].

3.3.3.4 Batch Normalization

Batch normalization, a pivotal technique in deep learning, brings a sophisticated layer of complexity and effectiveness to the training of neural networks. It operates by normalizing the activations of each layer within a mini-batch during the training process, a seemingly simple yet highly impactful strategy. This normalization plays a multifaceted role in the optimization of neural networks, with far-reaching implications for their overall performance and stability. One of the primary benefits of batch normalization lies in its capacity to stabilize the training process itself. Deep neural networks often exhibit the challenge of vanishing or exploding gradients, which can impede convergence and hinder effective learning. Batch normalization counteracts this issue by rescaling and shifting the activations within each mini-batch, effectively reducing the magnitude of gradients, thereby facilitating smoother and more efficient training [59, 60].

3.3.3.5 Early Stopping

The concept of early stopping represents a strategic and highly effective approach in deep learning. This technique revolves around the continuous monitoring of the model's performance on a separate validation dataset during the training process. Instead of allowing the model to train until it converges fully, early stopping intervenes by terminating the training process when a noticeable rise in the validation error occurs. This proactive measure is taken to ensure that the model reaches its optimal generalization capacity, thereby preventing it from overfitting the training data [61, 62].

3.3.4 Combination of Deep Learning Models

One effective strategy for enhancing the performance, robustness, and generalization of deep learning models is combining them. This method is referred to as model ensembling. Combining deep learning models can be done in a variety of ways, each with advantages and uses.

3.3.4.1 Transfer Learning

Transfer learning encapsulates a powerful paradigm that leverages the wealth of knowledge gained in one specific context or task to enhance performance in another, often closely related context or task. At its core, transfer learning adheres to the premise that models aren't isolated entities but rather dynamic reservoirs of information and expertise. When a model is exposed to and trained on a specific task, it not only acquires the ability to tackle that task but also accumulates a wealth of knowledge, patterns, and insights that can transcend the confines of its initial application. Transfer learning capitalizes on this notion, enabling models to repurpose and adapt the acquired knowledge to new, analogous problems, thereby streamlining the learning process and amplifying their effectiveness [63].

Pre-trained Models Deep learning's transfer learning strategy is intricately entwined with the utilization of pre-trained models, which serve as the bedrock for knowledge transfer and enhancement of model performance. These pre-trained models represent neural networks that have undergone rigorous training on vast and diverse datasets, typically tailored to a specific task, such as image classification or natural language processing. The extensive training and fine-tuning of these models equip them with a rich understanding of the underlying patterns and intricacies of their designated domain [64].

3.3.4.2 Architectural Combinations

In deep learning, architectural combinations involve the artful fusion of diverse neural network designs or layers to create intricate and highly specialized models. The essence of architectural combinations lies in their capacity to innovate and push the boundaries of what deep learning models can achieve. By strategically integrating various neural network components, practitioners can craft models that are not only more powerful but also more adaptable and versatile. These combinations are not bound by rigid rules; instead, they encourage creativity and experimentation, allowing researchers and engineers to tailor models to the unique requirements of specific tasks or problem domains. [65, 66].

3.3.4.2.1 Model Ensembles Model ensembles present an intriguing and highly effective strategy for enhancing the performance and generalization capacity of predictive models. This approach goes beyond the confines of a single model by harnessing the collective wisdom and diversity of multiple models, which are typically of the same architecture but initialized differently or trained on distinct subsets of data. The overarching goal of ensemble learning is to mitigate the common pitfalls of overfitting while simultaneously improving the robustness and reliability of predictions. [65].

3.3.4.2.2 Bagging Bagging, short for Bootstrap Aggregating, amplifies model performance and foster robust generalization. This ingenious process entails the creation of multiple instances of the same base model, each trained using a distinct and randomly selected subset of the training data. The key innovation behind bagging lies in its ability to harness the diverse perspectives and insights that these independently trained models bring to the table, ultimately leading to more accurate and stable predictions [67].

3.3.4.2.3 Boosting Boosting offers a compelling approach to model building that transcends the limitations of individual weak learners. At its core, boosting is designed to create a robust and high-performing model, often referred to as a strong learner, by ingeniously combining the collective wisdom of multiple weak learners—models that exhibit only marginal improvements over random guessing. However, the true power of boosting lies not just in the amalgamation of these modestly performing models but also in its dynamic and iterative approach to model improvement [68]

3.3.4.2.4 Stacking Stacking represents a powerful methodology for synthesizing the collective wisdom of multiple base models into a highly proficient and accurate final prediction model. At its core, stacking introduces an innovative layer into the ensemble paradigm: a meta-learner. This meta-learner is tasked with learning how to effectively integrate the predictions generated by the diverse array of base models. In essence, stacking orchestrates an ensemble of ensembles, each contributing its unique perspective and expertise to the final prediction, with the meta-learner acting as the conductor of this harmonious orchestration [69].

3.3.4.2.5 Voting Voting presents a versatile and reliable strategy for amalgamating predictions from an array of models, including classifiers and regressors. These techniques play a pivotal role in harnessing the collective intelligence of multiple models and leveraging their combined insights to make more robust and accurate predictions. The crux of voting techniques revolves around the concept of aggregating the individual predictions of these models, either through averaging or majority voting, depending on whether the task at hand is regression or classification. [70].

3.3.4.2.6 Weight Averaging Weight averaging is a strategic approach to enhance the generalization and overall performance of neural network models. It emerges as an invaluable tool when faced with scenarios where multiple models of similar architectures have been meticulously trained, each encapsulating its unique learning experiences and insights. The goal of weight averaging is to leverage the collective knowledge embedded in these diverse models, ultimately yielding predictions that exhibit superior accuracy and robustness. [71].

3.3.4.2.7 Multi-Modal Architectures Multi-modal architectures represent a cutting-edge and innovative approach in the field of artificial intelligence. These architectural marvels are meticulously designed to handle data emanating from multiple diverse sources or modalities, bridging the gap between various forms of information representation. A quintessential example of multi-modal architecture involves seamlessly integrating text and graphics into a single model, fostering a holistic understanding of input data that transcends the boundaries of traditional unimodal approaches [72].

3.4 Conclusion

In this chapter, we've taken a good look at the various risks linked to misdiagnosis in healthcare. We've stressed how these risks can seriously affect patients, not just physically but also mentally. This underlines the vital need for precise medical diagnosis. At the same time, we've pointed out how deep learning methods can help deal with these issues, making diagnoses more accurate by using data-driven insights.

The next chapter will focus on the complex field of securing medical data. We'll explore how healthcare data security is closely connected to protecting patient privacy, ensuring data accuracy, and following strict data protection laws and legal rules. This balancing act between improving medical diagnosis and keeping patient records safe highlights the intricate and diverse challenges faced in modern healthcare.

Chapter 4

Security and Privacy in Healthcare

4.1 Introduction

Security and privacy concerns in healthcare are paramount, given the sensitive nature of patient data. Blockchain technology offers a compelling solution by establishing a decentralized, tamper-proof, and transparent ledger for securely managing medical information. Through encryption and smart contracts, blockchain ensures data integrity and access control, allowing patients to grant and revoke consent for data sharing, enhancing transparency and accountability. This robust framework strengthens data security, prevents unauthorized access, and mitigates the risk of breaches, addressing critical concerns in healthcare security and privacy.

In this Chapter we present a summary of the most privacy threatening risks related to medical data and the possible solutions that have been offered over the years until the use of blockchains.

4.2 Security and Privacy Challenges in Healthcare

Because healthcare organizations deal with extremely sensitive patient information, security and privacy issues are of utmost concern. To keep patients' trust and adhere to numerous rules like the Health Insurance Portability and Accountability Act (HIPAA) in the United States, it is essential to safeguard patient data and ensure the security and privacy of healthcare systems. Several of the major security and privacy issues in healthcare are listed below:

4.2.1 Data Breaches

The escalating frequency of data breaches in the healthcare sector has ignited growing concerns and highlighted the profound repercussions that can ensue from such security lapses. These breaches represent a concerning trend wherein cybercriminals relentlessly target healthcare organizations in their quest to access patient records. Within these records, a treasure trove of sensitive data awaits, including not only the obvious medical histories and insurance information but also critical identifiers such as Social Security numbers. The aggregation of this highly confidential information makes healthcare establishments prime targets for cybercriminals seeking to exploit vulnerabilities for malicious purposes. The consequences of these healthcare data breaches are far-reaching and multifaceted. Financial fraud emerges as one of the immediate and severe repercussions, as cybercriminals often use stolen information to perpetrate fraudulent activities, ranging from unauthorized financial transactions to the creation of fake identities for illicit purposes. The fallout from such fraudulent activities can be financially devastating for both the individuals whose data has been compromised and the healthcare organizations involved [73].

4.2.2 IoT Vulnerabilities

The proliferation of Internet of Things (IoT) devices in the healthcare sector, including medical equipment and wearable technology, has ushered in a new era of innovation and patient care. However, this surge in connectivity also raises a critical concern - the emergence of novel attack vectors. The sheer volume and diversity of these IoT devices have led to a complex security landscape, characterized by a wide range of vulnerabilities that hackers can potentially exploit. Unlike traditional medical equipment, many IoT healthcare gadgets may not have robust security safeguards in place, leaving them alarmingly susceptible to malicious intrusions [74].

4.2.3 Lack of Standardization

Security and privacy practices in the healthcare sector stand at a critical juncture where standardization remains a pressing concern. The vast and intricate web of healthcare providers, each operating within its unique ecosystem, results in a lack of uniformity in data protection measures. This variance in security procedures across different healthcare organizations has profound implications for the confidentiality and integrity of patient data. One significant contributing factor to the lack of standardization is the inherent complexity of healthcare operations. Various healthcare entities, such as hospitals, clinics, insurance providers, and research institutions, handle patient data differently, each governed by distinct regulatory frameworks and operational requirements. This diversity in organizational structure and objectives makes it challenging to enforce consistent security and privacy practices across the entire healthcare landscape [75].

4.2.4 Third-Party Vulnerabilities

The increasing reliance on third-party vendors within the healthcare industry has ushered in a new era of operational efficiency and technological advancement. Healthcare companies often turn to these vendors to avail themselves of a wide array of services, ranging from cloud storage solutions to Electronic Health Records (EHR) systems, which streamline and enhance patient care. However, this extensive integration of external services into the healthcare ecosystem brings with it a host of security and privacy considerations, as patient data is entrusted to these vendors. The potential consequences of security issues or data breaches on the part of these third-party providers loom large and encompass a myriad of intricate challenges. One of the central concerns surrounding third-party vendors in healthcare is the security of patient data. The sensitivity and confidentiality of medical records and personal health information make them prime targets for cybercriminals. If a third-party vendor were to experience a security breach or data leak, the ramifications could be devastating, potentially resulting in the exposure of patients' private medical histories, treatment plans, and sensitive personal details. Such breaches not only jeopardize patient trust but also carry legal and regulatory consequences for healthcare organizations that entrusted their data to these vendors [76].

4.2.5 Compliance Challenges

Healthcare providers operate in a global landscape governed by a complex web of regulatory frameworks that aim to ensure the privacy and security of patient data. These regulations span continents and nations, with each region having its unique set of compliance requirements. In the United States, healthcare providers must adhere to the Health Insurance Portability and Accountability Act (HIPAA), which sets stringent standards for the safeguarding of patient information. Meanwhile, in Europe, the General Data Protection Regulation (GDPR) imposes its own set of rigorous data protection rules. Navigating this intricate regulatory terrain poses a formidable challenge for healthcare organizations as they strive to deliver efficient and uninterrupted healthcare services while simultaneously upholding compliance with these multifaceted regulations. The complexity of healthcare regulations extends far beyond mere data protection. These regulations encompass a wide range of requirements, including those related to data access and consent, breach notification, record-keeping, and patient rights. Ensuring compliance in all these dimensions demands meticulous attention to detail, robust data management systems, and rigorous staff training programs. Moreover, the dynamic nature of healthcare, with its continuous evolution in technologies and practices, introduces an additional layer of complexity. Healthcare providers must not only maintain compliance with existing regulations but also adapt swiftly to new regulatory updates and emerging cybersecurity threats. This ongoing commitment to compliance often necessitates substantial investments in cybersecurity infrastructure, ongoing audits, and the cultivation of a culture of data privacy and security within healthcare organizations [77, 78].

4.2.6 Patient Consent and Data Access

One of the paramount challenges in modern healthcare revolves around the delicate balance between respecting patient privacy and ensuring efficient, timely, and secure access to medical records. Healthcare providers are tasked with the formidable responsibility of not only delivering effective care but also safeguarding sensitive patient information, ensuring that only authorized individuals, including healthcare professionals directly involved in a patient's treatment, have access to these critical records. This balance extends to the core of patient-doctor relationships, where trust and confidentiality are foundational. Patients entrust healthcare providers with their most personal and sensitive information, including medical histories, treatment plans, and diagnostic results. Maintaining the confidentiality of this data is not just a legal requirement but also an ethical imperative, as breaches of trust can have profound repercussions on patient well-being and the healthcare provider-patient relationship [79].

4.2.7 Data Encryption

The imperative to encrypt patient data, both in transit and at rest, stands as a foundational pillar of modern healthcare cybersecurity. Encryption serves as a robust safeguard, rendering sensitive medical information unreadable and inaccessible to unauthorized parties. While the benefits of encryption are undeniable, the practical implementation of this critical security measure in healthcare settings can be rife with complexities and challenges, particularly when striving to maintain the seamless operation of healthcare procedures.The implementation of encryption involves not only the deployment of encryption algorithms but also the development of comprehensive encryption strategies that align with the specific needs and intricacies of healthcare workflows. One of the primary challenges lies in striking the right balance between robust data protection and uninterrupted healthcare operations. Encrypting data in transit, such as when it is transmitted between medical devices and electronic health records (EHR) systems, must not introduce significant latency or disrupt the real-time nature of patient care. Similarly, encrypting data at rest within EHR databases or on portable storage devices, like laptops or tablets, necessitates strategies that do not compromise the timely retrieval of patient information when needed [80].

4.2.8 Emerging Technologies

The advent of transformative technologies like telemedicine and AI-based diagnosis tools has ushered in a new era of healthcare delivery, marked by increased accessibility and enhanced diagnostic capabilities. However, alongside these innovations come a host of intricate security and privacy concerns that demand meticulous attention. Safeguarding the privacy of AI-driven health advice and ensuring the security of remote patient consultations represent paramount challenges in the ever-evolving landscape of modern healthcare. Telemedicine, which enables remote patient consultations and treatments, brings with it a unique set of security challenges. The transmission of sensitive patient data across digital channels necessitates robust encryption and secure communication protocols to protect against data breaches and unauthorized access. Additionally, healthcare organizations must implement stringent access controls and authentication mechanisms to verify the identity of both patients and healthcare providers participating in virtual consultations. AI-based diagnosis tools, while revolutionizing medical decision-making, introduce privacy concerns related to the handling of patient data. These tools often rely on vast datasets, including patient health records, medical images, and genetic information. Protecting the privacy of patients within these datasets is of utmost importance, requiring rigorous data anonymization and de-identification techniques to prevent the inadvertent disclosure of personal information. Moreover, healthcare organizations must ensure that AI systems comply with data protection regulations such as GDPR in Europe and HIPAA in the United States [81].

4.3 Solutions for Security and Privacy in Healthcare

In order to handle diverse risks and weaknesses, security and privacy in healthcare necessitate a multifaceted strategy. Various approaches and techniques to improve security and privacy in healthcare include the following:

4.3.1 Access Control and Authentication

For the protection of private patient information and to maintain the integrity of healthcare systems, effective access control and strong authentication mechanisms are indispensable. The following are some of the most common access control and authentication techniques used by healthcare organizations:

4.3.1.1 Role-Based Access Control (RBAC)

Role-Based Access Control (RBAC) is a widely used approach to managing and controlling access to computer systems, networks, and applications. RBAC is a security model that assigns permissions and access rights to users based on their roles and responsibilities within an organization. Instead of granting permissions to individual users, RBAC groups users into roles and assigns permissions to those roles, simplifying access management and enhancing security [82].

The cornerstone of access control strategies in the healthcare industry is role-based access control (RBAC), a robust and widely adopted framework that meticulously assigns permissions and access rights based on the specific roles and responsibilities of healthcare personnel within an organization. In this meticulously designed system, the fundamental principle is to align access privileges with job functions, ensuring that each user, whether they are physicians, nurses, administrative staff, or IT professionals, is granted access only to the data and systems that are directly pertinent to their respective tasks and responsibilities. By doing so, RBAC significantly reduces the risk of unauthorized access to sensitive patient information, critical medical records, and other confidential data, thereby bolstering the overall security posture of healthcare institutions and ensuring compliance with stringent data protection regulations such as the Health Insurance Portability and Accountability Act (HIPAA) in the United States. This fine-grained access control not only safeguards patient privacy but also enhances operational efficiency by streamlining data access management and minimizing the potential for human error in granting or revoking access permissions [83].

Limits Role-based access control has some security limitations:

- RBAC frequently uses centralized access control systems, which leaves them open to single points of failure as well as malicious attacks.
- RBAC demands confidence in the central authority in charge of access control, which is vulnerable to misuse or mistakes.
- RBAC systems may struggle with interoperability between different organizations and platforms.

Figure 3.1 represents the mechanism of Role-based access control.

4.3.1.2 Multi-Factor Authentication (MFA)

Multi-Factor Authentication (MFA), also known as Two-Factor Authentication (2FA), is a security mechanism that requires users to provide two or more separate forms of identification before gaining access to a system, application, or account. MFA adds an extra layer of security beyond the traditional username and password combination, making it more difficult for unauthorized individuals to access sensitive information [84].

Implementing multi-factor authentication (MFA) for accessing healthcare systems and patient data is a critical step toward enhancing the security and integrity of sensitive medical information. MFA goes beyond the traditional username and password model, adding an extra layer of protection by requiring users to provide multiple forms of identification before granting access. This multifaceted approach not only strengthens the overall security posture but also significantly reduces the risk of unauthorized access and data breaches. In the case of healthcare, where the confidentiality of patient records is paramount, MFA serves as a robust safeguard against potential threats, ensuring that only authorized personnel can access critical medical data. This can involve various authentication factors, such as something the user knows (a password), something the user has (a mobile device or smart card), and something the user is (biometric data like fingerprint or facial recognition). By requiring this combination of factors, MFA creates a formidable barrier against cyberattacks and unauthorized intrusions, thereby safeguarding the privacy and well-being of patients while bolstering compliance with stringent

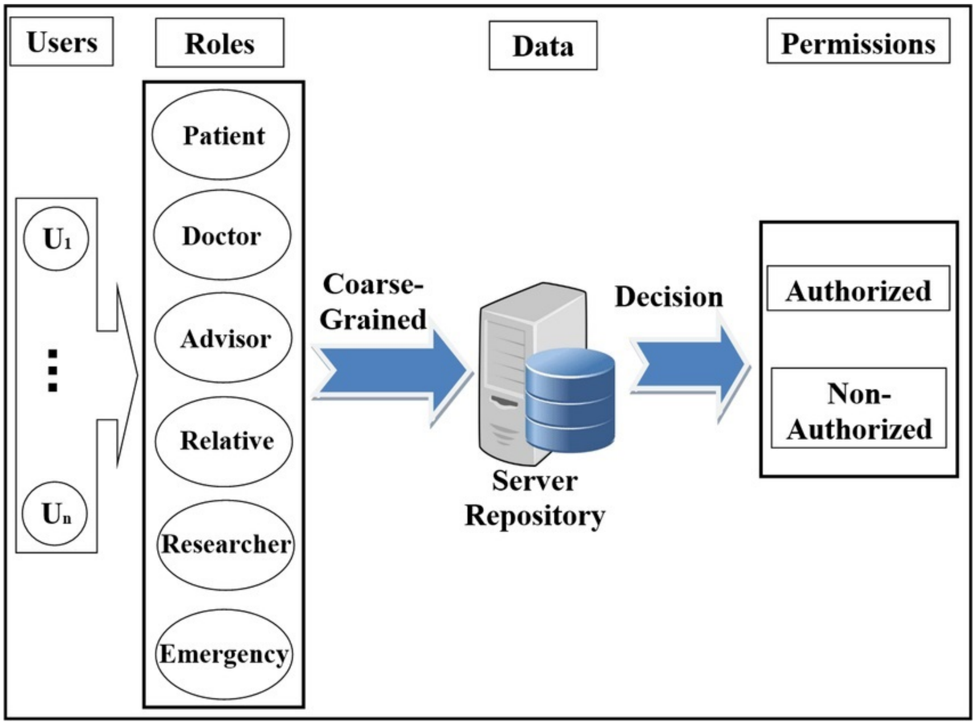


Figure 4.1 – Role-Based Access Control (RBAC) [4]

healthcare data protection regulations like the Health Insurance Portability and Accountability Act (HIPAA) in the United States [85].

Limits Multi-Factor Authentication has some security limitations:

— MFA systems may rely on centralized servers for authentication data, making them susceptible to single points of failure and data breaches.

— Users must trust MFA service providers with their sensitive data.

— MFA often involves sharing personal information with third parties.

Figure 3.2 represents the mechanism of Multi-Factor Authentication.

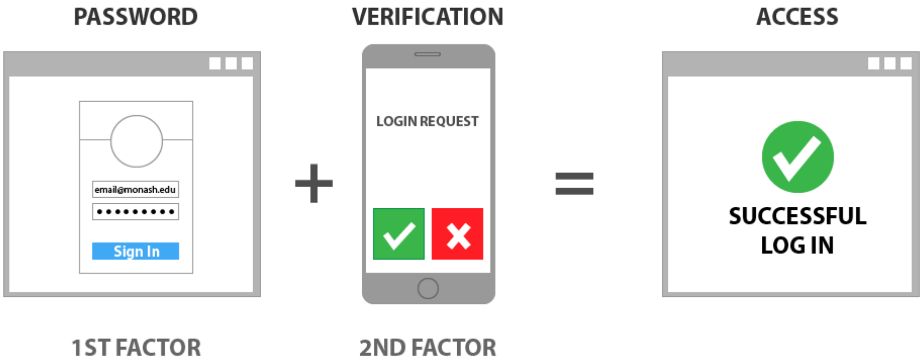


Figure 4.2 – Multi-Factor Authentication (MFA) [5]

4.3.2 Data Encryption

Data encryption is the process of converting plain, readable data (often referred to as plaintext) into an unreadable format (ciphertext) using encryption algorithms and cryptographic keys. The primary purpose of data encryption is to secure sensitive information during storage or transmission, making it difficult for unauthorized parties to access, interpret, or tamper with the data [86].

Ensuring the security and privacy of patient data in healthcare settings necessitates a comprehensive approach that includes robust encryption measures both at rest and in transit. Encrypting patient data at rest involves safeguarding it when it is stored in databases, servers, or other storage solutions. Strong encryption algorithms, such as Advanced Encryption Standard (AES) with appropriate key lengths, can be meticulously implemented to render the data unreadable to unauthorized individuals. Additionally, a well-structured key management system is crucial to securely generate, store, and manage encryption keys, preventing potential breaches resulting from compromised keys or weak key protection. Encrypting patient data in transit is equally vital, especially when transmitting information over networks, whether within a healthcare facility or across the internet. Transport Layer Security (TLS) or Secure Sockets Layer (SSL) protocols can be employed to establish secure connections and encrypt data during transmission. These protocols not only protect data from eavesdropping and interception but also ensure the integrity of the data, making it resistant to tampering during transit [87].

Limits

— Traditional encryption relies on centralized authorities for key management and verification, which can be vulnerable to breaches or misuse.

Figure 3.3 represents the mechanism of Data Encryption.

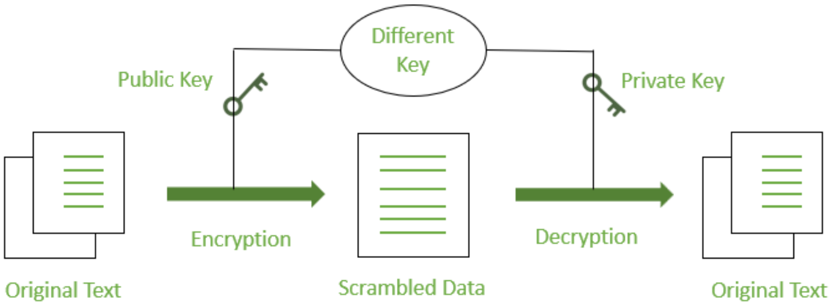


Figure 4.3 – Data Encryption

4.3.3 Secure Communication

4.3.3.1 Virtual Private Networks (VPNs)

A Virtual Private Network (VPN) is a technology that creates a secure and encrypted connection, often referred to as a "tunnel," over a less secure network, such as the internet. The primary purpose of a VPN is to enhance online privacy, security, and anonymity by routing the internet traffic through a remote server or network, making it appear as if the connection is originating from that server or network [88].

Leveraging Virtual Private Networks (VPNs) to ensure the security of data transmission over public networks is a fundamental strategy in safeguarding sensitive information. VPNs establish a secure and encrypted communication channel, effectively creating a virtual tunnel through which data can safely traverse public networks. This technology not only shields data from unauthorized access but also offers several additional layers of protection. By encrypting data packets and encapsulating them within the VPN tunnel, VPNs employ encryption algorithms like the aforementioned Advanced Encryption Standard (AES) to obfuscate data, rendering it virtually impossible for malicious actors to intercept or eavesdrop on sensitive information [89].

Limits Virtual Private Networks has some security limitations:

— VPNs rely on centralized servers, requiring trust in the VPN service provider.

— VPN security can be compromised if the devices or endpoints connecting to the VPN are themselves insecure or compromised.

Figure 3.4 represents the mechanism of Virtual Private Networks.

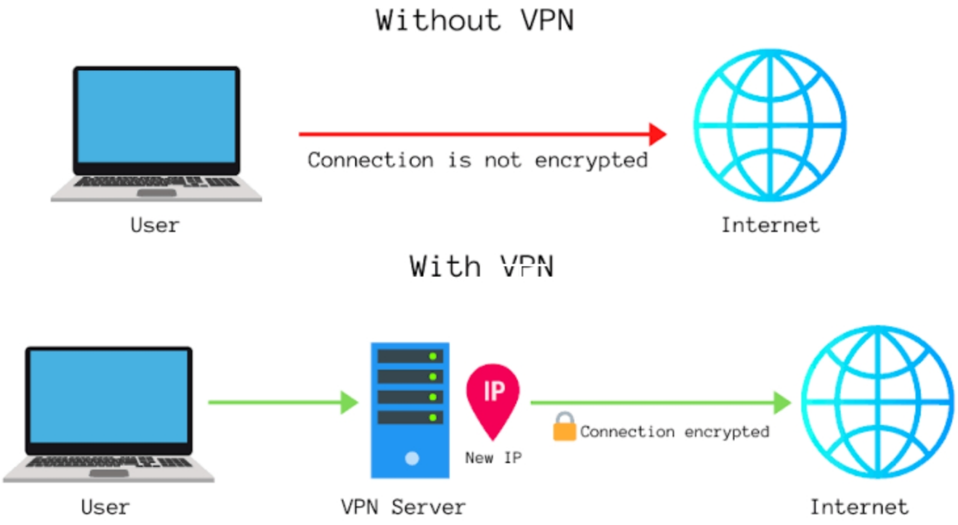


Figure 4.4 – Virtual Private Networks (VPNs) [6]

4.3.3.2 Secure Socket Layer (SSL)/Transport Layer Security (TLS)

Secure Socket Layer (SSL) and Transport Layer Security (TLS) are cryptographic protocols used to secure communication over computer networks, especially the internet. They provide a layer of security by encrypting data transmissions and ensuring the integrity and authenticity of the exchanged information. Both SSL and TLS are cryptographic protocols designed to establish secure connections over computer networks. While SSL was the original protocol, TLS has become the industry standard for secure communication due to its improved security features and ongoing development. Users often encounter TLS in the form of "HTTPS" in web browser URLs, indicating that the website is using TLS to secure the connection and protect sensitive data [90].

Implementing SSL/TLS (Secure Sockets Layer/Transport Layer Security) protocols to secure web-based communication is a crucial practice in today's healthcare landscape, where the exchange of sensitive information between users and healthcare systems occurs continuously. These protocols establish a robust and encrypted communication channel, thereby fortifying the security of data transmission and safeguarding it against interception, unauthorized access, or tampering. SSL/TLS, with its layers of encryption and authentication mechanisms, offers a multi-faceted security approach that extends far beyond mere data encryption. One key aspect of SSL/TLS is its ability to ensure the authenticity of the websites and healthcare systems users interact with. Through the use of digital certificates issued by trusted Certificate Authorities (CAs), SSL/TLS verifies the legitimacy of the server, creating a secure foundation for data exchange. This trust model not only protects users from phishing attacks and man-in-the-middle attacks but also enhances their confidence in the privacy and security of their interactions with healthcare portals, online patient records, and telehealth platforms [91].

Limits

— Older SSL/TLS versions may have vulnerabilities that can be exploited by attackers.

— SSL/TLS relies on trusted certificate authorities (CAs). If a CA is compromised or issues a fraudulent certificate, it can lead to security breaches.

— SSL/TLS alone cannot prevent phishing attacks where users are tricked into visiting malicious sites with valid certificates.

— Some SSL/TLS configurations may not provide perfect forward secrecy, meaning that past traffic could be decrypted if private keys are compromised.

Figure 3.5 and 3.6 represents the mechanisms of Secure Socket Layer and Transport Layer Security.

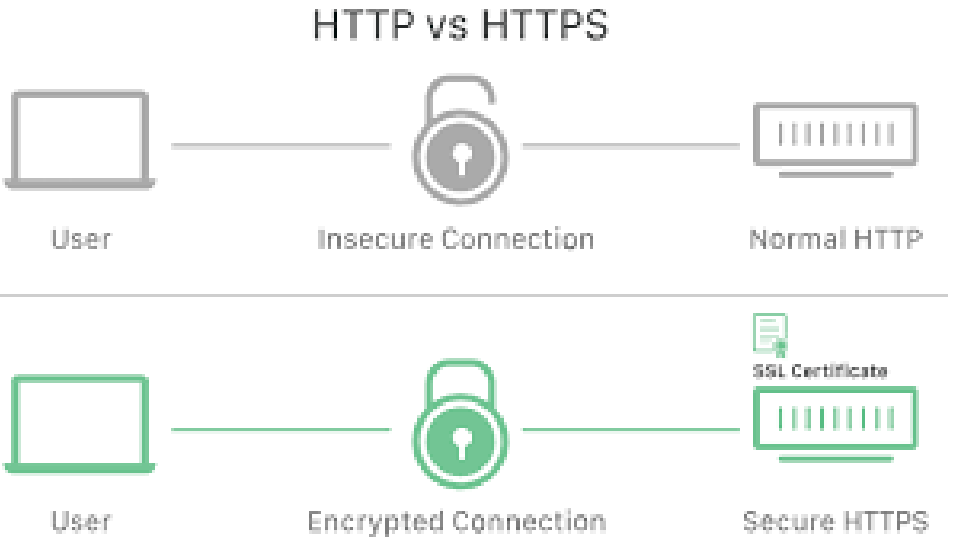


Figure 4.5 – Secure Socket Layer (SSL) [7]

4.3.4 Digital Signatures

A digital signature is a cryptographic technique used in the digital world to provide authentication, data integrity, and non-repudiation (meaning the signer cannot deny the authenticity

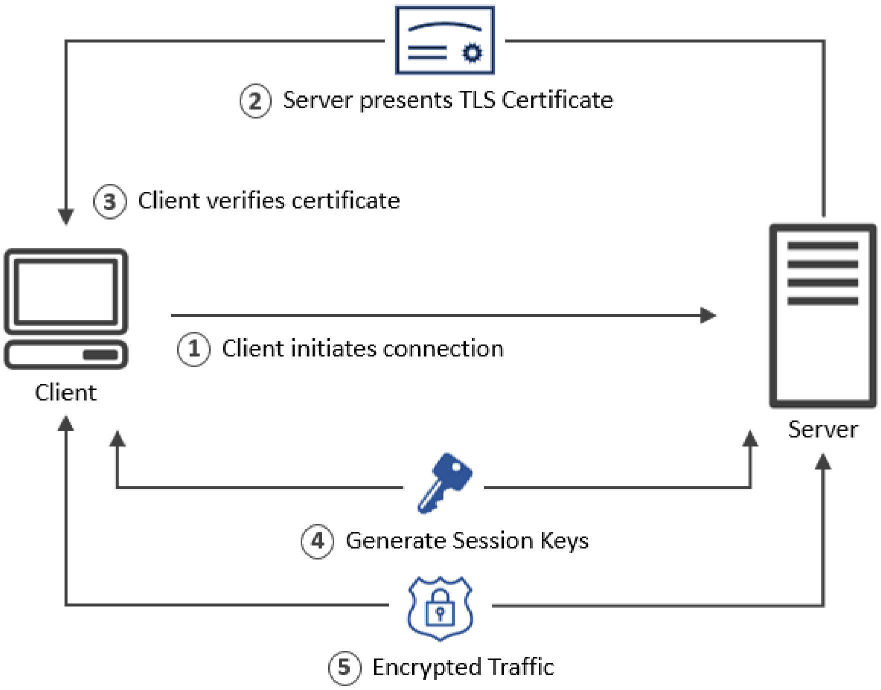

2 Server presents TLS Certificate
3 Client verifies certificate
Client
1 Client initiates connection
Server
4 Generate Session Keys
5 Encrypted Traffic

Figure 4.6 – Transport Layer Security (TLS) [8]

of their signature). It allows individuals or entities to sign electronic documents or messages to verify their identity and ensure that the content has not been altered since it was signed. Digital signatures rely on a key pair—comprising a private key, known only to the owner, and a public key, shared with others. When digitally signing a document or message, the owner uses their private key to generate a unique signature based on the content and the private key itself. Verification can be performed by anyone possessing the public key, ensuring the content's integrity and confirming the signature's authenticity. Additionally, digital signatures offer authentication, as trust in a specific public key implies trust in digital signatures generated with the corresponding private key, enhancing the overall security and reliability of digital transactions and communications [92].

Leveraging digital signatures for verifying the authenticity and integrity of electronic healthcare records is a pivotal component of modern healthcare information management systems. These sophisticated cryptographic techniques offer a comprehensive solution to the ever-growing challenge of safeguarding electronic health data. Digital signatures serve as a robust mechanism not only for ensuring data remains unaltered but also for establishing the legitimacy of healthcare records and their associated transactions. One of the key advantages of digital signatures lies in their ability to uniquely bind the identity of the sender or creator of a healthcare record to the document itself. This means that when a healthcare professional digitally signs a patient's medical history or a treatment plan, it not only confirms the document's integrity but also verifies the origin, allowing healthcare providers and patients to trust the source of the information [93].

Limits

— Digital signatures rely on the security of private keys. If a private key is compromised, it can lead to unauthorized signing and data manipulation.

- Trust in the certificate authority (CA) is essential. If the CA is compromised or issues fraudulent certificates, it can undermine trust in digital signatures.
- While digital signatures provide strong evidence of the signer's identity, there may still be challenges in proving non-repudiation in legal contexts.
- Vulnerabilities in software used for signing can be exploited by attackers, compromising the integrity of signatures.

Figure 3.7 represents the mechanism of Digital Signatures.

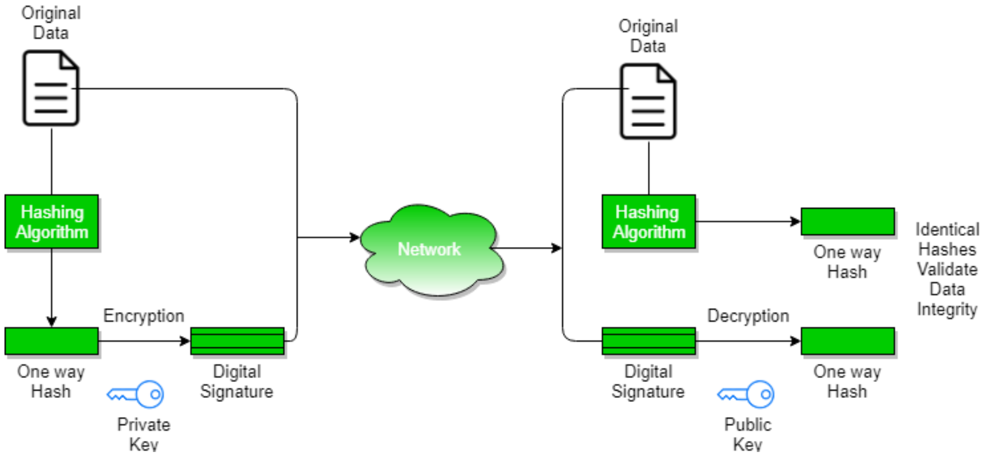


Figure 4.7 – Digital signature [9]

4.3.5 Secure Cloud Storage

Cloud storage refers to the online storage of data on remote servers hosted and managed by cloud service providers. Instead of storing data on a local hard drive or physical storage device, cloud storage enables users and organizations to store and access their files, documents, photos, videos, and other data over the internet [94].

Safeguarding the confidentiality, integrity, and availability of healthcare data in cloud storage solutions is of paramount importance in the digital age of healthcare information management. The responsibility lies not only in the adoption of cloud technology but also in ensuring that these cloud storage solutions meet and exceed stringent security standards and regulatory compliance requirements. An integral facet of this strategy involves the encryption of data stored in the cloud, a practice that serves as a potent defense against unauthorized access and data breaches. By encrypting healthcare data at rest in the cloud, organizations can ensure that even if a breach were to occur, the stolen information would remain indecipherable to malicious actors, significantly mitigating the impact of potential security incidents [95].

Limits

- Cloud storage providers have access to user data, raising privacy concerns and reliance on the provider's security measures.
- Cloud storage is still vulnerable to breaches, and if the provider's security is compromised, it can result in unauthorized access to stored data.
- Meeting industry-specific compliance regulations (e.g., HIPAA, GDPR) can be complex in the cloud, requiring careful configuration and management.

— Malicious or negligent actions by employees or cloud service provider staff can pose security risks.

4.3.6 Blockchain Technology

Blockchain is an innovative and transformative decentralized and distributed digital ledger technology that has revolutionized the way transactions are recorded and managed. It operates across a network of multiple computers, often referred to as nodes, where each node maintains a copy of the entire blockchain. This network structure is designed to guarantee the utmost security, transparency, and immutability of data. The term "blockchain" itself aptly describes its architecture, as it's composed of a series of individual transactions that are grouped together into blocks. These blocks are then linked together, forming a chronological chain of blocks that extends back to the very first transaction, creating an unbroken and tamper-resistant ledger. This inherent design ensures that once data is recorded on the blockchain, it becomes virtually impossible to alter, providing a robust foundation for a wide range of applications beyond its initial use in cryptocurrencies. Blockchain's decentralized nature, cryptographic security, and transparency make it a promising technology with far-reaching potential in industries such as finance, supply chain, healthcare, and more [96].

4.3.6.1 Blockchain Architecture

Figure 3.8 represents an overview of the typical elements found in a blockchain:

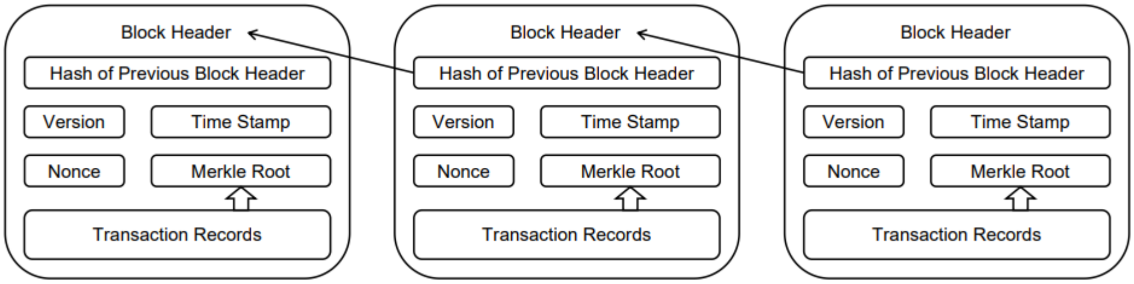


Figure 4.8 – The Chain Architecture of The Blockchain [10]

4.3.6.1.1 Block Header: A fundamental component within the architecture of a blockchain network is the block header, an integral part of distinguishing and identifying specific blocks within the broader blockchain. This block header undergoes a repetitive hashing process, generating the proof of work required for miners to earn rewards for their contributions to the network's security and transaction verification. The blockchain itself comprises an array of diverse block types, each serving as a container for recording transaction data, thereby ensuring the transparency and integrity of all network activities. These blocks, while inherently unique, are further distinguished by their individual block headers, with the block header hash playing a pivotal role in this identification process [97].

Within the block header resides a wealth of metadata, encompassing critical information that defines the block's attributes. Among these essential elements are the 4-byte blockchain version number, offering insights into the specific protocol version in use. The 32-byte representation of the previous block's hash links each block chronologically, contributing to the blockchain's immutability. Meanwhile, the 32-byte Merkle root facilitates efficient data verification by summarizing transaction details. The timestamp, another 4-byte component, records when the block was added to the blockchain, ensuring a chronological order of transactions. The difficulty target, also 4 bytes in length, signifies the level of complexity miners must overcome to add a new block to the chain, maintaining network security. Lastly, the 4-byte miner nonce, an essential component, introduces an element of randomness into the block header, influencing the outcome of the proof of work process. In aggregate, this comprehensive 80-byte block header encapsulates vital data and metadata, enabling the blockchain to operate as an unalterable ledger of transactions while preserving the security and integrity of the network [97].

4.3.6.1.2 Hash of the previous block header: The hash of the previous block header plays a pivotal role in the security and immutability of the blockchain. This hash, derived through cryptographic algorithms, serves as a digital fingerprint of the preceding block's entire content, encapsulating all its data and metadata. This cleverly designed linkage mechanism creates a continuous and unbroken chain of blocks, with each block securely bound to its predecessor. The result is a tamper-resistant structure where any alteration to the data within a single block would necessitate changes to all subsequent blocks, making the blockchain highly resistant to unauthorized modifications [97].

4.3.6.1.3 Version: A version number within the context of a blockchain serves as a vital reference point for participants in the network, facilitating the understanding of protocol-wide modifications and updates. As blockchain ecosystems continually evolve and adapt to meet the demands of various industries and applications, the version number becomes a beacon of clarity amidst the ever-changing landscape. It allows developers, users, and stakeholders to identify which iteration of the blockchain protocol is in use, enabling them to assess compatibility, understand the features and improvements introduced in newer versions, and make informed decisions about network participation [98].

4.3.6.1.4 TimeStamp: The timestamp in a blockchain's block structure is a succinct yet indispensable piece of data that is meticulously serialized and preserved within each individual block. Its core function extends far beyond the mere recording of chronological events; rather, it serves as an invaluable tool for precisely marking the moment at which the block underwent the complex process of mining and validation within the network. By embedding this timestamp, the blockchain not only offers a historical record of events but also establishes a secure and auditable timeline of transactions and block creation. This temporal dimension plays a pivotal role in maintaining the transparency and accountability of the blockchain, enabling participants to track the sequence of activities and ensuring the integrity of data over time [98].

4.3.6.1.5 Difficulty target: The difficulty target is a crucial parameter within the blockchain ecosystem, serving as a dynamic mechanism that adjusts the level of complexity miners face when undertaking the intricate task of solving a block. This dynamic adjustment of difficulty is a fundamental feature of blockchain technology, carefully calibrated to ensure the network's stability and security. It functions by regulating the computational effort required to validate transactions and add a new block to the chain. Miners must continually adapt to this ever-changing difficulty target, making blockchain a resilient and self-regulating system. By increasing or decreasing the difficulty based on network conditions, the blockchain remains robust, thwarting attempts to compromise its security and ensuring the equitable participation of miners in the validation process[97].

4.3.6.1.6 Nonce: A nonce serves as a critical piece of the puzzle, integral to the process of securing and validating transactions within the blockchain. Miners embark on a computational quest to discover this nonce, and the miner who successfully unveils the answer reaps the well-deserved rewards, typically in the form of the block reward. This block reward is a testament to the miner's dedication and computational prowess, marking the culmination of their efforts as they contribute to the blockchain's stability and security [98].

4.3.6.2 Blockchain Concepts

Blockchain is a complex and multifaceted technology that encompasses various key concepts and principles. Here are some fundamental blockchain concepts:

4.3.6.2.1 Blockchain technology Within the captivating realm of blockchain technology, every transaction, whether it has already transpired or is poised to occur imminently, is meticulously documented within the public ledger known as the blockchain. This historical repository of transactions is aptly named a "blockchain" because it takes the form of an unbroken and ever-growing chain of individual blocks. The unceasing expansion of this ledger is made possible by the diligent efforts of miners, who dedicate computational resources to solve intricate mathematical puzzles, thereby securing and validating transactions. As a testament to the blockchain's chronological integrity, each newly mined block seamlessly integrates into the existing chain, following a predetermined order. This methodical and linear addition of blocks ensures the blockchain's continuity, providing a robust and immutable foundation for recording and preserving a comprehensive history of transactions, a feature that has far-reaching implications across diverse applications and industries [99].

4.3.6.2.2 Miners A Miner is an individual equipped with the requisite computer hardware and specialized software, primed for the validation of fresh blockchain transactions prior to their incorporation into the ever-expanding blockchain ledger, assumes the esteemed role of a "blockchain miner." These miners, often comprising a network of participants, engage in a dynamic and competitive endeavor that relies on the application of a cryptographic hash method. This method is the cornerstone of their operation, serving as the key to unlocking the most optimal solution to a complex mathematical puzzle that stands as a gatekeeper to the validation process. The competition among miners for this solution is intense, demanding not only computational power but also an intricate understanding of the cryptographic principles at play. As they vie to decipher this puzzle, miners contribute to the blockchain's security, integrity, and transparency [100].

4.3.6.2.3 Consensus Miners play a critical role in blockchain networks, using "consensus" mechanisms to add new data records. In public blockchains like Bitcoin, Proof-of-Work (PoW) is a well-known consensus method. PoW involves solving complex mathematical puzzles in a process referred to as "mining." The first miner to solve the puzzle can add a block to the chain and receive a Bitcoin reward. Transactions are stored across the decentralized network of miners, and the consensus algorithm ensures transaction verification and confirmation. This collaborative process underpins blockchain's security and transparency, enabling various applications beyond cryptocurrencies [10].

4.3.6.2.4 Smart Contracts A smart contract is a self-executing contract with its terms and conditions directly encoded in code. These contracts automatically execute when predefined conditions are met, running on decentralized blockchain networks, which ensures transparency, security, and immutability. They eliminate the need for intermediaries, are transparent and verifiable by all network participants, and once deployed, are tamper-resistant. Smart contracts find applications beyond financial transactions, being used in supply chain management, legal agreements, healthcare, voting systems, and more. Ethereum is a prominent platform for creating and executing smart contracts, revolutionizing automation, trust, and efficiency across various industries [99].

4.3.6.3 Types of Blockchain

There are 3 main types of Blockchain:

4.3.6.3.1 Public Blockchain Public blockchains represent revolutionary decentralized networks that are accessible to all, fostering an extraordinary level of inclusiveness and transparency. These blockchain networks, epitomized by cryptocurrencies such as Bitcoin and Ethereum, extend an invitation to anyone to partake in their ecosystem, facilitating the validation of transactions and unfettered access to the entire ledger. The absence of a central authority overseeing access, a hallmark of their permissionless nature, paves the way for trustless interactions and establishes a foundation of robust security through decentralization [101].

4.3.6.3.2 Private Blockchain Private blockchains are different from public ones. They're designed for specific groups or organizations and offer a lot of control and privacy. Only certain people or organizations can join these networks, and there's usually a central authority or a group that manages access. Private blockchains are often used for things like managing supplies or sharing sensitive information among trusted partners. While they're not as decentralized as public blockchains, this controlled setup helps them follow rules and customize solutions for specific business needs. Private blockchains are a secure and efficient way to streamline processes within closed groups or organizations [101].

4.3.6.3.3 Consortium Blockchain Combining decentralization and limited access, consortium blockchains stand as a compromise between public and private blockchains. They are controlled by a number of organizations that work together to uphold the blockchain network, frequently from within a single industry or sector. Consortium blockchains strive to achieve a balance between openness and governance, making them suited for situations where several entities must communicate information while trust is divided among a set group. While retaining some degree of control over network operations, this kind of blockchain can increase transparency and efficiency. In fields like supply chain tracking, where several stakeholders require real-time insight into shared data, consortium blockchains are being used more frequently [101].

4.4 File Sharing methods Through Blockchain

File sharing methods through blockchain leverage the unique capabilities of blockchain technology to enhance the security, transparency, and efficiency of sharing files and data. Here are some key approaches to file sharing through blockchain:

4.4.1 Decentralized File Sharing Platforms

Decentralized file sharing platforms enable users to share files directly with one another without relying on centralized intermediaries like cloud storage providers. These platforms use distributed networks of nodes to store and manage files, ensuring data availability and redundancy. Users can access and share files securely, with the blockchain providing transparency and auditability. Examples include Filecoin and InterPlanetary File System (IPFS) [102].

4.4.2 Smart Contracts for File Sharing

Smart contracts play a pivotal role in automating and enforcing file sharing agreements on the blockchain. Users can create contracts that specify the terms of file sharing, such as access conditions and payment details. When these conditions are met, the smart contract automatically executes the file transfer and any associated payments. This approach ensures trust and eliminates the need for intermediaries in file sharing arrangements [103].

4.4.3 Blockchain-based Content Platforms

Blockchain-based content platforms are emerging as alternatives to traditional content-sharing websites. Content creators can upload their work to blockchain-based platforms, where they retain control and ownership of their content. Blockchain technology helps in tracking content usage, ensuring fair compensation through microtransactions, and protecting against unauthorized copying or distribution [104].

4.5 Blockchain and healthcare

Within healthcare systems, blockchain networks serve as a transformative force, revolutionizing the storage and exchange of vital patient data among an extensive network of stakeholders that encompass hospitals, diagnostic laboratories, pharmaceutical companies, and healthcare professionals. The applications of blockchain technology in this context extend far beyond mere data management; they possess the remarkable capacity to pinpoint critical errors, including those with life-threatening implications, within the intricate tapestry of the medical industry. This transformative potential translates into a multifaceted enhancement of the healthcare ecosystem, marked by elevated levels of efficiency, security, and transparency in the exchange of sensitive medical information across various sectors. By harnessing the power of blockchain, healthcare institutions can embark on a journey of knowledge acquisition and data-driven insights, fostering advancements in the analysis and treatment of patients' health conditions. This technological innovation not only propels the healthcare industry forward but also holds the promise of improved patient outcomes and a more responsive and interconnected healthcare landscape [105].

4.6 Conclusion

In this chapter, we've conducted a comprehensive overview of the significant privacy risks associated with medical data, delving into the potential consequences of these risks. We've explored various solutions proposed over the years to address these privacy threats, culminating in the transformative role of blockchain technology. Our investigation has revealed the remarkable ability of blockchain to surmount the limitations inherent in prior solutions, emerging as a powerful safeguard for medical data privacy.

The next chapter will take a closer look at our method, emphasizing the important connection between advanced deep learning models and blockchain technology. We'll explain how our approach, which relies on these advanced tools, effectively safeguards healthcare data from privacy breaches and vulnerabilities. This doesn't just enhance data protection; it also improves the accuracy of medical diagnoses.

Chapter 5

The proposed method and the implementation

5.1 Introduction

This chapter provides a thorough explanation of our research approach, focusing on two main aspects: our deep learning models and the security features provided by blockchain technology. We'll break down our approach step by step better understanding. First, we'll explain the deep learning models we used, including how they're designed, how we trained them, and how we prepared the data for them. Then, we'll dive into the world of blockchain technology and its security benefits. We'll talk about how blockchain uses encryption and smart contracts to control who can access data. Also we will talk about the Multi-Diagnosis application.

5.2 Proposed method

In the healthcare field, ensuring the security of data and protecting people's privacy are top priorities when developing deep learning models for medical diagnosis. The risks associated with data leaks or breaches can have serious consequences for patients and healthcare providers. To tackle this issue, we propose a new method that relies on Blockchain technology, a secure way to share locally trained deep learning models without exposing sensitive patient data.

Our approach involves using Blockchain to create a safe and decentralized network for sharing these locally trained models. This means that private patient information remains confidential and is not accessible to unauthorized parties.

In addition to enhancing security, our method improves the overall performance of the global deep learning model through a technique called model ensembling. By combining the knowledge from locally trained models, we can create a more accurate diagnostic model that is less prone to making mistakes. This is especially crucial in medical diagnosis, where even small errors can have serious consequences for patients.

In summary, our proposed method provides a unique solution to the challenge of developing deep learning models for medical diagnosis while safeguarding the security and privacy of patient data. By utilizing Blockchain technology and model ensembling, we achieve better performance and stronger data protection, ultimately benefiting both patients and healthcare providers.

An overview of the proposed method is represented in Figure 5.1.

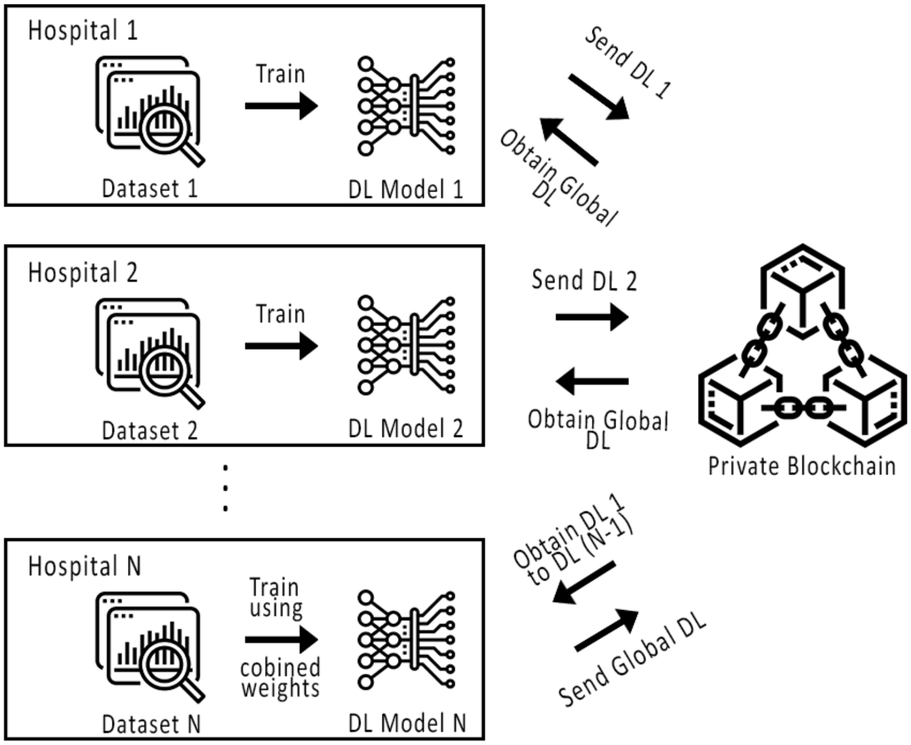


Figure 5.1 – Overview of our method

As we can see in Figure 5.1, each collaborator hospital trains a local deep learning model using its private data then shares it using blockchain except for the last hospital that is going to build the global model, the last hospital obtains the local models of the other hospitals from the blockchain, extracts their final weights, and uses them to train the global model on its private data. In the end, this last shares the global model back to the previous Hospitals.

5.3 Implementation

Suppose that there are three hospitals, each with a substantial amount of patient data that could potentially be combined used to create a powerful deep learning model. However, due to the sensitive nature of patient data, it's essential to guarantee the privacy and security of these data during the development process. To tackle this challenge, a solution is to construct local deep learning models within the first two hospitals using their own private data. Then, these models can be shared with the third hospital via blockchain technology which (The third hospital) creates the the final global model. In this case, we focused on three different diseases: Breast cancer, Lung cancer, and Diabetes, and each disease has two local models developed within Hospital 1 and Hospital 2. Blockchain technology provides a secure and unchangeable platform that enables data sharing while upholding privacy, security, and data integrity. By utilizing blockchain, hospitals can safely exchange their locally created models without violating patient data privacy, and they can be certain that the deep learning models remain unaltered.

Hospital 3 retrieves the final weights of both model 1 and model 2 for each disease, merge these weights, and employ them to train a final global model using their private data. Through the utilization of locally trained models from various hospitals, this overarching model can acquire knowledge from a broader spectrum of data, which enhances its accuracy and precision. This process is referred to as model ensembling.

In the end, Hospital 3 develops a multi-diagnosis application by utilizing these global models.

5.3.1 Datasets

To validate our method we used 3 different datasets.

5.3.1.1 Breast Cancer Wisconsin (Diagnostic) DataSet

The Breast Cancer Wisconsin (Diagnostic) dataset [106, 107] holds significant importance in the deep learning field, particularly in the development of models related to breast cancer diagnosis. This dataset, which we examined during our evaluation, is accessible via the Kaggle repository and was sourced from UCI Machine Learning. It contains 569 instances, each including an ID number, 30 features, a binary class denoting whether the cancer is malignant or benign, and an empty attribute, summing up to a total of 33 attributes.

Data for this dataset was procured by analyzing fluid samples from patients with solid breast cancer, employing Xcyt, a user-friendly and effective graphic computer software program. Xcyt computed 10 features for each cell sample using a curve-fitting technique. These features were subsequently employed to calculate the mean value, extreme value, and standard error for each characteristic, resulting in a 30-dimensional real-valued vector for each image in the dataset. This dataset plays a crucial role in the enhancement of breast cancer diagnosis accuracy by furnishing a substantial and varied pool of data for training and testing models.

The comprehensive set of features and binary class labels within the Breast Cancer Wisconsin (Diagnostic) dataset renders it an indispensable resource for deep learning researchers and practitioners dedicated to advancing the domain of cancer diagnosis.

5.3.1.2 Lung Cancer Prediction Dataset

The Lung Cancer Data, accessible on data.world [108] and Kaggle [109], originates from the Cancer Data Health Program (CDHP), an organization dedicated to granting access to publicly available cancer data. CDHP gathers data from diverse sources, including the National Cancer Institute's Surveillance, Epidemiology, and End Results (SEER) program, and offers it for research purposes. This database is a collection of 1000 instances with 23 attributes, furnishing insights into symptoms, risk factors, and potential lung cancer diagnoses.

For medical researchers and practitioners, this database stands as an invaluable resource in comprehending and addressing this disease. The attributes contained within the database encompass a wide spectrum of factors that could influence an individual's susceptibility to lung cancer. These encompass demographic particulars like age, gender, and smoking history. A notable feature of the database is the classification of risk levels into categories of Low, Medium, and High. This categorization provides crucial information about the gravity of an individual's lung cancer risk, aiding medical professionals in formulating targeted interventions and treatment strategies.

In addition to the risk level categories, each entry in the database carries a unique index number and patient ID. Altogether, this database emerges as an indispensable tool, enriching medical research and practice by offering comprehensive insights into the factors contributing to lung cancer risk and facilitating the development of effective prevention and treatment approaches against this devastating ailment.

5.3.1.3 Diabetes UCI Dataset

The dataset for early-stage diabetes risk prediction holds substantial value for both researchers and healthcare professionals dedicated to enhancing the early identification and prevention of diabetes. This dataset encompasses clinical and demographic information from 520 individuals who underwent diabetes screening at Sylhet Diabetes Hospital in Bangladesh.

Within this dataset, 16 variables offer crucial insights, including the patient's age, gender, and diabetes-related symptoms, all of which serve as pivotal indicators for assessing the risk of diabetes. This dataset's wealth of information provides a foundation for constructing machine learning models capable of accurately predicting the risk of early-stage diabetes.

Identifying individuals at risk of developing diabetes in its early stages empowers healthcare practitioners to initiate preventive measures, thereby reducing the likelihood of complications and enhancing overall patient well-being. Furthermore, this dataset presents a valuable resource for researchers investigating novel diagnostic methods and treatments for diabetes.

It's important to note that this dataset is publicly accessible through the UCI Machine Learning Repository [110] and Kaggle [111], and it has gained widespread utilization in research endeavors focused on diabetes risk prediction. This underscores its pivotal role as an invaluable asset within the scientific community.

5.3.2 Preparation of the Datasets

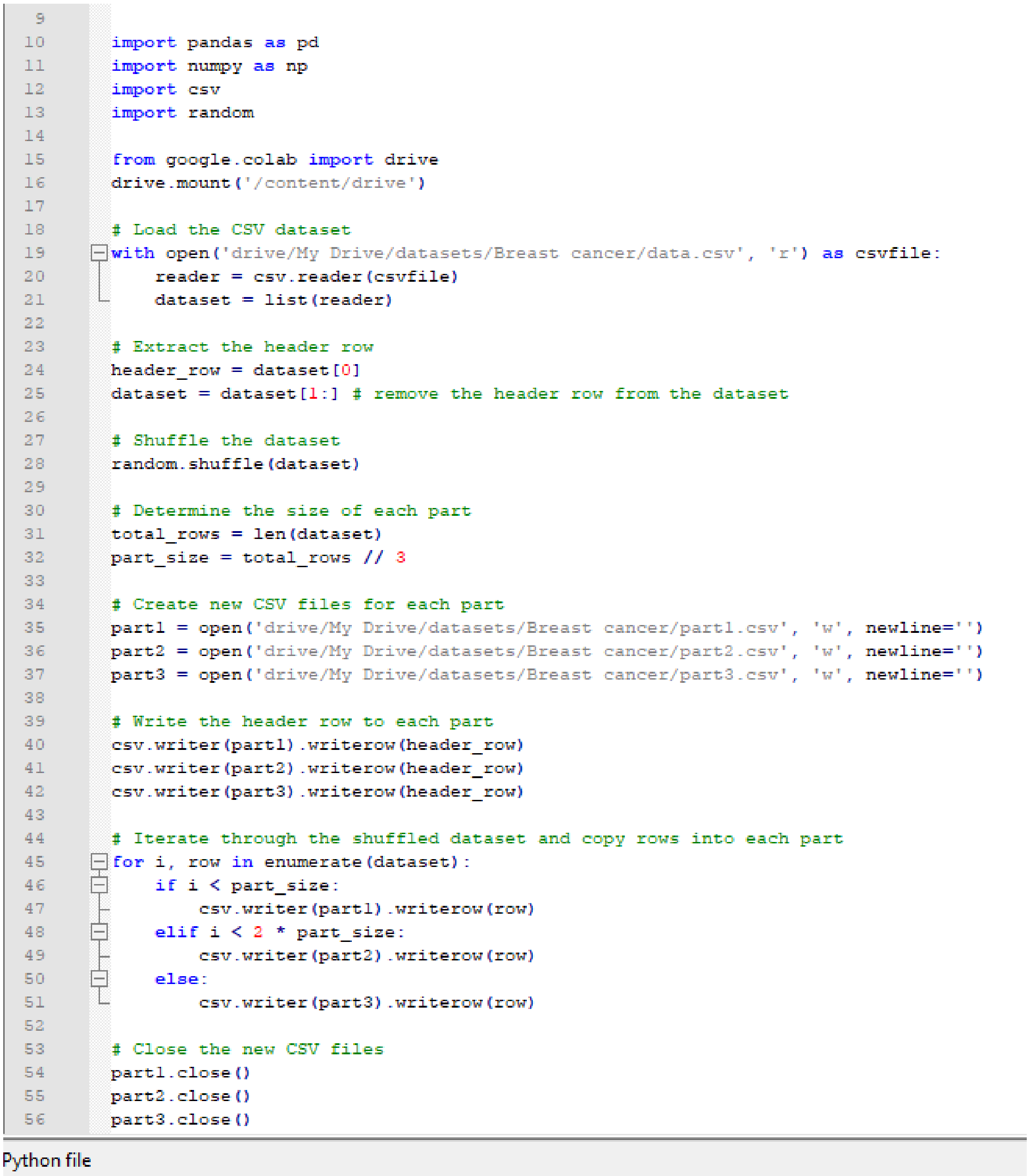
We consider a scenario involving three hospitals, and our objective is to implement the procedure depicted in Figure 5.1. To accomplish this, it becomes essential to partition the dataset into three separate datasets, all sharing an identical structure. The subsequent actions are applied to each of these three datasets.

We initiate the process by loading the CSV file 'data.csv' into the dataset variable, utilizing the csv.reader() function. Subsequently, we extract and remove the header row from the dataset variable. To introduce randomness into the dataset, we apply the random.shuffle() function, effectively shuffling the order of the rows. Determining the size of each portion is achieved by dividing the total row count by 3 and storing this result in the part_size variable.

Following this, we generate three distinct CSV files for each respective part using the open() function and incorporate the header row into each file via the csv.writer() function. For the purpose of partitioning the dataset into these three parts, we employ a for loop that iterates through the shuffled dataset. Each row is directed to the appropriate part's CSV file based on its index, which is accomplished using the csv.writer() function. Consequently, the initial part receives the first part_size rows, the subsequent part receives the subsequent part_size rows, and the final part accommodates the remaining rows.

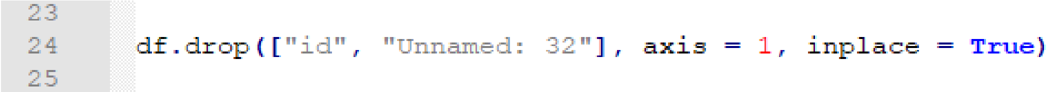
In conclusion, the provided code concludes this process by closing the three newly created CSV files, freeing up system resources. As a result, we now have three distinct CSV datasets, all possessing identical structures for each disease category. Table 5.1 furnishes information on instance counts and characteristics for each dataset section following the division process. Figure 5.2 represents the code for splitting data.

Clearly, Parts 1 and 2 encompass 189 instances along with 30 features, whereas Part 3 includes 191 instances, resulting in a total of 569 instances in the Breast cancer dataset. Notably, we have excluded the ID and Unnamed features from all datasets as they are deemed unnecessary (Code in Figure 5.3). Additionally, we have undertaken a transformation of the "diagnosis"



```python
import pandas as pd
import numpy as np
import csv
import random

from google.colab import drive
drive.mount('/content/drive')

# Load the CSV dataset
with open('drive/My Drive/datasets/Breast cancer/data.csv', 'r') as csvfile:
    reader = csv.reader(csvfile)
    dataset = list(reader)

# Extract the header row
header_row = dataset[0]
dataset = dataset[1:] # remove the header row from the dataset

# Shuffle the dataset
random.shuffle(dataset)

# Determine the size of each part
total_rows = len(dataset)
part_size = total_rows // 3

# Create new CSV files for each part
part1 = open('drive/My Drive/datasets/Breast cancer/part1.csv', 'w', newline='')
part2 = open('drive/My Drive/datasets/Breast cancer/part2.csv', 'w', newline='')
part3 = open('drive/My Drive/datasets/Breast cancer/part3.csv', 'w', newline='')

# Write the header row to each part
csv.writer(part1).writerow(header_row)
csv.writer(part2).writerow(header_row)
csv.writer(part3).writerow(header_row)

# Iterate through the shuffled dataset and copy rows into each part
for i, row in enumerate(dataset):
    if i < part_size:
        csv.writer(part1).writerow(row)
    elif i < 2 * part_size:
        csv.writer(part2).writerow(row)
    else:
        csv.writer(part3).writerow(row)

# Close the new CSV files
part1.close()
part2.close()
part3.close()
```

Figure 5.2 – Splitting Data Code



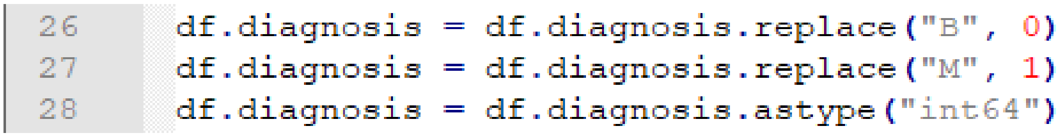
```python
df.drop(["id", "Unnamed: 32"], axis = 1, inplace = True)
```

Figure 5.3 – Dropping unnecessary features Code

Table 5.1 – Information on Different Parts of the Datasets for Three Diseases

Disease	Dataset Parts	Instances	Features
Breast Cancer	Part 1	189	30
	Part 2	189	30
	Part 3	191	30
Lung Cancer	Part 1	333	23
	Part 2	333	23
	Part 3	334	23
Diabetes	Part 1	173	16
	Part 2	173	16
	Part 3	174	16

column values, which represent the class, by converting "B" and "M" to binary values of 0 and 1, respectively (Conversion Code in Figure 5.4).



```
df.diagnosis = df.diagnosis.replace("B", 0)
df.diagnosis = df.diagnosis.replace("M", 1)
df.diagnosis = df.diagnosis.astype("int64")
```

Figure 5.4 – Converting Class Values to Binary Values Code

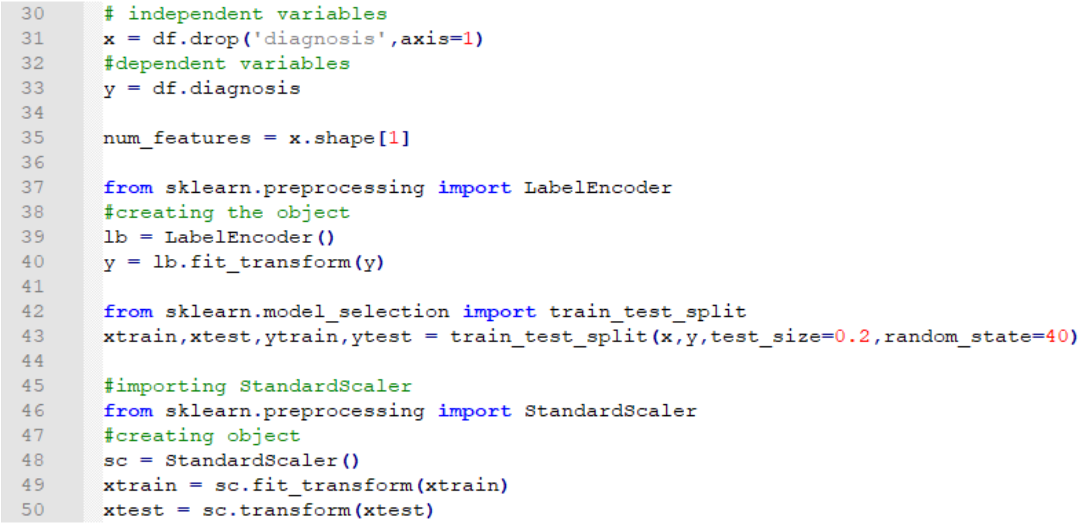
As for the Lung cancer dataset, Parts 1 and 2 consist of 333 instances and 23 features, while Part 3 consists of 334 instances, summing up to a total of 1000 instances. Similarly, we have removed the Patient ID and Index features as they do not hold relevance for any of the datasets. Moreover, the "Level" column values, signifying the class, have been converted into integer values of 0, 1, and 2, corresponding to "Low," "Medium," and "High."

Furthermore, within the Diabetes dataset, Parts 1 and 2 are comprised of 173 instances and 16 features, whereas Part 3 encompasses 174 instances, resulting in a total of 520 instances. Here, we have transformed all the features values from "Yes" and "No" into binary values of 1 and 0, respectively.

Following these data preparations, each dataset segment is segregated into two distinct groups: one group contains the independent variables (referred to as X), while the other group contains the dependent variable (referred to as Y). Subsequently, the dataset is divided into two subsets, one for training and the other for testing, a process facilitated by the use of the train_test_split function. Finally, the data undergoes standardization, accomplished by utilizing the StandardScaler object from scikit-learn. This scaling procedure ensures that the dataset's features exhibit an average of 0 and a variance of 1 (The code is in Figure 5.5).

5.3.3 Deep Learning Models

An Artificial Neural Network (ANN) is a machine learning model inspired by the functioning of the human brain. It comprises interconnected nodes that receive, process, and generate output data. The training of an ANN involves a process known as backpropagation, which fine-tunes the connection weights between nodes in the network. ANN has found effective applications in diverse fields including image recognition, natural language processing, speech recognition,



```
# independent variables
x = df.drop('diagnosis',axis=1)
#dependent variables
y = df.diagnosis

num_features = x.shape[1]

from sklearn.preprocessing import LabelEncoder
#creating the object
lb = LabelEncoder()
y = lb.fit_transform(y)

from sklearn.model_selection import train_test_split
xtrain,xtest,ytrain,ytest = train_test_split(x,y,test_size=0.2,random_state=40)

#importing StandardScaler
from sklearn.preprocessing import StandardScaler
#creating object
sc = StandardScaler()
xtrain = sc.fit_transform(xtrain)
xtest = sc.transform(xtest)
```

Figure 5.5 – Train test split + Standardization Code

and financial forecasting. Numerous renowned scientists and researchers have contributed to its development, and ongoing research efforts are dedicated to advancing this technology [112].

In our research, we constructed two variations of Artificial Neural Networks (ANNs). One was designed for binary classification, applied in cases involving Breast Cancer and Diabetes Classification. The second type was tailored for categorical classification, specifically for Lung Cancer, where it categorizes data into three distinct classes.

5.3.3.1 Binary Classification ANN (Breast Cancer and Diabetes)

The Artificial Neural Network (ANN) model was constructed using the Keras library's Sequential() function. This model initialization involves creating an empty sequence of layers. Subsequently, two dense layers were incorporated into the model using the add() function. The first dense layer comprises 30 neurons utilizing the Rectified Linear Unit (ReLU) activation function, and the input layer is automatically included with its dimensions specified by the variable 'number_features'. Following this, a BatchNormalization layer was added post the initial dense layer. Batch normalization is a technique that enhances the training of deep neural networks by standardizing the inputs in each batch to have zero mean and unit variance, contributing to the reduction of overfitting and improved generalization.

Additionally, a Dropout regularization layer was introduced after BatchNormalization with a rate of 0.5. This Dropout layer randomly sets 50% of the neuron outputs to zero during training, which aids in mitigating overfitting. The second dense layer was added similarly with 30 neurons, accompanied by the inclusion of another BatchNormalization and Dropout layers.

Finally, an output layer was appended, featuring a single neuron and sigmoid activation function tailored for binary classification. The model was then compiled, utilizing the Adam optimizer and binary cross-entropy loss function. An accuracy metric was also defined to monitor the model's performance during the training process.

To safeguard against overfitting, callbacks were configured to monitor the validation loss during training. Specifically, training would halt if the validation loss did not decrease for 25 consecutive epochs.

This model underwent training using the fit() function, with a batch size set to 32 and a total of 100 epochs. The previously defined callbacks were passed to the fit() function to oversee the model's performance during training. For the Code please refer to Figure 5.6.

The mathematical explanation is as follows:

- Notation: Let us define the following notation used in the equations:

— x: the input vector of size *number_features*

— $W^{(i)}$: the weight matrix of layer i

— $b^{(i)}$: the bias vector of layer i

— $z^{(i)}$: the linear combination of the inputs and weights of layer i

— $a^{(i)}$: the output (activation) vector of layer i

— $p^{(i)}$: the dropout probability of layer i

— $BN^{(i)}$: the batch normalization transformation of layer i

— y_{pred}: the predicted output of the network

- Equations: The equations for each layer are as follows:

Input layer:

There is no equation for the input layer, as it just passes the input vector x to the first hidden layer.

First hidden layer:

$$\begin{aligned} z^{(1)} &= W^{(1)}x + b^{(1)} \\ a^{(1)} &= \text{ReLU}(z^{(1)}) \\ a^{(1)'} &= BN^{(1)}(a^{(1)}) \\ a^{(1)''} &= \text{Dropout}(a^{(1)'}, p^{(1)}) \end{aligned}$$

where $\text{ReLU}(z) = \max(0, z)$ is the rectified linear unit activation function, $BN(a)$ is the batch normalization transformation, and $\text{Dropout}(a, p)$ randomly drops out a fraction p of the activations a during training to prevent overfitting.

Second hidden layer:

$$\begin{aligned} z^{(2)} &= W^{(2)}a^{(1)''} + b^{(2)} \\ a^{(2)} &= \text{ReLU}(z^{(2)}) \\ a^{(2)'} &= BN^{(2)}(a^{(2)}) \\ a^{(2)''} &= \text{Dropout}(a^{(2)'}, p^{(2)}) \end{aligned}$$

Output layer:

$$y_{pred} = \sigma(W^{(3)}a^{(2)''} + b^{(3)})$$

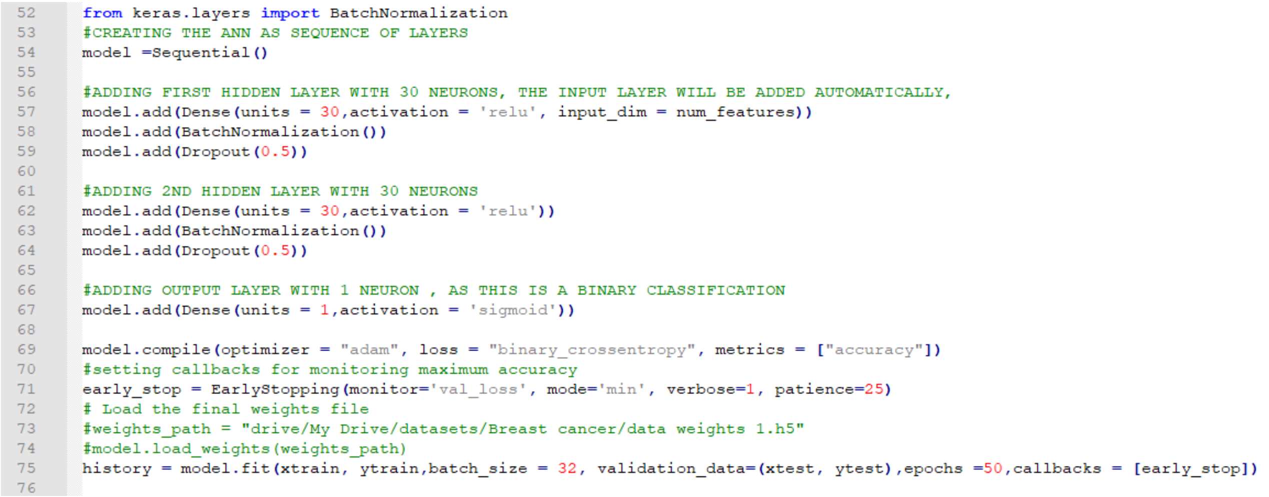
where $\sigma(z) = \frac{1}{1+e^{-z}}$ is the sigmoid activation function.

During training, the network is trained to minimize the binary cross-entropy loss function $L(y, y_{pred})$ using the Adam optimizer. The loss function is defined as:

$$L(y, y_{pred}) = -[y \log(y_{pred}) + (1 - y) \log(1 - y_{pred})]$$

where y is the true label (0 or 1), and y_{pred} is the predicted output of the network.

The accuracy metric is calculated as the percentage of correctly classified examples in the validation set.



```
from keras.layers import BatchNormalization
#CREATING THE ANN AS SEQUENCE OF LAYERS
model =Sequential()

#ADDING FIRST HIDDEN LAYER WITH 30 NEURONS, THE INPUT LAYER WILL BE ADDED AUTOMATICALLY,
model.add(Dense(units = 30,activation = 'relu', input_dim = num_features))
model.add(BatchNormalization())
model.add(Dropout(0.5))

#ADDING 2ND HIDDEN LAYER WITH 30 NEURONS
model.add(Dense(units = 30,activation = 'relu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))

#ADDING OUTPUT LAYER WITH 1 NEURON , AS THIS IS A BINARY CLASSIFICATION
model.add(Dense(units = 1,activation = 'sigmoid'))

model.compile(optimizer = "adam", loss = "binary_crossentropy", metrics = ["accuracy"])
#setting callbacks for monitoring maximum accuracy
early_stop = EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=25)
# Load the final weights file
#weights_path = "drive/My Drive/datasets/Breast cancer/data weights 1.h5"
#model.load_weights(weights_path)
history = model.fit(xtrain, ytrain,batch_size = 32, validation_data=(xtest, ytest),epochs =50,callbacks = [early_stop])
```

Figure 5.6 – Binary Classification Model Code

5.3.3.2 Categorical Classification ANN (Lung Cancer case)

The construction of the Artificial Neural Network (ANN) model entails employing the Sequential() function to establish an initial empty layer sequence that will constitute the model's architecture. This architecture takes shape by adding two dense layers to the model through the use of the add() function.

The first dense layer is comprised of 30 neurons, employing the Rectified Linear Unit (ReLU) activation function, and the input dimension is automatically integrated, determined by the 'number_features' variable. To enhance model robustness, a BatchNormalization layer is introduced following the initial dense layer. This layer serves to normalize the inputs, effectively reducing overfitting and promoting better generalization. To further combat overfitting, a Dropout regularization layer is implemented post-BatchNormalization, with a 0.5 rate. During training, this Dropout layer randomly deactivates 50% of neuron outputs.

The second dense layer follows the same pattern, featuring 30 neurons, with additional BatchNormalization and Dropout layers incorporated afterward. For multi-class classification, an output layer housing 3 neurons with a softmax activation function is appended.

Upon defining the model's architecture, it is compiled using the Adam optimizer and categorical cross-entropy loss function. Additionally, an accuracy metric is specified to track the model's performance during training.

To prevent overfitting, the model is equipped with callbacks that monitor validation loss during training. If the validation loss remains unchanged for 10 consecutive epochs, training is halted as a precautionary measure against overfitting.

Subsequently, model training is executed using the fit() function, with a batch size set to 32 and a total of 37 epochs. The previously configured callbacks are provided to the fit() function for performance monitoring. The training history is stored within the 'history' object. For the Code please refer to Figure 5.7.

The mathematical explanation is as follows:
- Notation:
Let us define the following notation used in the equations:
— $num\_classes$: the number of output classes
The equations for layer 1 and 2 are the same as the binary classification.
The output layer is as follows:
Output layer:

$$y_{pred} = softmax(W^{(3)}a^{(2)''} + b^{(3)})$$

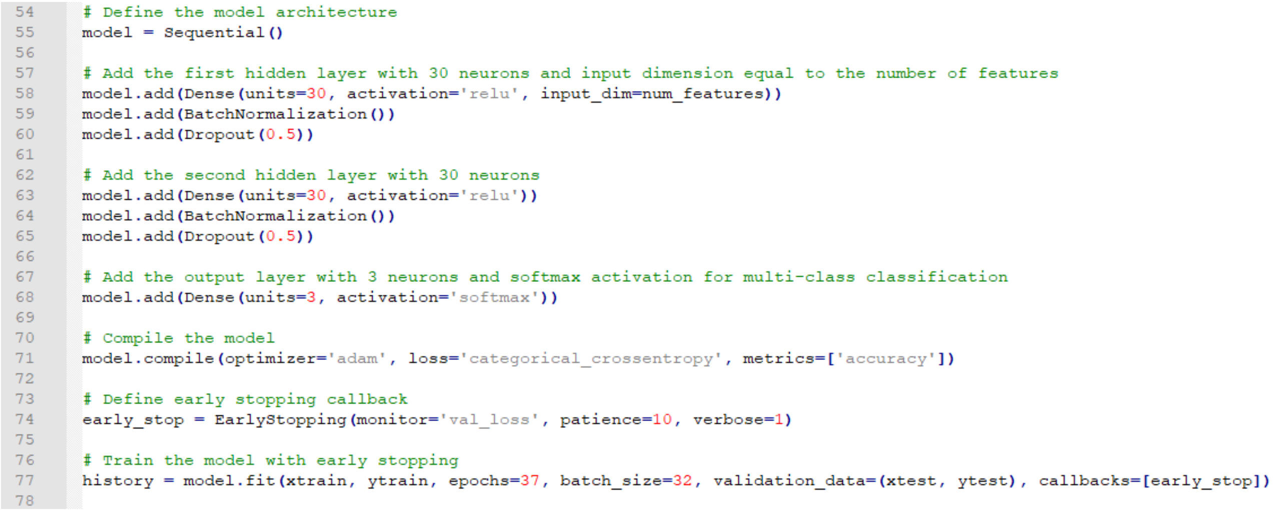
where $softmax(z) = \frac{e^z}{\sum_{i=1}^{num\_classes} e^{z_i}}$ is the softmax activation function for multi-class classification.

During training, the network is trained to minimize the categorical cross-entropy loss function $L(y, y_{pred})$ using the Adam optimizer. The loss function is defined as:

$$L(y, y_{pred}) = -\sum_{i=1}^{num\_classes} y_i \log(y_{pred_i})$$

where y is the true label (one-hot encoded vector) and y_{pred} is the predicted output of the network.

The accuracy metric is calculated as the percentage of correctly classified examples in the validation set.



```
# Define the model architecture
model = Sequential()

# Add the first hidden layer with 30 neurons and input dimension equal to the number of features
model.add(Dense(units=30, activation='relu', input_dim=num_features))
model.add(BatchNormalization())
model.add(Dropout(0.5))

# Add the second hidden layer with 30 neurons
model.add(Dense(units=30, activation='relu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))

# Add the output layer with 3 neurons and softmax activation for multi-class classification
model.add(Dense(units=3, activation='softmax'))

# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

# Define early stopping callback
early_stop = EarlyStopping(monitor='val_loss', patience=10, verbose=1)

# Train the model with early stopping
history = model.fit(xtrain, ytrain, epochs=37, batch_size=32, validation_data=(xtest, ytest), callbacks=[early_stop])
```

Figure 5.7 – Binary Classification Model

Table 5.2 is a summary of the models equations.

5.3.4 Model Ensembling method

Model ensembling is a valuable technique in machine learning that combines multiple models trained on the same dataset to create a more accurate global model. These models are trained on different data subsets or using various algorithms, allowing them to capture different

Table 5.2 – Table of Equations

Equation	Binary Classification	Categorical Classification
First Hidden Layer	$z^{(1)} = W^{(1)}x + b^{(1)}$ $a^{(1)} = \text{ReLU}(z^{(1)})$ $a^{(1)'} = BN^{(1)}(a^{(1)})$ $a^{(1)''} = \text{Dropout}(a^{(1)'}, p^{(1})$	$z^{(1)} = W^{(1)}x + b^{(1)}$ $a^{(1)} = \text{ReLU}(z^{(1)})$ $a^{(1)'} = BN^{(1)}(a^{(1)})$ $a^{(1)''} = \text{Dropout}(a^{(1)'}, p^{(1})$
Second Hidden Layer	$z^{(2)} = W^{(2)}a^{(1)''} + b^{(2)}$ $a^{(2)} = \text{ReLU}(z^{(2)})$ $a^{(2)'} = BN^{(2)}(a^{(2)})$ $a^{(2)''} = \text{Dropout}(a^{(2)'}, p^{(2})$	$z^{(2)} = W^{(2)}a^{(1)''} + b^{(2)}$ $a^{(2)} = \text{ReLU}(z^{(2)})$ $a^{(2)'} = BN^{(2)}(a^{(2)})$ $a^{(2)''} = \text{Dropout}(a^{(2)'}, p^{(2})$
Output Layer	$y_{pred} = \sigma(W^{(3)}a^{(2)''} + b^{(3)})$	$y_{pred} = \text{softmax}(W^{(3)}a^{(2)''} + b^{(3)})$
Loss Function	$L(y, y_{pred}) = -[y\log(y_{pred}) + (1 - y)\log(1 - y_{pred})$	$L(y, y_{pred}) = -\sum_{i=1}^{num\_classes} y_i \log(y_{pred_i})$

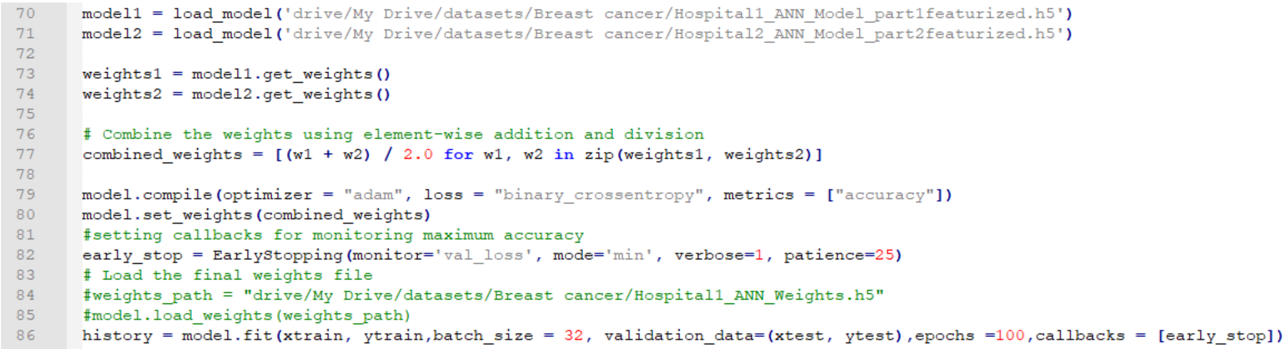
aspects of the problem and make unique predictions. Ensembling helps reduce overfitting, where a model becomes too complex and fits the training data closely but struggles with new data. By using multiple models, ensembling ensures that each model learns different patterns in the data, improving the generalization performance of the final model [113].

In our context, the process of model ensembling involves training the first two artificial neural networks (ANNs) at each hospital using the initial two parts of the datasets for each disease. The third ANN then learns from the collective knowledge of these initial two models, resulting in a notable enhancement in the overall accuracy of the ensemble model. Once the initial two models are trained, their saved files are sent via blockchain to the third hospital, responsible for amalgamating the knowledge from both models. To achieve this, the weights of each model are extracted using the get_weights() method and stored in separate variables. These weights are subsequently combined by iterating through each weight set and performing element-wise addition and division. A list comprehension is utilized to apply the operation (w1 + w2) / 2.0 to each pair of weights obtained from the zip() function. The resulting combined_w variable comprises a list of combined weights that can be utilized to initialize the third neural network model. Figure 5.8 represents the code of the model emsembling method.

Note: It's important to note that the execution of the models was carried out using the Google Colab NVIDIA Tesla T4 GPU (Parallel execution).

5.3.5 Ethereum Blockchain

The Ethereum blockchain technology represents a revolutionary advancement that has taken the blockchain concept to an entirely new dimension. It stands as a decentralized and distributed digital ledger system, garnering widespread attention owing to its flexibility and adaptability. While Ethereum shares a technological foundation with Bitcoin, it diverges in significant ways that make it distinct. One noteworthy distinction lies in Ethereum's utilization of a Turing-complete programming language. This empowers developers to craft intricate and



```
model1 = load_model('drive/My Drive/datasets/Breast cancer/Hospital1_ANN_Model_part1featurized.h5')
model2 = load_model('drive/My Drive/datasets/Breast cancer/Hospital2_ANN_Model_part2featurized.h5')

weights1 = model1.get_weights()
weights2 = model2.get_weights()

# Combine the weights using element-wise addition and division
combined_weights = [(w1 + w2) / 2.0 for w1, w2 in zip(weights1, weights2)]

model.compile(optimizer = "adam", loss = "binary_crossentropy", metrics = ["accuracy"])
model.set_weights(combined_weights)
#setting callbacks for monitoring maximum accuracy
early_stop = EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=25)
# Load the final weights file
#weights_path = "drive/My Drive/datasets/Breast cancer/Hospital1_ANN_Weights.h5"
#model.load_weights(weights_path)
history = model.fit(xtrain, ytrain,batch_size = 32, validation_data=(xtest, ytest),epochs =100,callbacks = [early_stop])
```

Figure 5.8 – Model Ensembling Code

advanced decentralized applications (dApps) capable of executing a wide array of tasks. In contrast, Bitcoin was primarily conceived as a digital currency and possesses limited capabilities in comparison.

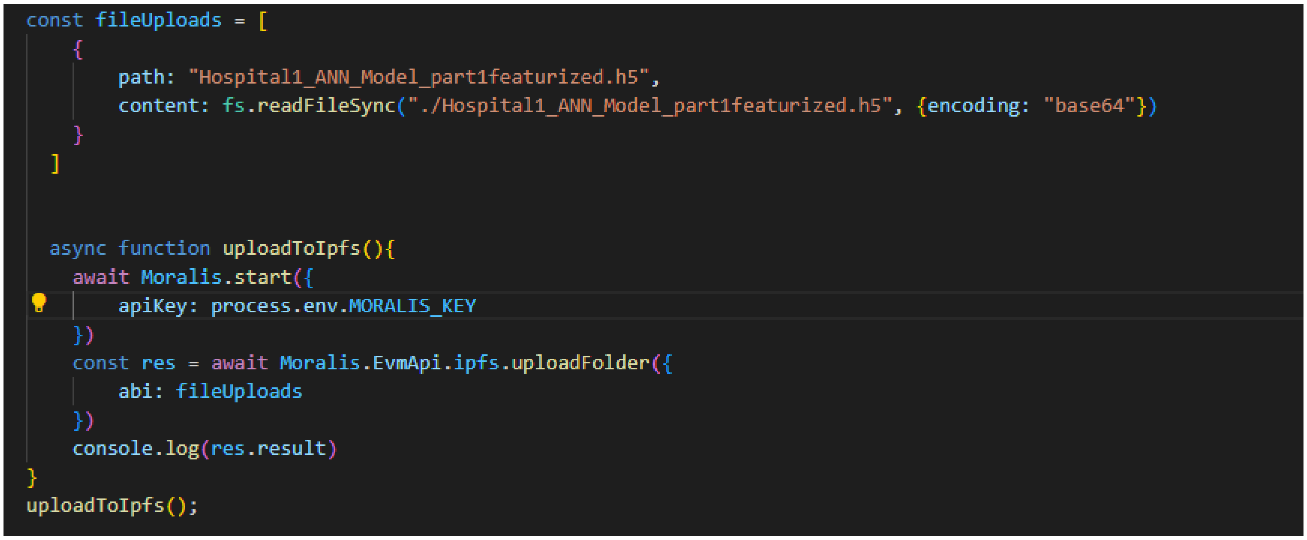
Solidity, the Turing-complete programming language employed by Ethereum, serves as a robust instrument that empowers developers to compose smart contracts. These smart contracts are self-automated agreements designed to trigger automatically when specific conditions are fulfilled. They constitute a pivotal component of the Ethereum ecosystem, possessing boundless possibilities for application across diverse industries like finance, real estate, supply chain management, and numerous others [114].

5.3.6 IPFS

IPFS (InterPlanetary File System) is an innovative technology with the aim of transforming the way we store and distribute files online. IPFS operates as a decentralized protocol and network, facilitating the storage and retrieval of files in a distributed manner, eliminating the dependence on centralized servers. This approach ensures that files are not confined to a single location, bolstering resilience against failures, censorship, and other security concerns. The architecture of IPFS relies on a peer-to-peer network, fragmenting files into smaller segments and distributing them across a network of nodes. This methodology guarantees that users can retrieve files from multiple nodes, resulting in swifter and more efficient file transfers. Additionally, IPFS adopts a content-addressed system where each file is uniquely identified by a hash. This enables files to be located and fetched from any node within the network, ensuring easy accessibility and sharing across various platforms and devices. IPFS has already gained significant traction within the tech community and found applications in various domains, including decentralized social networks, distributed file storage systems, and blockchain-based solutions. As the internet continues to evolve, IPFS holds the potential to emerge as a critical infrastructure component that fosters a genuinely decentralized and open web [115].

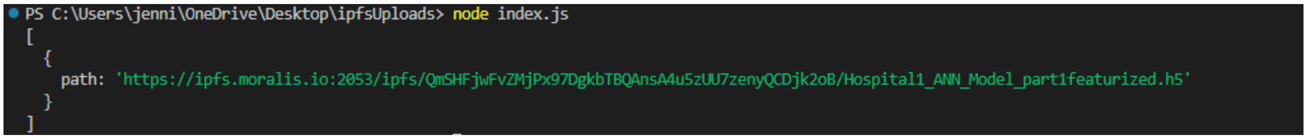
Our Implementation

To begin, hospital 1 and hospital 2 initiate the process by uploading their models to IPFS. They achieve this by constructing an array named "fileUploads," containing an object that represents the file set for upload. This object encompasses both the file's path and its content, which undergo encoding into base64 format through the utilization of the "fs" library. Concurrently, we have established an asynchronous function named "uploadToIpfs," responsible for the actual file upload to IPFS. This function is supplied with the "fileUploads" array as an argument, utilizing it to define the Application Binary Interface (ABI) for the uploaded file. Subsequently, the uploadToIpfs function is invoked, resulting in the generation of a hash representing the file's location on IPFS [116]. The code is in Figure 5.9. The execution is in Figure 5.10.



```
const fileUploads = [
    {
        path: "Hospital1_ANN_Model_part1featurized.h5",
        content: fs.readFileSync("./Hospital1_ANN_Model_part1featurized.h5", {encoding: "base64"})
    }
  ]


  async function uploadToIpfs(){
    await Moralis.start({
        apiKey: process.env.MORALIS_KEY
    })
    const res = await Moralis.EvmApi.ipfs.uploadFolder({
        abi: fileUploads
    })
    console.log(res.result)
}
uploadToIpfs();
```

Figure 5.9 – Model Upload on IPFS Code



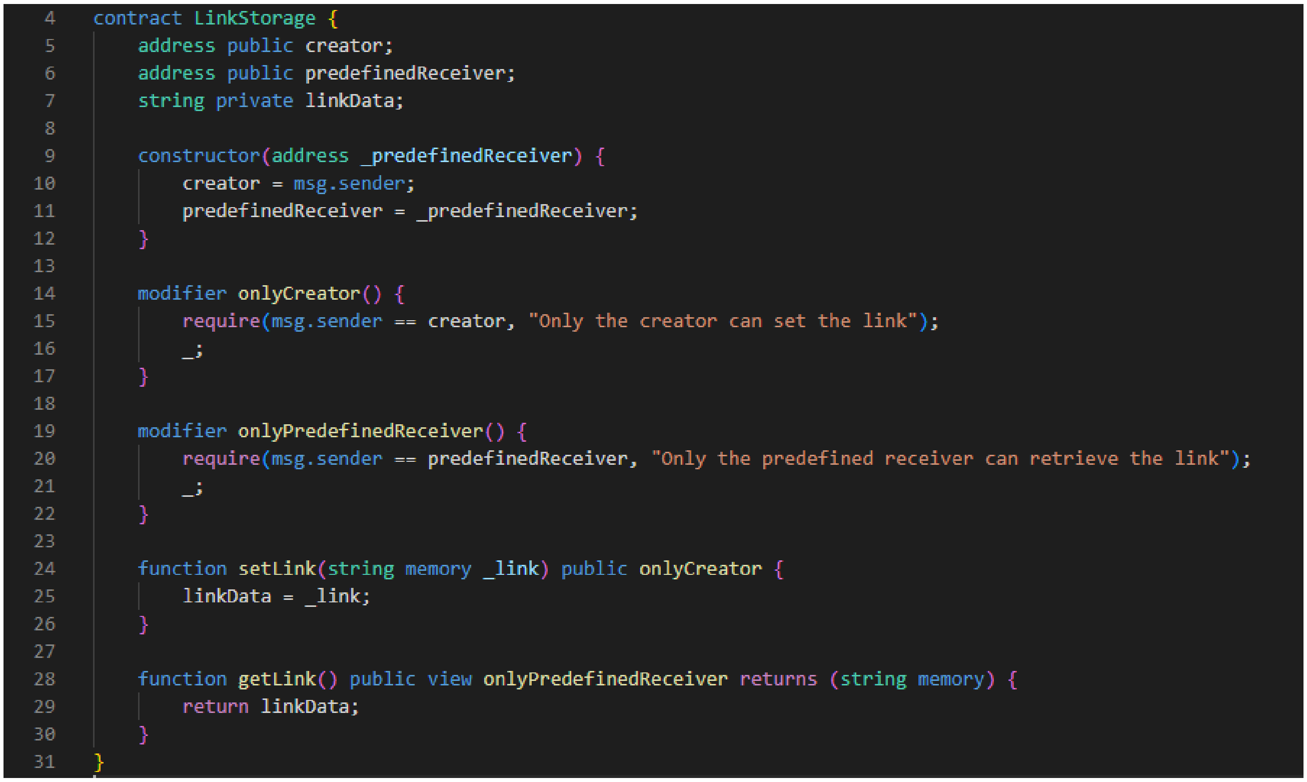
```
PS C:\Users\jenni\OneDrive\Desktop\ipfsUploads> node index.js
[
  {
    path: 'https://ipfs.moralis.io:2053/ipfs/QmSHFjwFvZMjPx97DgkbTBQAnsA4u5zUU7zenyQCDjk2oB/Hospital1_ANN_Model_part1featurized.h5'
  }
]
```

Figure 5.10 – Model Upload on IPFS Execution

Now that we have the IPFS link where the model is stored, we proceed to transmit it via the Blockchain to the third hospital. The Solidity smart contract is designed to manage a link, allowing the creator to set the link and a predefined receiver to retrieve it. It includes state variables to store the Ethereum addresses of the contract creator and predefined receiver, as well as a private variable for storing the link data. The constructor initializes these addresses, with the contract creator being the deployer of the contract. The contract employs two modifiers, "onlyCreator" and "onlyPredefinedReceiver," to restrict access to certain functions. The "setLink" function allows the creator to set the link data, while the "getLink" function permits the predefined receiver to retrieve the link data. This contract enforces a strict access control mechanism, ensuring that only the creator can set the link, and only the predefined receiver can access it, making it suitable for secure link sharing on the Ethereum blockchain [116].

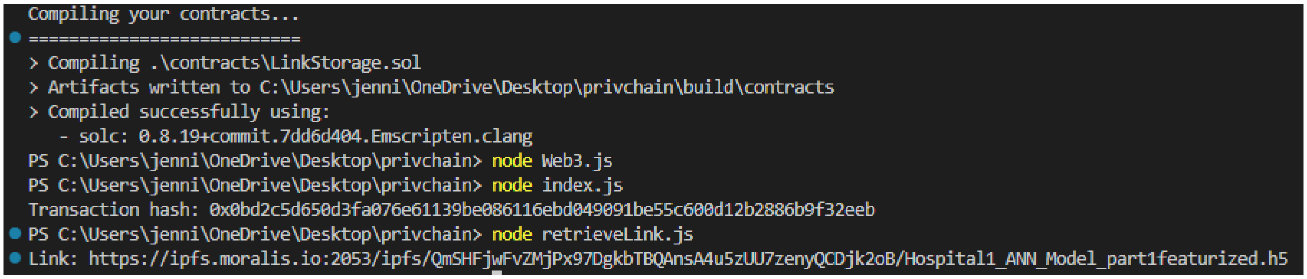
The Code of the smart contract is in Figure 5.11. The execution in represented in Figure 5.12

Ultimately, the third hospital develops a multi-diagnosis application utilizing the global models it has generated and reciprocally shares it with hospital 1 and hospital 2 through the blockchain, employing the same method as before.



```
contract LinkStorage {
    address public creator;
    address public predefinedReceiver;
    string private linkData;

    constructor(address _predefinedReceiver) {
        creator = msg.sender;
        predefinedReceiver = _predefinedReceiver;
    }

    modifier onlyCreator() {
        require(msg.sender == creator, "Only the creator can set the link");
        _;
    }

    modifier onlyPredefinedReceiver() {
        require(msg.sender == predefinedReceiver, "Only the predefined receiver can retrieve the link");
        _;
    }

    function setLink(string memory _link) public onlyCreator {
        linkData = _link;
    }

    function getLink() public view onlyPredefinedReceiver returns (string memory) {
        return linkData;
    }
}
```

Figure 5.11 – Smart contract Code



```
Compiling your contracts...
===========================
> Compiling .\contracts\LinkStorage.sol
> Artifacts written to C:\Users\jenni\OneDrive\Desktop\privchain\build\contracts
> Compiled successfully using:
   - solc: 0.8.19+commit.7dd6d404.Emscripten.clang
PS C:\Users\jenni\OneDrive\Desktop\privchain> node Web3.js
PS C:\Users\jenni\OneDrive\Desktop\privchain> node index.js
Transaction hash: 0x0bd2c5d650d3fa076e61139be086116ebd049091be55c600d12b2886b9f32eeb
PS C:\Users\jenni\OneDrive\Desktop\privchain> node retrieveLink.js
Link: https://ipfs.moralis.io:2053/ipfs/QmSHFjwFvZMjPx97DgkbTBQAnsA4u5zUU7zenyQCDjk2oB/Hospital1_ANN_Model_part1featurized.h5
```

Figure 5.12 – Sending Link On Blockchain Execution

Note: The process of training the local models, uploading them, and transmitting the hash of their IPFS locations is carried out by authorized entity from hospital 1 for local model 1 of each disease and by authorized entity from hospital 2 for local model 2 of each disease. Hospital 3, similarly, has an authorized entity who retrieves the hash values from the blockchain, downloads the models, employs them in constructing the global model for each disease, develops the multi-diagnosis application, and subsequently distributes it back to hospital 1 and 2 through the blockchain.

Within the Ethereum network, charges are applied both for data storage on the blockchain and for conducting transactions. The purpose of these fees is to prevent network abuse and ensure the efficient allocation of resources. the computational work required to execute a transaction or smart contract is quantified using a unit known as "gas." These expenses are settled in ETH, which is the native cryptocurrency of the Ethereum network.

Since all data submitted to the blockchain is permanently retained on every network node, storing data on the Ethereum blockchain can incur significant costs. The gas fees linked to this data storage are directly related to the volume of data being stored.

In the physical world, the prices of gas are determined by the interplay of supply and demand, closely influenced by factors such as market conditions, miner activity, and network congestion.

Reducing transaction expenses and enhancing scalability necessitates careful monitoring of these costs and the consideration of alternative scaling strategies or layer-two solutions, such as sidechains or state channels. IPFS represents one option for cost reduction.

In order to calculate the amount of Ether (ETH) utilized in an Ethereum network transaction, it is essential to consider both the gas consumed and the gas price.

1. Gas Consumption: The total computational work performed during a transaction is denoted by the gas used. It is contingent on the transaction's type, the actions executed, and the interactions between contracts. You can ascertain the gas used value through either the transaction receipt or by querying an Ethereum blockchain explorer.

2. Gas Price: The gas price is denominated in Gwei, signifying the cost for each unit of gas (1 ETH = 1,000,000,000 Gwei). The quantity of ETH paid for each unit of gas consumed is contingent upon the prevailing gas price. Typically, users specify this price when submitting a transaction, and it may vary based on the network and the user's preferences [117].

By multiplying the gas consumed by the gas price, then dividing the outcome by 1,000,000,000, we can ascertain the amount of Ether expended in the transaction.

$Etherused = \frac{\text{Gas used}\times\text{Gas price}}{1{,}000{,}000{,}000}$

For example, if the gas used is 21,000 and the gas price is 10 Gwei:

$Etherused = \frac{21{,}000\times10}{1{,}000{,}000{,}000} = 0.00021ETH$

5.4 Complexity Calculation

This section offers a comparison of the complexity levels between the ANN models, specifically comparing ANN1/ANN2 and ANN3.

5.4.1 Binary classification

5.4.1.1 ANN1/ANN2

The code analysis reveals the following complexities:

Time Complexity:

— Model creation: O(1).

— First layer: O(num_features * 30).

— Batch Normalization and Dropout: O(1).

— Second layer: O(30^2).

— Output layer: O(30).

— Compilation of the model: O(1).

— Early Stopping callback: O(1).

— Model training: O(100 * N).

Space Complexity:

— First layer: O(num_features * 30 + 30).

— Batch Normalization and Dropout: O(30).

— Second layer: O(30^2 + 30).

— Output layer: O(30 + 1).

— Early Stopping callback: O(1).
— Model training: O(1).
The total time complexity of the code is O(number_features * 30 + 30^2 + 100 * N).
The total space complexity of the code is O(number_features * 30 + 30^2 + 31).

5.4.1.2 ANN3

Time Complexity:
— Model creation, First 2 layers, Batch Normalization, Dropout, Output layer, Model compilation, Early Stopping callback, Training are the same as ANN1/ANN2.
— Extracting the weights: O(1).
— Creating combined weights: O(total number of weights).
— Setting combined weights to the model: O(total number of weights).
Space Complexity:
— Model creation, First 2 layers, Batch Normalization, Dropout, Output layer, Model compilation, Early Stopping callback: O(same as ANN1/ANN2).
— Extracting the weights: O(total number of weights).
— Creating combined weights: O(total number of weights).
— Setting combined weights to the model: O(total number of weights).

The total time complexity of the code is O(number_features * 30 + 30 + 30^2 + total number of weights + 100 * N).

The total space complexity of the code is O(number_features * 30 + 30^2 + 30 + 31 + total number of weights + 1).

5.4.2 categorical classification

5.4.2.1 ANN1/ANN2

Here's a summary of the code's complexities:
Complexities shared with the binary classification model:
— Model creation.
— First 2 layers.
— Batch Normalization.
— Dropout.
— Early Stopping Callback.
Complexities specific to this model:
— Output layer: Time complexity O(30 * 3), space complexity O(30 * 3 + 3).
— Training: Time complexity O(37 * N), space complexity O(1).
The total time complexity of the code is O(number_features * 30 + 30^2 + 30 * 3 + 37 * N).
The total space complexity of the code is O(number_features * 30 + 30^2 + 30 * 5 + 3 + 1).

5.4.2.2 ANN3

Complexities shared with the binary classification model:
— Model creation.
— First 2 layers.
— Batch Normalization.
— Dropout.

— Compilation.
— Early Stopping Callback.
Complexities specific to this model:
— Output layer: Time complexity O(30 * 3), space complexity O(30 * 3 + 3).
— Training: Time complexity O(37 * N), space complexity O(1).
The total time complexity of the code is O(number_features * 30 + 30^2 + 30 * 3 + 37 * N).
The total space complexity of the code is O(number_features * 30 + 30^2 + 30 * 5 + 3 + 1).

5.5 Environment and Libraries

To implement our method, we use the following environment and libraries:
— **Environnement:** Google Colab.
— **Libraries:** Numpy, Matplotlib, Pandas, Os, Keras, Sklearn, Seaborn, Streamlit.
— **Programming language:** Python.

5.5.1 Google Colab

Google Colab (short for Google Colaboratory) is a cloud-based programming environment known for its popularity among data scientists, machine learning practitioners, and researchers. It offers a Jupyter Notebook-like interface for running Python code. It has several notable features and advantages:

1. No Setup Required: Google Colab operates entirely in the cloud, eliminating the need to install any software or establish a local development environment. Access is provided via a web browser, making it highly accessible.

2. Free GPU Access: Colab offers free access to Graphics Processing Units (GPUs) and, occasionally, Tensor Processing Units (TPUs). This is particularly valuable for tasks demanding substantial computational power, such as deep learning and extensive data analysis.

3. Jupyter Notebook Integration: Colab seamlessly supports Jupyter notebooks, a favored tool within the data science and machine learning communities. Users can create, edit, and execute Jupyter notebooks directly within Colab.

4. Collaboration: Google Colab facilitates real-time collaboration on shared notebooks among multiple users. It allows for collaborative work on code projects and data analysis.

5. Cost-Efficiency: Colab is offered as a free service, making it a cost-effective choice for those who require cloud-based computing resources.

6. Pre-installed Libraries: The environment comes equipped with a comprehensive set of pre-installed Python libraries and packages commonly employed in data science and machine learning, thus streamlining the development process.

7. Google Drive Integration: Users can effortlessly save and access their Colab notebooks within Google Drive, offering storage and backup convenience.

8. Markdown Support: Colab includes support for Markdown cells, permitting users to incorporate formatted text, images, and explanations alongside their code.

9. Code Execution: Colab facilitates cloud-based code execution, enabling resource-intensive tasks to be executed without taxing local machines. Users can schedule long-running operations and leave them running in the cloud.

10. Data Access: Importing data from various sources, including Google Drive and Google Sheets, is straightforward via Colab's built-in functions.

In summary, Google Colab is a versatile and potent tool for a range of programming and data science tasks. It offers a cloud-based alternative to local development environments and boasts numerous features conducive to collaborative work and efficient data analysis [118].

5.5.2 Google Colab Hardware Configurations

— **RAM Capacity:** 12.7GB.
— **Disk:** 107.7GB.
— **GPU:** NVIDIA Tesla T4.

5.5.3 Programming Language

Python, created by Guido van Rossum in 1991, is a widely-used programming language known for its simplicity and readability. It offers features like ease of use, interpreted nature, dynamic typing, code indentation, a rich standard library, cross-platform compatibility, high-level abstraction, open-source availability, extensibility, and a vibrant community. Python's extensive ecosystem includes tools for data science, machine learning, web development, and scripting. Its readability and versatility have made it popular among developers, making it a valuable choice across various domains, from software development to scientific research [119].

5.5.3.1 Libraries

Numpy: NumPy is a fundamental Python library for numerical and scientific computing, offering versatile arrays and a comprehensive suite of mathematical functions for efficient data manipulation. It enables the creation of multi-dimensional arrays, supports various mathematical operations, including broadcasting for array compatibility, and provides powerful indexing and slicing capabilities. NumPy's integration with other libraries like SciPy, pandas, and Matplotlib enhances its utility in data analysis, scientific research, and data visualization. Its efficiency, thanks to its C and Fortran implementations, makes it a preferred choice for numerical tasks, and it is indispensable in fields such as data science, machine learning, physics, and engineering [120].

Matplotlib: Matplotlib is a widely-used Python library for creating versatile and high-quality data visualizations and plots. It offers a plethora of customization options, supports various plot types, and can generate publication-ready graphics. Whether used through its convenient 'pyplot' interface or its more advanced object-oriented approach, Matplotlib is an essential tool for creating static, interactive, or animated visualizations in fields such as data analysis, scientific research, and data presentation. Its seamless integration with other Python libraries and active user community make it a go-to choice for anyone needing to visualize data in Python [121].

Pandas: Pandas is a widely-used Python library for data manipulation and analysis, offering essential data structures like DataFrames and Series to simplify working with structured data. It excels in data cleaning, preprocessing, and transformation, enabling users to handle missing data, merge datasets, and perform powerful data aggregation. Pandas supports various data input/output formats, making it versatile for reading and writing data from diverse sources. With efficient time series analysis capabilities and seamless integration with data visualization libraries, Pandas is indispensable for data scientists, analysts, and researchers seeking to explore, clean, and analyze data efficiently in Python [122].

OS: The 'os' library in Python provides essential functionalities for interacting with the operating system. It enables users to perform a wide range of system-related tasks, including file and directory operations, environment variable manipulation, process management, path manipulation, permissions handling, and file information retrieval. Whether you need to create, delete, or navigate directories and files, modify environment variables, start and manage processes, or work with file paths in a platform-independent way, the 'os' library serves as a crucial tool for system-level operations and system-related tasks in Python programs [123].

Keras: Keras is a user-friendly, high-level deep learning library in Python, widely adopted for its simplicity and flexibility. It offers an intuitive API for designing, training, and evaluating neural networks, abstracting the complexities of deep learning frameworks. Keras supports multiple backends, making it adaptable to various underlying frameworks like TensorFlow and Theano. With modularity and extensibility, users can customize and create their neural network components. It covers a broad spectrum of neural network types, including CNNs and RNNs, and provides pre-trained models for quick adoption. Keras is favored by the deep learning community for its accessibility and seamless integration with other machine learning tools, making it a top choice for developing and deploying neural networks across diverse applications [124].

Sklearn: Scikit-learn, often referred to as sklearn, is a Python machine learning library renowned for its versatility and simplicity. It provides a rich collection of machine learning algorithms, tools for data preprocessing, feature selection, and model evaluation. With a consistent API, it offers a user-friendly experience for experimenting with various algorithms. Scikit-learn supports diverse tasks such as classification, regression, clustering, and dimensionality reduction, making it suitable for a wide range of machine learning applications. Its seamless integration with other scientific Python libraries, hyperparameter tuning capabilities, and features like pipelines for structured workflows make it a go-to choice for both beginners and experts in the field of machine learning and data analysis [125].

Seaborn: Seaborn is a Python data visualization library that builds upon Matplotlib to simplify the creation of informative and visually appealing statistical graphics. It specializes in a wide range of statistical plots, offering simplicity and ease of use with built-in styles and color palettes that enhance plot aesthetics. Seaborn excels in visualizing complex datasets, especially categorical data, and integrates seamlessly with Pandas DataFrames. Its features include facet grids for multidimensional exploration, support for custom color palettes, and statistical estimation within plots. With an emphasis on both functionality and aesthetics, Seaborn is a valuable tool for data professionals seeking to visually explore, understand, and communicate insights from their data [121].

Streamlit: Streamlit is a Python library that simplifies the creation of web applications for data science and machine learning. It offers an incredibly user-friendly and intuitive API, enabling data scientists and developers to turn data scripts into interactive web apps with minimal effort. Streamlit provides a wide range of widgets for user interaction and integrates seamlessly with popular data visualization libraries. It's well-suited for rapid prototyping, allowing users to quickly iterate on data-driven apps and share insights in real-time. Streamlit's deployment options make it accessible to a wide audience, and its active community and ecosystem offer numerous extensions and resources for enhancing app functionality, making it a powerful tool for creating interactive data applications without the need for extensive web development expertise [126].

5.6 Conclusion

In this chapter we explained the contribution of this thesis underlining the potential power that came from combining the Deep Learning technology with the Blockchain technology for building a robust system that enables hospitals and healthcare providers to make collaborations for improving the healthcare sector without compromising the data privacy of the patients.

Chapter 6

Results and Discussion

6.1 Introduction

This chapter provides the results of applying our research on three types of diseases (Breast cancer, Lung cancer, and diabetes) proving that our research is reliable in terms of security and accuracy.

The results shown by each model are represented in Tables 5.2 to 5.4, and Figures 6 to 17.

6.1.1 Breast Cancer

Table 5.2 displays the outcomes achieved by training three Artificial Neural Network (ANN) models with the breast cancer dataset, showcasing metrics such as Accuracy, Precision, Recall, and F1-score. Additionally, Figures 5.13, 5.14, and 5.15 illustrate graphical representations of the Training vs. Validation Accuracy and loss for each ANN model. Figure 5.16 provides a comparative analysis of the results obtained from these models, while Figure 5.17 presents a confusion matrix depicting the performance of the 3rd model that made predictions on a dataset consisting of 39 instances.

The matrix comprises two rows and two columns. The first row pertains to instances classified within the Malignant class, and the second row pertains to instances classified within the Benign class. The first column contains instances predicted as positive by the model, while in the second column the instances predicted as negative.

In this case, the confusion matrix shows that:

There are 22 instances that belong to the Malignant class, and the model accurately predicted all of them (i.e., correctly identified as Malignant). There are 16 instances that belong to the Benign class, and the model accurately predicted all of them (i.e., correctly identified as Benign). One instance that belongs to the Malignant class was, however, predicted as Benign by the model (i.e., incorrectly classified as Benign). There are no instances from the Benign class that were mistakenly predicted as Malignant by the model (i.e., no false Malignant predictions).

Table 6.1 – Breast Cancer Results

	Accuracy	**Precision**	**Recall**	**F1-score**
ANN 1	94.74%	100%	90%	95%
ANN 2	94.73%	100%	86%	92%
ANN 3	97.44%	100%	94%	97%

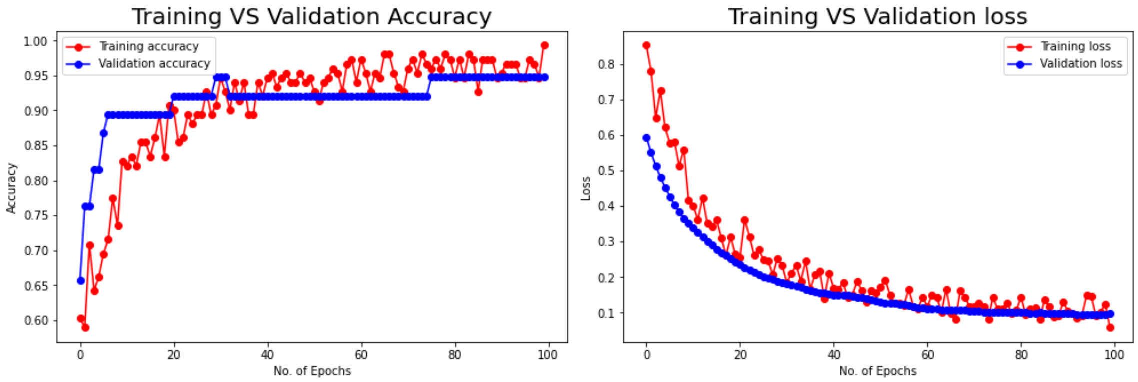


Figure 6.1 – Training vs Validation Accuracy and Loss of ANN 1

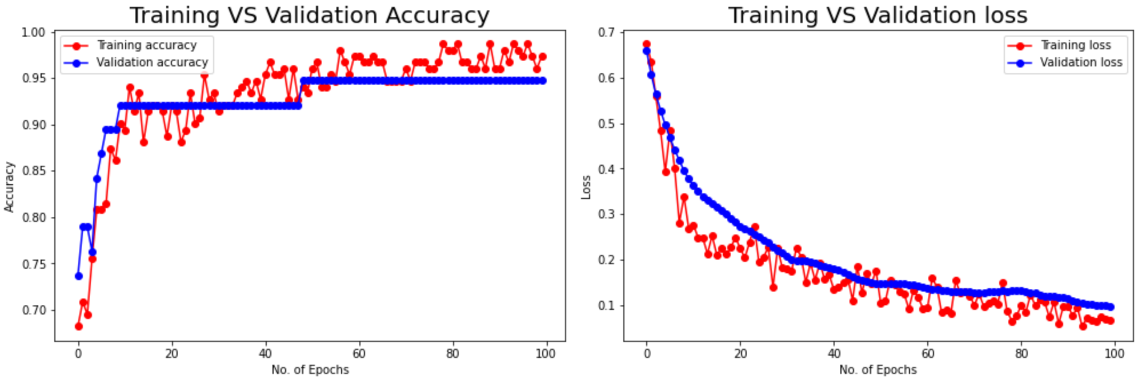


Figure 6.2 – Training vs Validation Accuracy and Loss of ANN 2

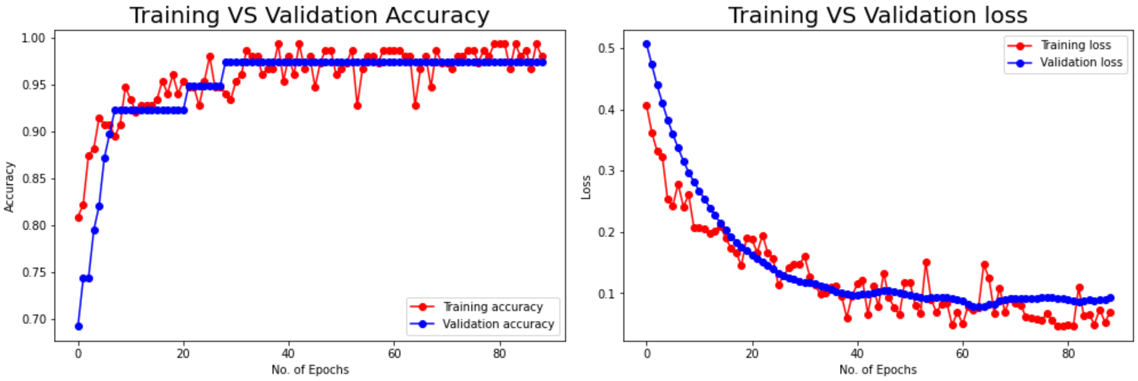


Figure 6.3 – Training vs Validation Accuracy and Loss of ANN 3

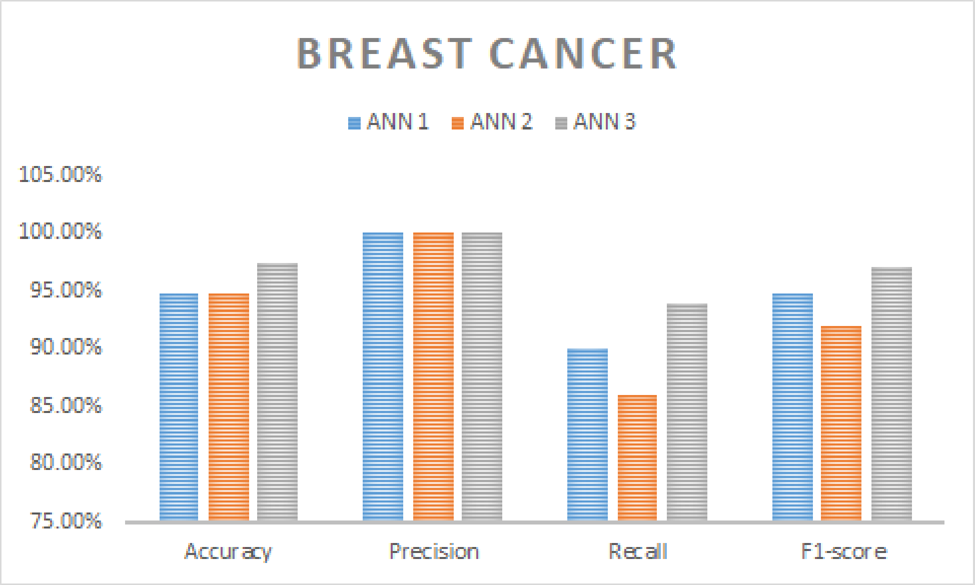


Figure 6.4 – Comparison between the 3 Models

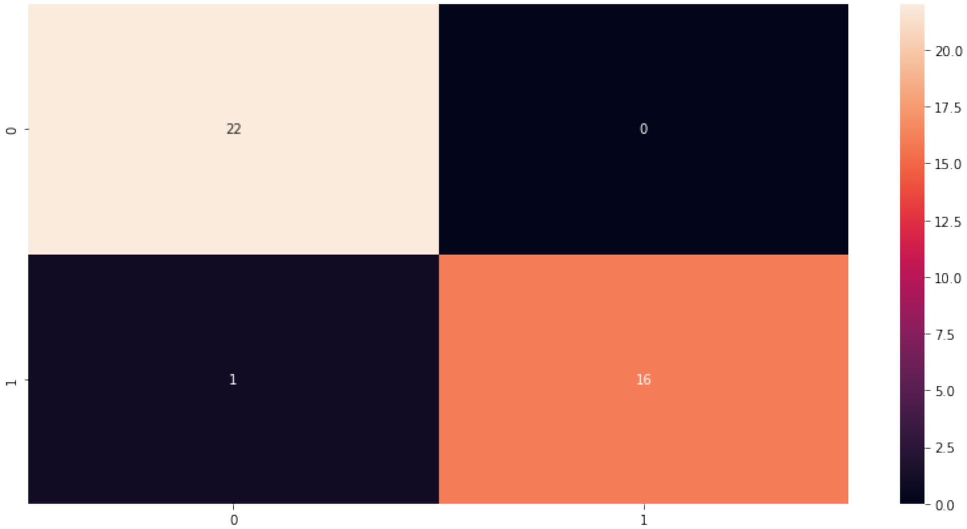


Figure 6.5 – Confusion Matrix of the third ANN

6.1.2 Lung Cancer

Table 5.3 displays the outcomes achieved by training three Artificial Neural Network (ANN) models using the Lung cancer dataset, showcasing metrics such as Accuracy, Precision, Recall, and F1-score. Furthermore, Figures 5.18, 5.19, and 5.20 illustrate graphical representations of Training vs. Validation Accuracy and loss for each ANN model. Figure 5.21 provides a comparative analysis of the results obtained from these models, while Figure 5.22 presents a confusion matrix depicting the performance of the third model that made predictions on a dataset consisting of 67 instances.

The matrix is structured with three rows and three columns. The initial row pertains to instances classified under the "low" class, the second row corresponds to instances in the "medium" class, and the third row represents instances categorized as "high" class. In parallel, the first column denotes instances predicted as "low" by the model, the second column signifies instances predicted as "medium," and the third column designates instances predicted as "high."

In this case, the confusion matrix shows that:

There are 16 instances within the "low" class, and the model accurately predicted all of them, signifying true positives for the "low" category. In the "medium" class, there are 23 instances, all of which the model correctly identified, demonstrating true positives for the "medium" class. Within the "high" class, there exist 27 instances, all of which the model accurately classified, representing true positives for the "high" class. One instance originally classified as "medium" was incorrectly predicted as "low" by the model, resulting in a false negative for the "medium" class. There are no instances that were truly categorized as either "low" or "high" classes but were misclassified as "medium" or "low/high." Thus, there are no instances of false positives.

Table 6.2 – Lung Cancer Results

	Accuracy	**Precision**	**Recall**	**F1-score**
ANN 1	95.52%	90%	100%	95%
ANN 2	92.54%	91%	100%	95%
ANN 3	98.51%	94%	100%	97%

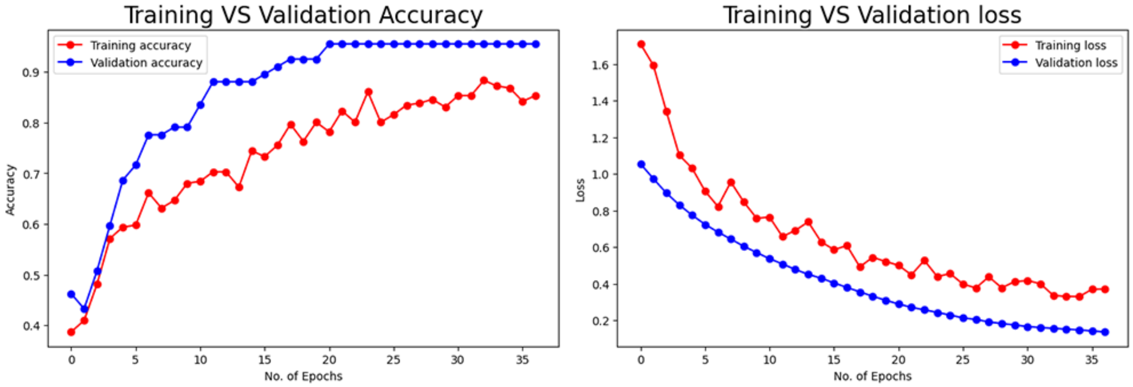


Figure 6.6 – Training vs Validation Accuracy and Loss of ANN 1

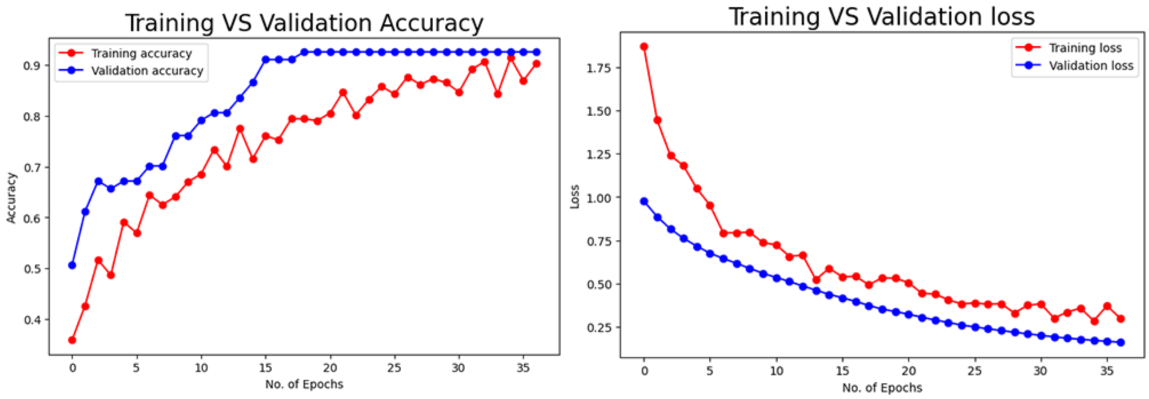


Figure 6.7 – Training vs Validation Accuracy and Loss of ANN 2

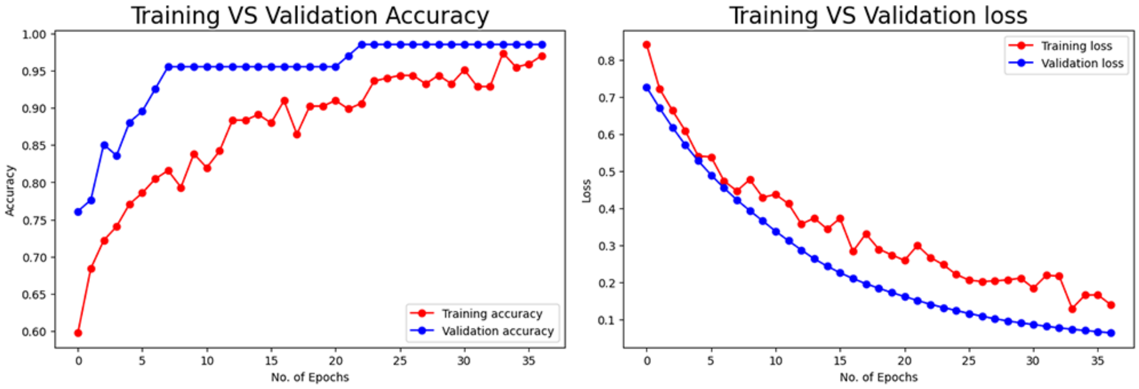


Figure 6.8 – Training vs Validation Accuracy and Loss of ANN 3

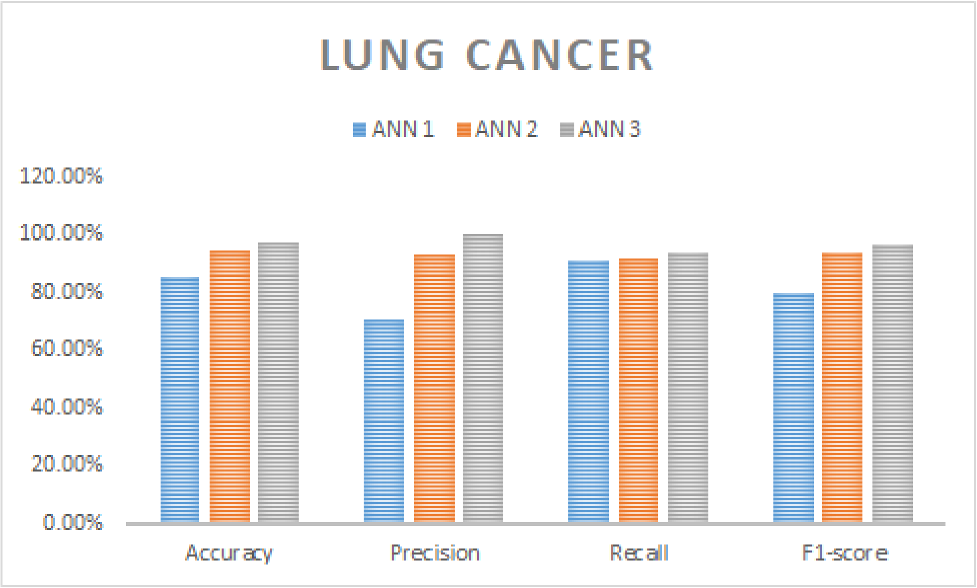


Figure 6.9 – Comparison between the 3 Models

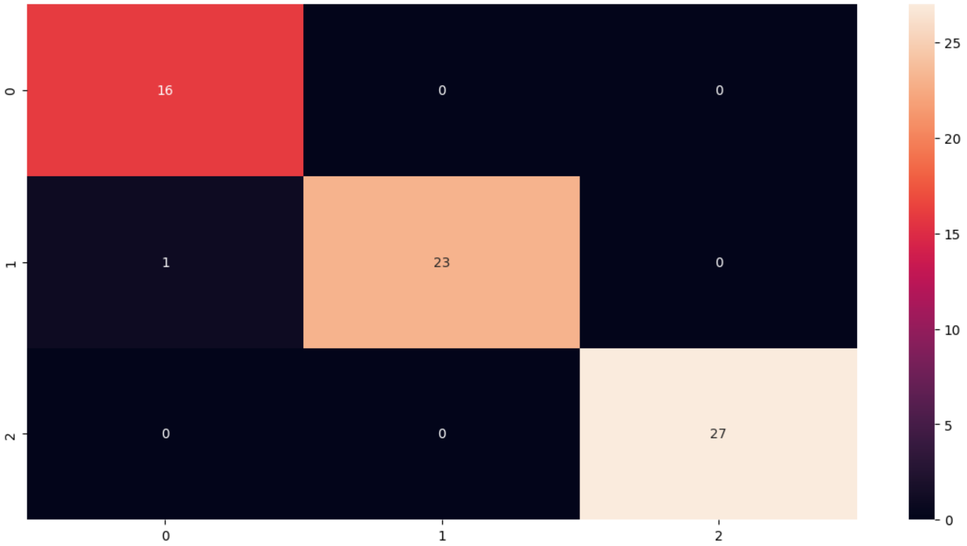


Figure 6.10 – Confusion Matrix of the third ANN

6.1.3 Diabetes

Table 5.4 showcases the outcomes achieved through the training of three Artificial Neural Network (ANN) models using the Diabetes dataset, presenting metrics such as Accuracy, Precision, Recall, and F1-score. Additionally, Figures 5.23 5.24, and 5.25 illustrate graphical representations of Training vs. Validation Accuracy and loss for each individual ANN model. Figure 5.26 provides a comparative analysis of the results obtained from these models, while Figure 5.27 presents a confusion matrix depicting the performance of the third model that made predictions on a dataset comprising 35 instances.

The matrix consists of two rows and two columns. The initial row pertains to instances categorized as part of the positive class, while the subsequent row is associated with instances classified within the negative class. The primary column indicates instances predicted as positive by the model, while the secondary column represents instances predicted as negative.

In this case, the confusion matrix shows that:

There are 16 instances in the positive class, all of which the model accurately predicted (referred to as true positives). There are 18 instances in the negative class, all of which the model correctly predicted (referred to as true negatives). One instance originally from the positive class was incorrectly predicted as negative by the model (referred to as a false negative). No instances from the negative class were erroneously predicted as positive by the model (no false positives).

Table 6.3 – Diabetes Results

	Accuracy	**Precision**	**Recall**	**F1-score**
ANN 1	85.71%	71%	91%	80%
ANN 2	94.73%	93%	92%	94%
ANN 3	97.14%	100%	94%	97%

The evaluation of the global models produced by the third hospital in contrast to the local models generated by hospitals 1 and 2 underscores the merit of our methodology. The findings clearly demonstrate that the global models have surpassed all of the local models in terms of

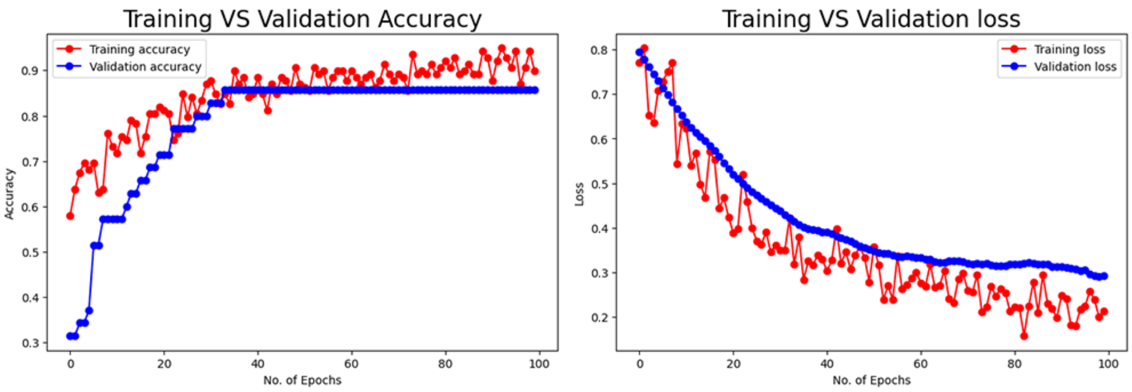


Figure 6.11 – Training vs Validation Accuracy and Loss of ANN 1

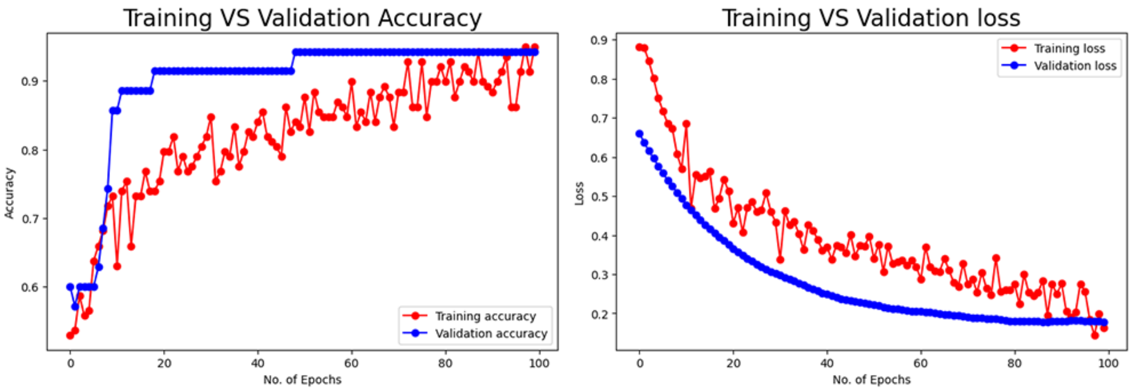


Figure 6.12 – Training vs Validation Accuracy and Loss of ANN 2

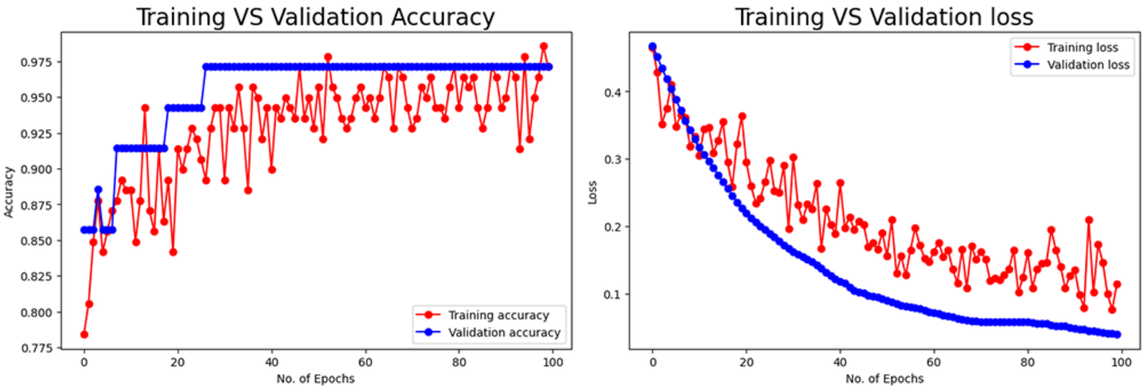


Figure 6.13 – Training vs Validation Accuracy and Loss of ANN 3

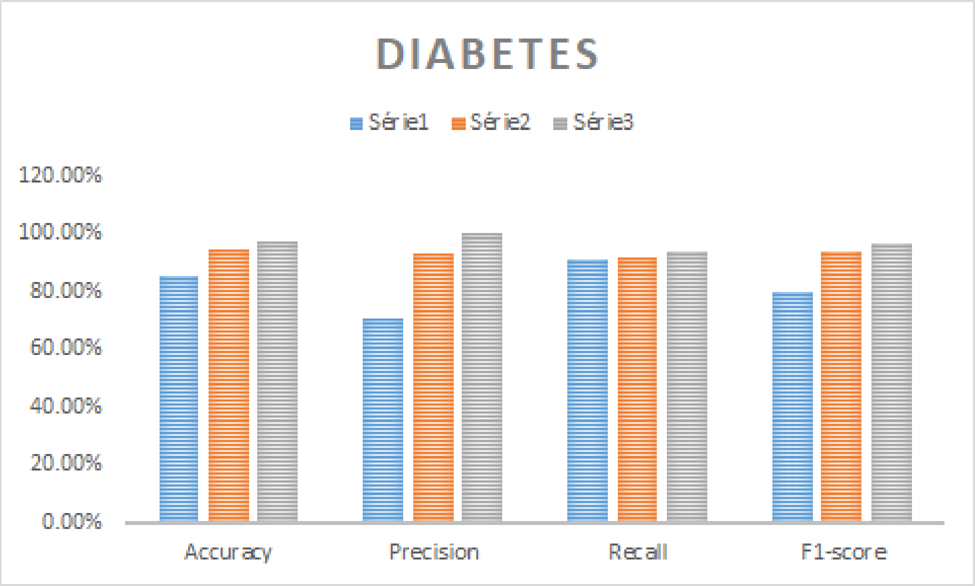


Figure 6.14 – Comparison between the 3 Models

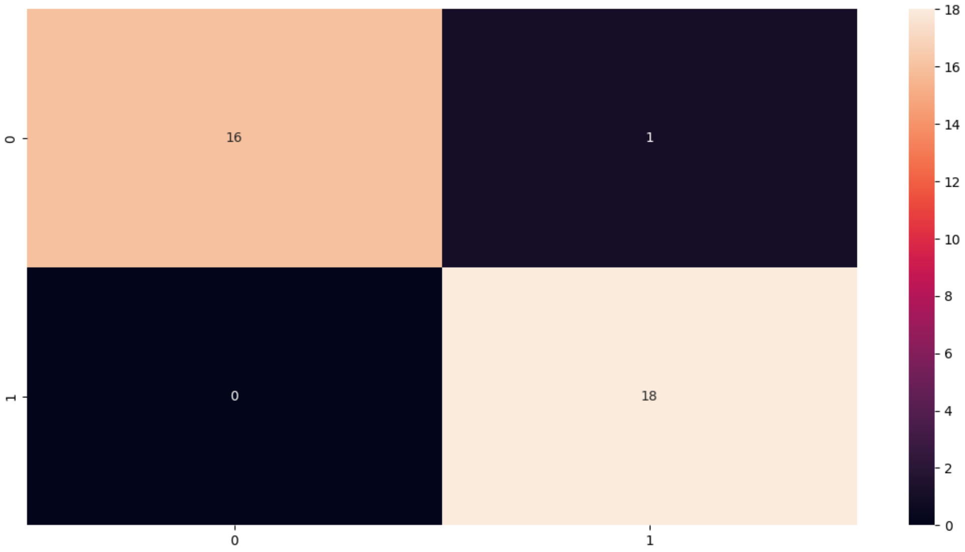


Figure 6.15 – Confusion Matrix of the third ANN

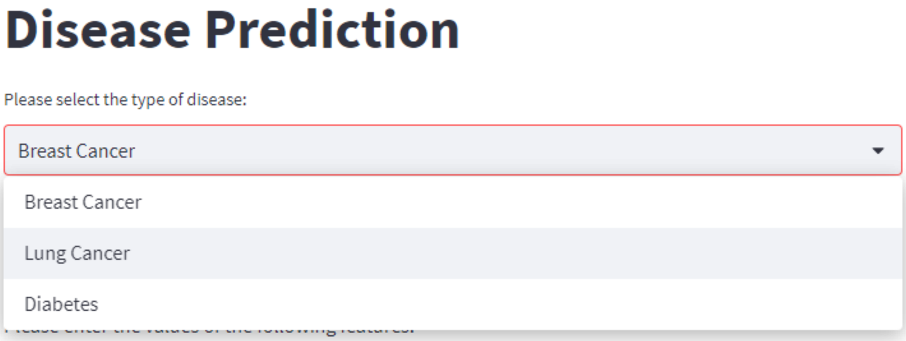
accuracy, precision, recall, and F1-score across all diseases. This underscores that our strategy of combining models from various origins and utilizing them to construct a unified global model has resulted in a notable enhancement in the precision and effectiveness of disease diagnosis.

The findings also indicate that hospitals can derive substantial advantages from embracing our approach to developing global models. This would empower hospitals to harness the combined knowledge and proficiency of multiple institutions, leading to enhanced accuracy and effectiveness in disease diagnosis. Moreover, implementing such methodology can contribute to lowering the expenses and time associated with data collection and analysis, while simultaneously elevating the standard and precision of diagnoses and preserving the confidentiality of patient information.

6.2 Application Overview

Following the completion of training the global models, the third hospital proceeds to develop a multi-diagnosis application using these models. It's crucial to highlight that the scalers, which were adjusted using the dataset, should also be saved to facilitate their application to new data for predictive purposes.

The application is constructed using the Streamlit library, which facilitates the straightforward development of interactive data applications. The code imports the pre-trained deep learning models with the scalers of each disease category. Users can choose the specific disease they wish to predict from a dropdown menu and subsequently fill out a form with relevant feature values for that particular disease. Upon form submission, the input data undergoes preprocessing using the corresponding scaler, and the processed data is subsequently inputted into the pre-trained Artificial Neural Network (ANN) model to estimate the probability of the disease. Finally, the application presents the prediction results to the user. Figures 5.28, 5.29, 5.30 and 5.31 represent an overview of the application interfaces and features.



Disease Prediction

Please select the type of disease:

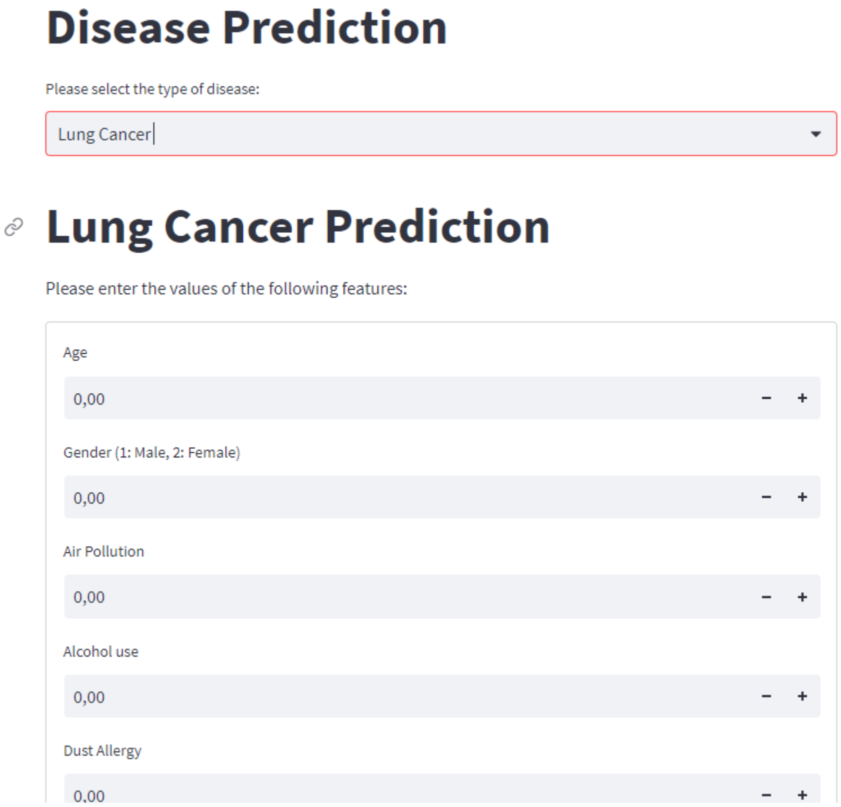
Breast Cancer

Figure 6.16 – Selection of the disease

Figure 5.29 shows the interface of the selected disease with the corresponding form.

Figure 5.30 shows the interface after inputting the symptoms of the patient that will be used in the prediction after clicking the predict button.

After clicking the predict button the prediction result will be shown in the bottom of the interface as represented in Figure 5.31.



Disease Prediction

Please select the type of disease:

Lung Cancer

Lung Cancer Prediction

Please enter the values of the following features:

Age

0,00

Gender (1: Male, 2: Female)

0,00

Air Pollution

0,00

Alcohol use

0,00

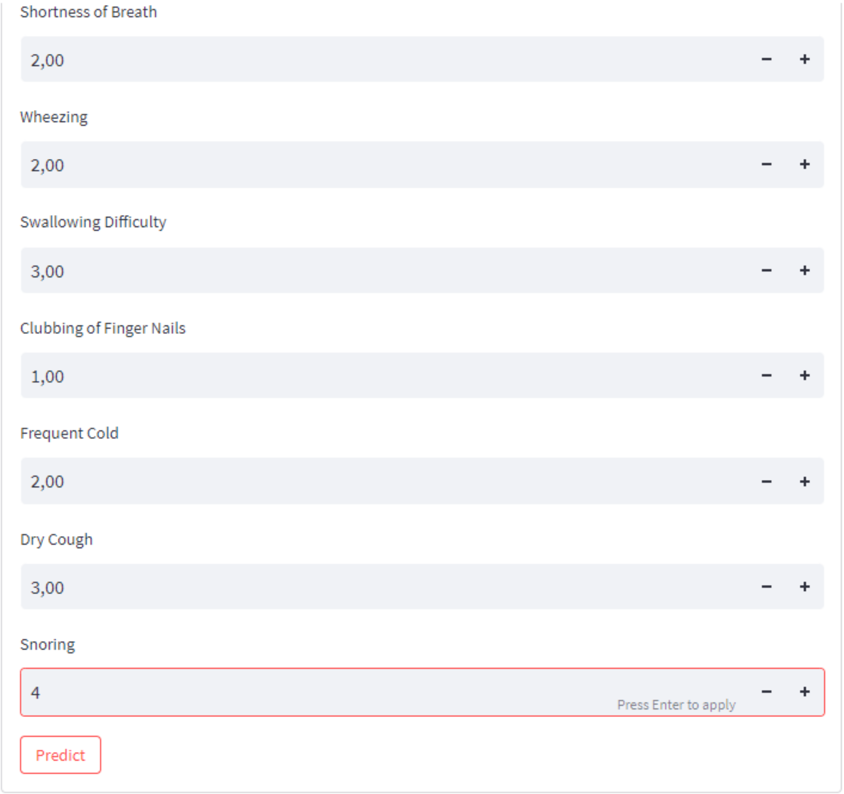
Dust Allergy

0,00

Figure 6.17 – Interface of the selected disease

6.3 Conclusion

The resuls in this chapter showed the efficacy of this approach, achieving high accuracy rates in the diagnosis of three diseases (accuracy of 97.44 % for the Breast Cancer, 97.14 % for the Diabetes, and 98.51 % for the Lung Cancer) that surpass those of individual local models.



Shortness of Breath
2,00

Wheezing
2,00

Swallowing Difficulty
3,00

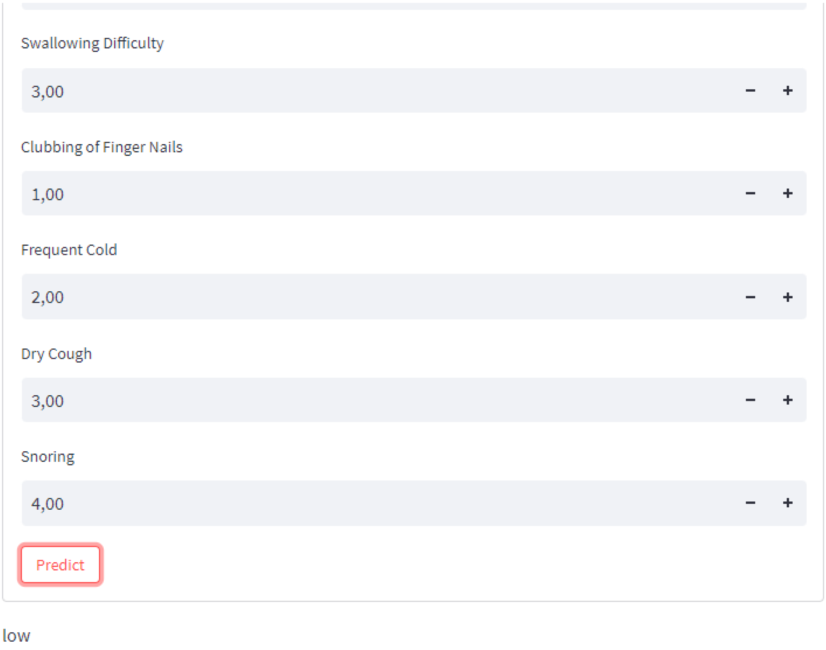
Clubbing of Finger Nails
1,00

Frequent Cold
2,00

Dry Cough
3,00

Snoring
4

Figure 6.18 – Inputting data



Swallowing Difficulty
3,00

Clubbing of Finger Nails
1,00

Frequent Cold
2,00

Dry Cough
3,00

Snoring
4,00

low

Figure 6.19 – Showing the result

Chapter 7

Achievements and Conclusions

In summary, this thesis presents a comprehensive and in-depth exploration of the intersection between advanced computer learning methodologies and the security of digital records, particularly in the healthcare sector. The research thoroughly investigates how integrating sophisticated artificial intelligence models with secure data storage mechanisms can revolutionize medical diagnostics and data protection. Specifically, we have delved into the application of deep learning models in improving the accuracy of disease diagnosis while simultaneously employing blockchain technology to ensure the confidentiality and integrity of sensitive medical records.

The study begins by reviewing existing literature on similar works, analyzing how previous researchers have approached the combination of machine learning and data security in healthcare. Through our extensive review, we identified that many previous works have fallen short in effectively securing medical data. While some methodologies partially ensured data security, they still exhibited significant vulnerabilities related to data leakage, which could expose patient records to unauthorized access and breaches. This major limitation highlighted the pressing need for a more robust and resilient system that could provide a seamless blend of high-performance predictive analytics and airtight data security.

Our research proposes a novel approach that addresses these gaps by utilizing a federated learning strategy, where multiple local deep learning models are trained independently within different healthcare institutions. These locally trained models are then aggregated to form a global deep learning model without ever requiring the actual patient data to be exchanged between entities. This method not only enhances the predictive performance of the deep learning model but also ensures that sensitive patient information remains protected from potential breaches or unauthorized access.

To demonstrate the feasibility and effectiveness of our approach, we successfully developed and implemented a Multi-Diagnosis application capable of accurately diagnosing three different diseases. The application was built upon a global deep learning model, which was created by aggregating multiple local models trained on diverse datasets from different healthcare organizations. The results of our experiments demonstrated that the global model significantly outperformed the individual local models, showcasing improved accuracy, robustness, and generalization capabilities.

A key contribution of our study is the implementation of blockchain technology as a foundational security measure. Instead of exchanging raw patient data across different institutions, only the locally trained deep learning models are shared and aggregated, thereby mitigating the risk of data leakage and unauthorized access. Blockchain ensures that the data-sharing process remains transparent, verifiable, and immutable, thus fostering trust among collaborating healthcare organizations while maintaining the highest standards of data privacy.

Furthermore, the methodology we introduced is not confined solely to the healthcare domain but has broad applicability across multiple industries where secure data-sharing mechanisms are critical. The principles underlying our approach can be seamlessly adapted to fields such as finance, e-commerce, autonomous vehicle technology, cybersecurity, and other domains that require robust data privacy measures while leveraging the immense power of deep learning for data-driven decision-making and predictive analytics. By integrating blockchain technology with federated deep learning, our methodology provides a universal solution that preserves data privacy without compromising the performance and reliability of machine learning applications.

Building upon the success of our study, there are several directions for future research and development that can further enhance the applicability, efficiency, and security of our proposed approach.

One of the foremost future endeavors is to scale our methodology to national and even global datasets. The current study has demonstrated the effectiveness of our method using a limited dataset; however, its true potential can be realized by collaborating with national healthcare organizations to test it on significantly larger datasets. This would allow us to comprehensively assess the benefits and privacy-preserving capabilities of our approach in real-world scenarios involving vast amounts of heterogeneous medical records. By expanding the scope of our study, we aim to showcase the transformative potential of our method to healthcare stakeholders, policymakers, and institutions, thereby encouraging widespread adoption in large-scale healthcare systems.

Another crucial aspect we intend to address in future research is the challenge of dataset heterogeneity. Given that data collected from different healthcare institutions often vary in structure, format, and scale, directly integrating them into a unified deep learning framework poses significant challenges. To overcome this, we plan to develop an advanced data normalization method that can intelligently standardize disparate datasets while preserving their intrinsic characteristics. This will ensure seamless compatibility between various data sources, thereby enhancing the effectiveness and efficiency of our federated learning model. The development of such a robust normalization technique will not only streamline data preprocessing but also enhance the accuracy and reliability of the predictive model, making it more adaptable to diverse medical datasets.

Additionally, we aim to refine and expand the functionalities of our Multi-Diagnosis application to improve its usability and accessibility. Currently, the application requires manual entry of patient information through a form-based interface. While this approach is functional, it may not be the most efficient for large-scale usage in healthcare facilities. To address this limitation, we will enhance the application by integrating a feature that allows users to upload patient data in Excel format. This will significantly improve efficiency by enabling bulk data entry, thereby reducing the time and effort required for inputting medical records. Moreover, incorporating automated data extraction and validation mechanisms will ensure the accuracy and consistency of the input data, further strengthening the application's reliability.

Beyond these improvements, we also recognize the potential of incorporating additional security measures, such as homomorphic encryption and differential privacy, into our framework. These advanced cryptographic techniques will provide an additional layer of protection, ensuring that even if data is intercepted or accessed unlawfully, it remains incomprehensible and unusable by unauthorized entities. By continuously enhancing our methodology with state-of-the-art security techniques, we strive to set new benchmarks for privacy-preserving deep learning applications.

Furthermore, the scalability of our approach will be a key area of focus. As more organizations and institutions adopt our methodology, the computational demands of aggregating numerous deep learning models will increase. To address this challenge, we plan to explore distributed computing techniques, such as edge computing and cloud-based federated learning, to optimize the efficiency and scalability of our approach. By leveraging decentralized computing architectures, we can ensure that our model aggregation process remains seamless, even when dealing with extensive and complex datasets.

In conclusion, this thesis not only lays a strong foundation for integrating deep learning with secure digital record-keeping in healthcare but also paves the way for future advancements in privacy-preserving collaborative learning techniques. The potential applications of our methodology extend far beyond the healthcare sector, offering a revolutionary framework for any industry that prioritizes data security and intelligent analytics. Through continuous refinement and expansion of our approach, we aspire to drive significant progress in both machine learning and data security, ultimately contributing to a safer and more efficient digital ecosystem.

Bibliography

[1] Sandrine Chemla, Thierry Viéville, and Pierre Kornprobst. *Biologically plausible computation mechanisms in cortical areas.* PhD thesis, 10 2006.

[2] Ikram Remadna. *Deep Learning for predictive maintenance.* PhD thesis, Université de mohamed kheider biskra, 2023.

[3] Eugenia Anello. A comprehensive guide of regularization techniques in deep learning. *Towards Data Science*, 2021.

[4] Mishall Al-Zubaidie, Zhongwei Zhang, and Ji Zhang. Pax: Using pseudonymization and anonymization to protect patients' identities and data in the healthcare system. *International Journal of Environmental Research and Public Health*, 16:1490, 04 2019.

[5] Government of British Columbia. Information Security - Multi-Factor Authentication (MFA).

[6] Cyber Yodha. What is a virtual private network(vpn)?, 2023.

[7] Timothy Shim. Everything you need to know about let's encrypt free ssl, 2023.

[8] Cyber Security Agency of Singapore. Transport layer security (tls).

[9] Arpit Jain, Jaspreet Singh, Sandeep Kumar, Țurcanu Florin-Emilian, Mihaltan Traian Candin, and Premkumar Chithaluru. Improved recurrent neural network schema for validating digital signatures in vanet. *Mathematics*, 10(20):3895, 2022.

[10] Yao Du, Zehua Wang, and Victor CM Leung. Blockchain-enabled edge intelligence for iot: Background, emerging trends and open issues. *Future Internet*, 13(2):48, 2021.

[11] Misha Abraham, AH Vyshnavi, Chungath Srinivasan, and PK Namboori. Healthcare security using blockchain for pharmacogenomics. *Journal of International Pharmaceutical Research*, 46:529–533, 2019.

[12] Hamid Nasiri and Seyed Ali Alavi. A novel framework based on deep learning and anova feature selection method for diagnosis of covid-19 cases from chest x-ray images. *Computational intelligence and neuroscience*, 2022, 2021.

[13] Ricardo Carreño Aguilera, Miguel Patino Ortiz, Adan Acosta Banda, and Luis Enrique Carreño Aguilera. Blockchain cnn deep learning expert system for healthcare emergency. *Fractals*, 29(06):2150227, 2021.

[14] Tien-En Tan, Ayesha Anees, Cheng Chen, Shaohua Li, Xinxing Xu, Zengxiang Li, Zhe Xiao, Yechao Yang, Xiaofeng Lei, Marcus Ang, et al. Retinal photograph-based deep learning algorithms for myopia and a blockchain platform to facilitate artificial intelligence medical research: a retrospective multicohort study. *The Lancet Digital Health*, 3(5):e317–e329, 2021.

[15] Xin Guo, Muhammad Arslan Khalid, Ivo Domingos, Anna Lito Michala, Moses Adriko, Candia Rowel, Diana Ajambo, Alice Garrett, Shantimoy Kar, Xiaoxiang Yan, et al. Smartphone-based dna diagnostics for malaria detection using deep learning for local decision support and blockchain technology for security. *Nature Electronics*, 4(8):615–624, 2021.

[16] Randhir Kumar, Prabhat Kumar, Rakesh Tripathi, Govind P Gupta, AKM Najmul Islam, and Mohammad Shorfuzzaman. Permissioned blockchain and deep-learning for secure and efficient data sharing in industrial healthcare systems. *IEEE Transactions on Industrial Informatics*, 2022.

[17] Ji Woong Kim, Su Jin Kim, Won Chul Cha, and Taerim Kim. A blockchain-applied personal health record application: Development and user experience. *Applied Sciences*, 12(4):1847, 2022.

[18] Hesheng Song, Carlos Enrique Montenegro-Marin, et al. Secure prediction and assessment of sports injuries using deep learning based convolutional neural network. *Journal of Ambient Intelligence and Humanized Computing*, 12(3):3399–3410, 2021.

[19] Mohamed Abdelkader Aboamer, Mohamed Yacin Sikkandar, Sachin Gupta, Luis Vives, Kapil Joshi, Batyrkhan Omarov, and Sitesh Kumar Singh. An investigation in analyzing the food quality well-being for lung cancer using blockchain through cnn. *Journal of Food Quality*, 2022, 2022.

[20] Hemang Subramanian, Susmitha Subramanian, et al. Improving diagnosis through digital pathology: Proof-of-concept implementation using smart contracts and decentralized file storage. *Journal of medical Internet research*, 24(3):e34207, 2022.

[21] Huru Hasanova, Muhammad Tufail, Ui-Jun Baek, Jee-Tae Park, and Myung-Sup Kim. A novel blockchain-enabled heart disease prediction mechanism using machine learning. *Computers and Electrical Engineering*, 101:108086, 2022.

[22] Eric Appiah Mantey, Conghua Zhou, Vinodhini Mani, John Kingsley Arthur, and Ebuka Ibeke. Maintaining privacy for a recommender system diagnosis using blockchain and deep learning. *Human-centric computing and information sciences*, 2022.

[23] JM Seely and T Alhassan. Screening for breast cancer in 2018—what should we be doing today? *Current Oncology*, 25(s1):115–124, 2018.

[24] Jorine de Haan, Vincent Vandecaveye, Sileny N Han, Koen K Van de Vijver, and Frédéric Amant. Difficulties with diagnosis of malignancies in pregnancy. *Best Practice & Research Clinical Obstetrics & Gynaecology*, 33:19–32, 2016.

[25] C Burnett, JC Bestall, S Burke, E Morgan, RL Murray, S Greenwood-Wilson, GF Williams, and KN Franks. Prehabilitation and rehabilitation for patients with lung cancer: a review of where we are today. *Clinical Oncology*, 2022.

[26] Richard D Neal, Iain J Robbé, Malcolm Lewis, Ian Williamson, and Jane Hanson. The complexity and difficulty of diagnosing lung cancer: findings from a national primary-care study in wales. *Primary health care research & development*, 16(5):436–449, 2015.

[27] Cristina Quispe, Jesús Herrera-Bravo, Zeeshan Javed, Khushbukhat Khan, Shahid Raza, Zehra Gulsunoglu-Konuskan, Sevgi Durna Daştan, Oksana Sytar, Miquel Martorell, Javad

Sharifi-Rad, et al. Therapeutic applications of curcumin in diabetes: a review and perspective. *BioMed Research International*, 2022, 2022.

[28] Merita Arini, Dianita Sugiyo, and Iman Permana. Challenges, opportunities, and potential roles of the private primary care providers in tuberculosis and diabetes mellitus collaborative care and control: a qualitative study. *BMC Health Services Research*, 22(1):215, 2022.

[29] World Health Organization. *Diagnostic Errors: Technical Series on Safer Primary Care*. World Health Organization, Geneva, 2016. Licence: CC BY-NC-SA 3.0 IGO.

[30] CloudHospital. Global misdiagnosis insides - medical error statistics by countries. `https://icloudhospital.com/articles/global-misdiagnosis-insides-medical-error-statistics-by-countries`, 28 Apr 2023.

[31] Alexander Selvikvåg Lundervold and Arvid Lundervold. An overview of deep learning in medical imaging focusing on mri. *Zeitschrift für Medizinische Physik*, 29(2):102–127, 2019. Special Issue: Deep Learning in Medical Physics.

[32] Behnoush Rezaeianjouybari and Yi Shang. Deep learning for prognostics and health management: State of the art, challenges, and opportunities. *Measurement*, 163:107929, 2020.

[33] Weibo Liu, Zidong Wang, Xiaohui Liu, Nianyin Zeng, Yurong Liu, and Fuad E Alsaadi. A survey of deep neural network architectures and their applications. *Neurocomputing*, 234:11–26, 2017.

[34] 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. *electronics*, 8(3):292, 2019.

[35] Shaveta Dargan, Munish Kumar, Maruthi Rohit Ayyagari, and Gulshan Kumar. A survey of deep learning and its applications: a new paradigm to machine learning. *Archives of Computational Methods in Engineering*, 27:1071–1092, 2020.

[36] Issam El Naqa and Martin J Murphy. What are machine and deep learning? In *Machine and deep learning in oncology, medical physics and radiology*, pages 3–15. Springer, 2022.

[37] Janardan Misra and Indranil Saha. Artificial neural networks in hardware: A survey of two decades of progress. *Neurocomputing*, 74(1-3):239–255, 2010.

[38] Oludare Isaac Abiodun, Aman Jantan, Abiodun Esther Omolara, Kemi Victoria Dada, Nachaat AbdElatif Mohamed, and Humaira Arshad. State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11), 2018.

[39] D. Adi Pratama, M. Abu Bakar, M. Man, and M. Mashuri. Anns-based method for solving partial differential equations: A survey. *Preprints.org*, 2021(020160), 2021.

[40] Mondher Frikha and Ahmed Ben Hamida. A comparative survey of ann and hybrid hmm/ann architectures for robust speech recognition. *American Journal of Intelligent Systems*, 2(1):1–8, 2012.

[41] Ozan Irsoy and Claire Cardie. Opinion mining with deep recurrent neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 720–728, 2014.

[42] Kiran Baktha and BK Tripathy. Investigation of recurrent neural networks in the field of sentiment analysis. In *2017 International Conference on Communication and Signal Processing (ICCSP)*, pages 2047–2050. IEEE, 2017.

[43] Mathias Berglund, Tapani Raiko, Mikko Honkala, Leo Kärkkäinen, Akos Vetek, and Juha T Karhunen. Bidirectional recurrent neural networks as generative models. *Advances in neural information processing systems*, 28, 2015.

[44] Martin Sundermeyer, Tamer Alkhouli, Joern Wuebker, and Hermann Ney. Translation modeling with bidirectional recurrent neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 14–25, 2014.

[45] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

[46] Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 2021.

[47] Jianxin Wu. Introduction to convolutional neural networks. *National Key Lab for Novel Software Technology. Nanjing University. China*, 5(23):495, 2017.

[48] Umberto Michelucci. An introduction to autoencoders. *arXiv preprint arXiv:2201.03898*, 2022.

[49] David Meyer. Introduction to autoencoders, 2015.

[50] Wei Bao, Jun Yue, and Yulei Rao. A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PloS one*, 12(7):e0180944, 2017.

[51] Zhuoyue Lyu, Safinah Ali, and Cynthia Breazeal. Introducing variational autoencoders to high school students. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12801–12809, 2022.

[52] Simukayi Mutasa, Shawn Sun, and Richard Ha. Understanding artificial intelligence based radiology studies: What is overfitting? *Clinical imaging*, 65:96–99, 2020.

[53] Will Koehrsen. Overfitting vs. underfitting: A complete example. *Towards Data Science*, 405, 2018.

[54] Reza Moradi, Reza Berangi, and Behrouz Minaei. A survey of regularization strategies for deep models. *Artificial Intelligence Review*, 53:3947–3986, 2020.

[55] Harouna Soumare, Alia Benkahla, and Nabil Gmati. Deep learning regularization techniques to genomics data. *Array*, 11:100068, 2021.

[56] Mark Schmidt, Alexandru Niculescu-Mizil, Kevin Murphy, et al. Learning graphical model structure using l1-regularization paths. In *AAAI*, volume 7, pages 1278–1283, 2007.

[57] Twan Van Laarhoven. L2 regularization versus batch and weight normalization. *arXiv preprint arXiv:1706.05350*, 2017.

[58] Alex Labach, Hojjat Salehinejad, and Shahrokh Valaee. Survey of dropout methods for deep neural networks. *arXiv preprint arXiv:1904.13310*, 2019.

[59] Ping Luo, Xinjiang Wang, Wenqi Shao, and Zhanglin Peng. Towards understanding regularization in batch normalization. *arXiv preprint arXiv:1809.00846*, 2018.

[60] Lakshmi Annamalai and Chetan Singh Thakur. Theroretical insight into batch normalization: Data dependant auto-tuning of regularization rate. *arXiv preprint arXiv:2209.07587*, 2022.

[61] Garvesh Raskutti, Martin J Wainwright, and Bin Yu. Early stopping and non-parametric regression: an optimal data-dependent stopping rule. *The Journal of Machine Learning Research*, 15(1):335–366, 2014.

[62] Maren Mahsereci, Lukas Balles, Christoph Lassner, and Philipp Hennig. Early stopping without a validation set. *arXiv preprint arXiv:1703.09580*, 2017.

[63] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):1–40, 2016.

[64] Charles Lovering, Rohan Jha, Tal Linzen, and Ellie Pavlick. Predicting inductive biases of pre-trained models. In *International Conference on learning representations*, 2020.

[65] Mudasir A Ganaie, Minghui Hu, AK Malik, M Tanveer, and PN Suganthan. Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence*, 115:105151, 2022.

[66] Ammar Mohammed and Rania Kora. A comprehensive review on ensemble deep learning: Opportunities and challenges. *Journal of King Saud University-Computer and Information Sciences*, 2023.

[67] Rising Odegua. An empirical study of ensemble techniques (bagging, boosting and stacking). In *Proc. Conf.: Deep Learn. IndabaXAt*, 2019.

[68] Martin Sewell. Ensemble learning. *RN*, 11(02):1–34, 2008.

[69] Shaohua Wan and Hua Yang. Comparison among methods of ensemble learning. In *2013 International Symposium on Biometrics and Security Technologies*, pages 286–290. IEEE, 2013.

[70] Shan Lin, Hong Zheng, Bei Han, Yanyan Li, Chao Han, and Wei Li. Comparative performance of eight ensemble learning approaches for the development of models of slope stability prediction. *Acta Geotechnica*, 17(4):1477–1502, 2022.

[71] Xiaozhe Gu, Zixun Zhang, Yuncheng Jiang, Tao Luo, Ruimao Zhang, Shuguang Cui, and Zhen Li. Hierarchical weight averaging for deep neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[72] Nikhilanand Arya and Sriparna Saha. Multi-modal advanced deep learning architectures for breast cancer survival prediction. *Knowledge-Based Systems*, 221:106965, 2021.

[73] Adil Hussain Seh, Mohammad Zarour, Mamdouh Alenezi, Amal Krishna Sarkar, Alka Agrawal, Rajeev Kumar, and Raees Ahmad Khan. Healthcare data breaches: insights and implications. In *Healthcare*, volume 8, page 133. MDPI, 2020.

[74] Farha Nausheen and Sayyada Hajera Begum. Healthcare iot: Benefits, vulnerabilities and solutions. In *2018 2nd International Conference on Inventive Systems and Control (ICISC)*, pages 517–522. IEEE, 2018.

[75] David R Nerenz, Bernadette McFadden, Cheryl Ulmer, et al. Race, ethnicity, and language data: standardization for health care quality improvement. *National Academies Press*, 2009.

[76] Seyedmostafa Safavi, Ahmad Moaaz Meer, Ed Keneth Joel Melanie, and Zarina Shukur. Cyber vulnerabilities on smart healthcare, review and solutions. In *2018 Cyber Resilience Conference (CRC)*, pages 1–5. IEEE, 2018.

[77] Derek Mohammed, Ronda Mariani, Shereeza Mohammed, et al. Cybersecurity challenges and compliance issues within the us healthcare sector. *International Journal of Business and Social Research*, 5(2):55–66, 2015.

[78] Mohammad Mohammad Amini, Marcia Jesus, Davood Fanaei Sheikholeslami, Paulo Alves, Aliakbar Hassanzadeh Benam, and Fatemeh Hariri. Artificial intelligence ethics and challenges in healthcare applications: A comprehensive review in the context of the european gdpr mandate. *Machine Learning and Knowledge Extraction*, 5(3):1023–1035, 2023.

[79] Giovanni Russello, Changyu Dong, and Naranker Dulay. Consent-based workflows for healthcare management. In *2008 IEEE Workshop on Policies for Distributed Systems and Networks*, pages 153–161. IEEE, 2008.

[80] Nicolas Terry. Existential challenges for healthcare data protection in the united states. *Ethics, Medicine and Public Health*, 3(1):19–27, 2017.

[81] Pouyan Esmaeilzadeh. Use of ai-based tools for healthcare purposes: a survey study from consumers' perspectives. *BMC medical informatics and decision making*, 20(1):1–19, 2020.

[82] Matthew J Moyer and Mustaque Abamad. Generalized role-based access control. In *Proceedings 21st International Conference on Distributed Computing Systems*, pages 391–398. IEEE, 2001.

[83] David Ferraiolo, Janet Cugini, D Richard Kuhn, et al. Role-based access control (rbac): Features and motivations. In *Proceedings of 11th annual computer security application conference*, pages 241–48, 1995.

[84] Sanjar Ibrokhimov, Kueh Lee Hui, Ahmed Abdulhakim Al-Absi, Mangal Sain, et al. Multi-factor authentication in cyber physical system: A state of art survey. In *2019 21st international conference on advanced communication technology (ICACT)*, pages 279–284. IEEE, 2019.

[85] Yogendra Shah, Vinod Choyi, and Lakshmi Subramanian. Multi-factor authentication as a service. In *2015 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering*, pages 144–150. IEEE, 2015.

[86] Rui Zhang, Rui Xue, and Ling Liu. Searchable encryption for healthcare clouds: A survey. *IEEE Transactions on Services Computing*, 11(6):978–996, 2017.

[87] Karim Abouelmehdi, Abderrahim Beni-Hessane, and Hayat Khaloufi. Big healthcare data: preserving security and privacy. *Journal of big data*, 5(1):1–18, 2018.

[88] Ramachandran Venkateswaran. Virtual private networks. *IEEE potentials*, 20(1):11–15, 2001.

[89] J Ghebadne, M Dossou, A Vianou, H Yatakpo, and M Assogba. Using secure virtual private networks for increasing the patient prviacy in case of telemonitoring services. *Int. J. Comput. Inf. Technol*, 8:8–11, 2019.

[90] Roza Dastres and Mohsen Soori. Secure socket layer (ssl) in the network and web security. *International Journal of Computer and Information Engineering*, 14(10):330–333, 2020.

[91] Norbert Pohlmann. Transport layer security (tls)/secure socket layer (ssl). In *Cyber-Sicherheit: Das Lehrbuch für Konzepte, Prinzipien, Mechanismen, Architekturen und Eigenschaften von Cyber-Sicherheitssystemen in der Digitalisierung*, pages 439–473. Springer, 2022.

[92] SR Subramanya and Byung K Yi. Digital signatures. *IEEE Potentials*, 25(2):5–8, 2006.

[93] LF Carvalho, G Fernandes Jr, MVO De Assis, JJPC Rodrigues, and M Lemes Proença Jr. Digital signature of network segment for healthcare environments support. *Irbm*, 35(6):299–309, 2014.

[94] Isaac Odun-Ayo, Olasupo Ajayi, Boladele Akanle, and Ravin Ahuja. An overview of data storage in cloud computing. In *2017 International Conference on Next Generation Computing and Information Systems (ICNGCIS)*, pages 29–34. IEEE, 2017.

[95] Barbara Calabrese and Mario Cannataro. Cloud computing in healthcare and biomedicine. *Scalable Computing: Practice and Experience*, 16(1):1–18, 2015.

[96] Massimo Di Pierro. What is the blockchain? *Computing in Science & Engineering*, 19(5):92–95, 2017.

[97] Leila Ismail and Huned Materwala. A review of blockchain architecture and consensus protocols: Use cases, challenges, and solutions. *Symmetry*, 11(10):1198, 2019.

[98] Zibin Zheng, Shaoan Xie, Hongning Dai, Xiangping Chen, and Huaimin Wang. An overview of blockchain technology: Architecture, consensus, and future trends. In *2017 IEEE international congress on big data (BigData congress)*, pages 557–564. Ieee, 2017.

[99] Partha Pratim Ray, Dinesh Dash, Khaled Salah, and Neeraj Kumar. Blockchain for iot-based healthcare: background, consensus, platforms, and use cases. *IEEE Systems Journal*, 15(1):85–94, 2020.

[100] Kosala Yapa Bandara, Subhasis Thakur, and John G Breslin. End-to-end tracing and congestion in a blockchain: A supply chain use case in hyperledger fabric. In *Industry Use Cases on Blockchain Technology Applications in IoT and the Financial Sector*, pages 68–91. IGI Global, 2021.

[101] Harsh Sheth and Janvi Dattani. Overview of blockchain technology. *Asian Journal For Convergence In Technology (AJCT) ISSN-2350-1146*, 2019.

[102] Olli-Pekka Heinisuo, Valentina Lenarduzzi, and Davide Taibi. Asterism: Decentralized file sharing application for mobile devices. In *2019 7th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud)*, pages 38–47. IEEE, 2019.

[103] Miguel Pincheira, Elena Donini, Massimo Vecchio, and Salil Kanhere. A decentralized architecture for trusted dataset sharing using smart contracts and distributed storage. *Sensors*, 22(23):9118, 2022.

[104] Amna Qureshi and David Megías Jiménez. Blockchain-based multimedia content protection: Review and open challenges. *Applied Sciences*, 11(1):1, 2020.

[105] Abid Haleem, Mohd Javaid, Ravi Pratap Singh, Rajiv Suman, and Shanay Rab. Blockchain technology applications in healthcare: An overview. *International Journal of Intelligent Networks*, 2:130–139, 2021.

[106] William H Wolberg, W Nick Street, and Olvi L Mangasarian. Breast cancer wisconsin (diagnostic) data set. *UCI Machine Learning Repository [http://archive. ics. uci. edu/ml/]*, 1992.

[107] UCI Machine Learning Repository and Kaggle. Breast cancer wisconsin (diagnostic) data, 2017.

[108] Cancer Data Health Program. Lung cancer data. `https://data.world/cancerdatahp/lung-cancer-data`, 2016.

[109] Cancer Data Health Program. Lung cancer prediction: Air pollution, alcohol, smoking & risk of lung cancer. `https://www.kaggle.com/datasets/thedevastator/cancer-patients-and-air-pollution-a-new-link`, 2022.

[110] Alimuddin Ahmed, Tanzila Ali, and Atiya Khanum. Early stage diabetes risk prediction dataset. `https://archive.ics.uci.edu/ml/datasets/Early+stage+diabetes+risk+prediction+dataset`, 2014.

[111] Ahmad Alakaaay. Diabetes uci dataset. `https://www.kaggle.com/datasets/alakaaay/diabetes-uci-dataset`, 2020.

[112] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[113] Jason Brownlee. *Ensemble learning algorithms with Python: Make better predictions with bagging, boosting, and stacking*. Machine Learning Mastery, 2021.

[114] Andreas M Antonopoulos and Gavin Wood. *Mastering ethereum: building smart contracts and dapps*. O'reilly Media, 2018.

[115] Peter E Morrison. Ipfs: The interplanetary file system for decentralized web applications. *Journal of Web Engineering*, 20(3):245–265, 2021.

[116] Hakima Rym Rahal, Sihem Slatnia, Okba Kazar, Ezedin Barka, and Saad Harous. Blockchain-based multi-diagnosis deep learning application for various diseases classification. *International Journal of Information Security*, 2023.

[117] Ethereum Foundation. Ethereum gas documentation. `https://ethereum.org/en/developers/docs/gas/`, May 30, 2023.

[118] Ekaba Bisong and Ekaba Bisong. Google colaboratory. *Building machine learning and deep learning models on google cloud platform: a comprehensive guide for beginners*, pages 59–64, 2019.

[119] Why Python. Python. *Python Releases for Windows*, 24, 2021.

[120] Travis E Oliphant et al. *Guide to numpy*, volume 1. Trelgol Publishing USA, 2006.

[121] Ekaba Bisong and Ekaba Bisong. Matplotlib and seaborn. *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*, pages 151–165, 2019.

[122] Randy Betancourt, Sarah Chen, Randy Betancourt, and Sarah Chen. pandas library. *Python for SAS Users: A SAS-Oriented Introduction to Python*, pages 65–109, 2019.

[123] Hajime Tazaki, Frédéric Uarbani, Emilio Mancini, Mathieu Lacage, Daniel Camara, Thierry Turletti, and Walid Dabbous. Direct code execution: Revisiting library os architecture for reproducible network experiments. In *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*, pages 217–228, 2013.

[124] Nikhil Ketkar and Nikhil Ketkar. Introduction to keras. *Deep learning with python: a hands-on introduction*, pages 97–111, 2017.

[125] Brent Komer, James Bergstra, and Chris Eliasmith. Hyperopt-sklearn. *Automated Machine Learning: Methods, Systems, Challenges*, pages 97–111, 2019.

[126] Mohammad Khorasani, Mohamed Abdou, and Javier Hernández Fernández. Streamlit use cases. In *Web Application Development with Streamlit: Develop and Deploy Secure and Scalable Web Applications to the Cloud Using a Pure Python Framework*, pages 309–361. Springer, 2022.

[127] Hakima Rym Rahal, Sihem Slatnia, Okba Kazar, and Ezedin Barka. Blockchain for optimized pattern recognition: Comparative study. *International Journal of Computing and Digital Systems*, 2023.

[128] Hakima Rym Rahal, Sihem Slatnia, Okba Kazar, and Ezedin Barka. Blockchain for medical security data: a review and perspectives. In *2023 International Conference on Advances in Electronics, Control and Communication Systems (ICAECCS)*, pages 1–6. IEEE, 2023.

List Of Publications

Journal Papers

1. Rahal, Hakima Rym, Sihem Slatnia, Okba Kazar, and Ezedin Barka. "Blockchain for Optimized Pattern Recognition: Comparative Study." International Journal of Computing and Digital Systems (2023) [127].
2. Rahal, Hakima Rym, Sihem Slatnia, Okba Kazar, Ezedin Barka, and Saad Harous. "Blockchain-based Multi-Diagnosis Deep Learning Application for Various Diseases Classification." International Journal of Information Security (2023) [116].

Conference Papers

1. Rahal, Hakima Rym, Sihem Slatnia, Okba Kazar, and Ezedin Barka. "Blockchain for medical security data: a review and perspectives." In 2023 International Conference on Advances in Electronics, Control and Communication Systems (ICAECCS), pp. 1-6. IEEE, 2023 [128].