People's Democratic Republic of Algeria
Ministery of Higher Education and Scientific Research
Mohamed El Bachir El Ibrahimi University of Bordj Bou Arréridj
Faculty of Mathematics and Informatics
Informatics Department

DISSERTATION

Presented in fulfillment of the requirements of obtaining the degree

Master in Informatics

Specialty: Nerworks and Multimedia

THEME

Machine learning web-based application for blood cancer detection

Presented by : **Ait Kaci Arab Ines**

Publicly defended on: 21 /06/2023

In front of the jury composed of:

President: BADAOUI Atika.

Examiner: BENSEFIA Hassina.

Supervisor: SABRI Lyazid.

2022/2023

Acknowledgments

I would like to extend my gratitude to all the individuals who have made this study possible and have contributed to its development in any form.

I am thankful to Dr. ***Sabri Lyazid****, my supervisor, for accepting to guide this master's thesis. You have instilled in me a capacity for research and adaptation. Thank you for the attention and close supervision you provided throughout this work. Your insightful observations and contributions, both in terms of form and content, have pushed me further in striving for excellence. I am delighted and consider myself fortunate to have had you as my thesis advisor. Thank you for your guidance.*

I would also like to express my heartfelt thanks to the members of the jury who have agreed to evaluate this thesis without any reservations and provide me with their undoubtedly relevant remarks. With the benefit of hindsight, these remarks will undoubtedly contribute to the improvement of this work.

Dedication

I dedicate this research to all those who nurture and illuminate the garden of my heart, namely:

- My beloved parents! ***Omar*** *and* ***Sihem Rouabah****, may your sacrifices for us never be in vain. Your upbringing has shaped and rooted the pillars of our identities as members of the cosmopolitan community. Here, you will find the signature of our co-presence and co-belonging to the same life. Thank you to God, thank you to you, thank you to life.*

- To my grandparents: ***Rouabah Abdellah*** *and* ***Mebarkia Seghira****, my beloved grandparents who have always been present and continue to be, to bring happiness to all their children and grandchildren! Thank you for tirelessly working, despite the challenges of age, health, and life's problems, for the well-being of your large family! Thank you for being grandparents who hold education sacred!*

- My two brothers: ***Mohammed Amine*** *and* ***Iyed****.*

- My dear sister ***Nada****.*

- My aunts and uncles.

- My dear departed aunt ***Khalissa****, may Allah have mercy on her*

I dedicate this research too to those who are no longer with us, but who would have loved to know: my dear deceased grandparents, ***Hamou*** *and* ***Bousaleh Baya****. May your spark always shine within us. May your souls rest in peace with the Merciful and the Compassionate.*

Finally, I sincerely hope that this entire world, my world, finds a word of gratitude here. I hope that the effort put into this work meets the expectations of everyone involved.

Remerciements

Qu'il me soit permis de présenter à tout un petit monde de personnes qui ont rendu possible la présente étude et qui ont contribué à son élaboration sous quelque forme que ce soit.

Mes remerciements au docteur ***Sabri Lyazid****, mon encadreur, d'avoir accepté de diriger ce mémoire de master, qui a développé en moi une capacité de recherche et d'adaptation. Merci pour l'attention, la proximité avec laquelle vous avez suivi ce travail. Vos judicieuses observations, vos apports multiples tant du point de vue de la forme que du fond m'ont davantage poussée à l'exigence. Je suis ravie et dois-je en réalité m'estimer chanceuse de vous avoir eu comme directeur de mémoire. Merci pour votre formation.*

Je ne manquerais pas non plus de dire un grand merci aux membres du jury qui ont accepté, sans réserve aucune, d'évaluer ce mémoire à sa juste valeur, et de me faire part de leurs remarques, sûrement pertinentes qui, avec un peu de recul, contribueront, sans nul doute, au perfectionnement du présent travail.

Dédicace

Je dédie cette recherche à tous ceux et celles qui fleurissent et font rayonner le jardin de mon cœur, à savoir:

- *Mes adorables parents !* ***Omar*** *et* ***Sihem Rouabah*** *Que votre sacrifice pour nous ne soit pas vain, daigne. Votre éducation a formé et enraciné les piliers de nos personnes comme membres de la communauté cosmopolite. Trouvez ici la signature de notre co-présence et de notre co-appartenance à une même vie. Merci à Dieu, merci à vous, merci à la vie.*
- *À mes grands-parents :* ***Rouabah Abdellah*** *et* ***Mebarkia Seghira****, mes adorables grands-parents, qui étaient toujours présents et continuent de l'être pour Faire le bonheur de tous leurs enfants et petits enfants, aussi ! Merci de trimer, sans relâche aucune, malgré les péripéties de l'âge, de la santé, des problèmes de la vie, au bien-être de votre grande famille ! Merci d'être des grands parents pour qui les études sont sacrées !*
- *Mes deux frères :* ***Mohammed amine*** *et* ***Iyed****.*
- *Ma chère sœur* **Nada***.*
- *Mes tantes et mes oncles.*
- *Ma chère tante défunte* ***Khalissa*** *, que Dieu ait pitié d'elle.*

Je dédie cette recherche aussi à ceux qui ne sont plus parmi nous, mais qui auraient pourtant aimé, de tout cœur, à savoir : mes chers grands-parents défunts, **Hamou** *et* ***Bousaleh Baya****. Que votre étincelle brille toujours au fond de nous. Que vos âmes se reposent en paix auprès du Clément et du Miséricordieux.*

Enfin, j'espère du fond du cœur que tout ce petit monde, mon monde à moi, trouve ici un mot de reconnaissance. J'espère que l'effort déployé dans le présent travail réponde aux attentes des uns et des autres.

Abstract

While there are various methods for blood cancer diagnosis, including blood tests, bone marrow biopsy, and imaging tests, these methods have limitations regarding accuracy, cost, and patient comfort. Therefore, there is a need for more research on the development of accurate and efficient diagnostic tools for blood cancer detection.

Despite the promising results, there are still gaps in the existing research, and more studies are needed to improve the accuracy and reliability of machine learning-based diagnostic tools for blood cancer detection. Therefore, this study aims to contribute to the existing knowledge by developing a machine learning-based web application for blood cancer detection, which can aid in the early and accurate diagnosis of blood cancer. Our contribution addresses a significant literature gap by developing a machine learning-based web application for blood cancer detection. Moreover, showcasing its potential to enhance diagnostic accuracy, facilitate timely interventions, and keep up-to-date with medical information.

Keywords: Machine learning, Convolutional Neural Network, Support Vector Machines, Blood Cancer Detection, Deep Learning, Medical Information, Web Application, Medical Image Analysis.

Résumé

Bien qu'il existe diverses méthodes de diagnostic du cancer du sang, notamment les tests sanguins, la biopsie de la moelle osseuse et les tests d'imagerie, ces méthodes présentent des limites en termes de précision, de coût et de confort du patient. Par conséquent, il est nécessaire de poursuivre les recherches sur le développement d'outils de diagnostic précis et efficaces pour la détection du cancer du sang.

Malgré les résultats prometteurs, il existe encore des lacunes dans la recherche existante, et d'autres études sont nécessaires pour améliorer la précision et la fiabilité des outils de diagnostic basés sur l'apprentissage automatique pour la détection du cancer du sang. Par conséquent, cette étude vise à contribuer aux connaissances existantes en développant une application Web basée sur l'apprentissage automatique pour la détection du cancer du sang, qui peut aider au diagnostic précoce et précis du cancer du sang. Notre contribution comble une lacune importante dans la littérature en développant une application Web basée sur l'apprentissage automatique pour la détection du cancer du sang. De plus, elle montre son potentiel pour améliorer la précision du diagnostic, faciliter les interventions en temps opportun et se tenir au courant des informations médicales.

Mots clés : Apprentissage Automatique, Réseau de neurones convolutif, Machines à vecteurs de support, Détection Du Cancer Du Sang, Apprentissage En Profondeur, Informations Médicales, Application Web, Analyse D'images Médicales.

ملخص

في حين أن هناك طرقًا مختلفة لتشخيص سرطان الدم ، بما في ذلك اختبارات الدم ، وخزعة نخاع العظم ، واختبارات التصوير ، فإن هذه الطرق لها حدود فيما يتعلق بالدقة والتكلفة وراحة المريض. لذلك، هناك حاجة لمزيد من البحث حول تطوير أدوات تشخيص دقيقة وفعالة للكشف عن سرطان الدم.

على الرغم من النتائج الواعدة ، لا تزال هناك فجوات في البحث الحالي ، وهناك حاجة إلى مزيد من الدراسات لتحسين دقة وموثوقية أدوات التشخيص القائمة على التعلم الآلي للكشف عن سرطان الدم. لذلك ، تهدف هذه الدراسة إلى المساهمة في المعرفة الحالية من خلال تطوير تطبيق ويب قائم على التعلم الآلي للكشف عن سرطان الدم ، والذي يمكن أن يساعد في التشخيص المبكر والدقيق لسرطان الدم. تعالج مساهمتنا فجوة كبيرة من خلال تطوير تطبيق ويب قائم على التعلم الآلي للكشف عن سرطان الدم. علاوة على ذلك ، عرض إمكانياته لتعزيز دقة التشخيص ، وتسهيل التدخلات في الوقت المناسب ، ومواكبة المعلومات الطبية.

الكلمات المفتاحية: التعلم الآلي ، تطبيق ويب ، شبكة عصبية تكاملية، الآلة الفاصلة الداعمة، كشف سرطان الدم ، التعلم العميق ، المعلومات الطبية ، تحليل الصور الطبية.

Table of contents

Abreviations list

DT	Decision Trees
SVM	Support Vector Machines
LR	Logistic Regression
IRB	Institutional Review Board
ANN	Artificial Neural Networks
PCA	Principal Component Analysis
RFE	Recursive Feature Elimination
WHO	World Health Organization
HIV	Human Immunodeficiency Virus
AIDS	Acquired Immunodeficiency Syndrome
CBC	Complete Blood Count
CT	Computed Tomography
MRI	Magnetic Resonance Imaging
CNNs	Convolutional Neural Networks
RFE	Recursive Feature Elimination
DNA	Deoxyribonucleic Acid
IDE	Integrated Development Environment
GUI	Graphical User Interface
CSV	Comma Separated Values
CSS	Cascading Style Sheets
HTML	HyperText Markup Language

Keyword Definition & Meaning

For clarity's sake, we define some key terms and concepts relevant to our research topic of blood cancer detection using machine learning. Furthermore, the definitions provide a common language for communicating and discussing the research findings.

- **Blood cancer:** Also known as hematologic cancer, it is a type of cancer that affects the blood, bone marrow, or lymphatic system.
- **Leukaemia:** is a type of blood cancer affecting the bone marrow and blood. It causes an abnormal increase in white blood cells, affecting the body's ability to fight infections.
- **Lymphoma:** A type of blood cancer that affects the lymphatic system. It causes the abnormal growth of lymphocytes, a type of white blood cell.
- **Myeloma:** is a type of blood cancer affecting the plasma cells, a type of white blood cell-producing antibodies.
- **Machine learning:** is a subset of artificial intelligence that uses algorithms to analyze and learn from data and make predictions or decisions.
- **Deep learning:** A type of machine learning that involves using neural networks with multiple layers to learn and make predictions.
- **Web application:** An application accessed through a web browser designed to perform specific tasks or functions.

List of figures

Chapter 01: General Introduction

1.1. Context

Blood cancer, also known as hematological malignancy, encompasses a group of cancers that affect the production and function of blood cells. Timely and accurate detection of blood cancer is crucial for effective treatment and improved patient outcomes. Traditional diagnostic methods often rely on a manual examination of blood samples, which can be time-consuming, subjective, and prone to human error. The use of machine learning algorithms in healthcare has been a rapidly growing area of research in recent years. The ability of these algorithms to analyze large amounts of data quickly and accurately has led to numerous applications in medical diagnosis and treatment. One area of interest is using machine learning for blood cancer detection, as this type of cancer can be challenging to diagnose and treat. In traditional diagnostic methods, patients often undergo invasive tests and procedures, which can be time-consuming and expensive. Moreover, with the advent of CNNs, we can enhance the diagnostic process through automated and reliable image classification techniques. Our study has the potential to provide a new and practical approach to blood cancer detection, which can contribute to improving the quality of healthcare services and patient outcomes.

1.2. Objectives

In terms of scope, the study focuses on developing a web application for blood cancer detection using machine-learning algorithms. While the study may not be able to address all aspects of blood cancer diagnosis, it has the potential to provide valuable insights into this area and contribute to the broader understanding of machine learning in healthcare. Web-based diagnostics allow keeping up to date with medical information.

1.3. Methodology and results

Our study can ultimately improve patient outcomes and reduce healthcare costs associated with delayed or misdiagnosed cases. Additionally, the proposed approach has the potential to be used in remote or underserved areas where access to specialized healthcare services is limited. The development of a user-friendly web application can also make the diagnostic process more accessible and convenient for both patients and healthcare providers. The existing diagnostic methods for blood cancer detection have inherent limitations that hinder their effectiveness. These methods can be invasive, requiring invasive procedures such as biopsies, which can cause discomfort and pose patient risks. Moreover, they can be time-consuming, as the analysis of samples and interpretation of results may take a significant amount of time, leading to delays in diagnosis and treatment. Machine learning-based solutions leveraging advanced computational techniques to enhance accuracy, enable non-invasive diagnostics, and optimize resource utilization. Moreover, Data transformation techniques are applied to ensure the data is suitable for analysis and modelling. By leveraging Python and specialized machine learning libraries like scikit-learn and TensorFlow, the researchers can effectively implement and experiment with the chosen algorithms. Developing a machine learning-based web application for blood cancer detection can revolutionize the way blood cancer is diagnosed, leading to improved patient outcomes and healthcare efficiency.

1.4. Report outline

We highlight that combing machine learning technics and ontology should increase the efficiency of models for knowledge extraction in medical reports.

All study reviews the current state of the development and evaluation of Natural Language Processing algorithms that map clinical text ontology concepts to quantify the heterogeneity of methodologies used.

Chapter 02 Machine learning for healthcare applications

2.1. Introduction

Machine learning is a rapidly evolving field with significant promise in improving healthcare services. Detecting blood cancer as early as possible is crucial for effective treatment and the best way to improve patient outcomes. In this chapter, we provide an overview of the background and motivation for the study, the problem statement and research questions, the significance and scope of the study, and the methodology and structure of the memorandum.

2.2. Background and motivation for the study

The current state of knowledge on blood cancer detection involves a wide range of research studies, which have explored various aspects related to the diagnosis, treatment, and prevention of blood cancer [1][3][15]. The literature review has identified several key findings related to using machine learning algorithms for blood cancer detection, including the potential for these algorithms to improve the accuracy and efficiency of current diagnostic methods [5]. Studies have also explored the various challenges associated with blood cancer diagnosis, including the need for more accurate and reliable diagnostic tools and the limitations of current diagnostic methods. Additionally, the literature review has identified several gaps in the existing research, particularly in developing machine learning-based diagnostic tools that accurately classify blood cancer subtypes [2].

Overall, the literature review suggests a need for further research in blood cancer detection, particularly in developing more accurate and efficient diagnostic tools to aid in the early and accurate diagnosis of blood cancer. By addressing these gaps in the existing knowledge, this study aims to contribute to the overall understanding of blood cancer detection and pave the way for more effective and efficient diagnostic tools in the future.

Machine learning for healthcare applications has gained traction recently due to its potential to improve diagnostic accuracy, reduce healthcare costs, and enhance patient outcomes. Blood cancer is a complex disease that requires accurate and timely diagnosis to ensure effective treatment. However, current diagnostic methods have limitations, such as low sensitivity and specificity, which can lead to misdiagnosis and delayed treatment. Therefore, developing a machine learning-based web application for blood cancer detection can revolutionize the way blood cancer is diagnosed, leading to improved patient outcomes and healthcare efficiency, as highlighted in [4]. This study explores machine learning algorithms' feasibility and effectiveness in developing a web application for blood cancer detection and evaluating its performance against existing diagnostic methods.

2.3. Blood Cancer: Symptoms, Diagnosis & Treatments

Blood cancer, also known as hematological malignancy, is a type of cancer that affects the blood, bone marrow, or lymphatic system. A complex disease can present with a wide range of symptoms and can be challenging to diagnose accurately. Blood cancers include leukaemia, lymphoma, and myeloma, among the top 10 most common types of cancer worldwide [2].

Early detection and accurate diagnosis are crucial in managing blood cancers and improving patient outcomes. However, the current diagnostic methods for blood cancer detection can be time-consuming, expensive, and invasive. In recent years, machine learning has emerged as a promising approach for improving the accuracy and efficiency of blood cancer detection.

Machine learning is artificial intelligence that enables computers to learn from data and make predictions or decisions without being explicitly programmed. Machine learning-based web applications can analyze large volumes of data and identify patterns and trends that may not be apparent to human observers [4]. By leveraging this technology, we can develop accurate and efficient tools for blood cancer detection, which can improve patient outcomes and save lives.

In this study, we aim to develop a machine learning-based web application for blood cancer detection. We will use a dataset of blood cancer patients and healthy controls to train and evaluate the performance of different machine learning algorithms. Our study has the potential to provide a new and practical approach to blood cancer detection, which can contribute to improving the quality of healthcare services and patient outcomes.

2.4. Problem Statement and Research Questions

The problem statement of our study revolves around the need for an accurate and efficient method for blood cancer detection. Despite the availability of various diagnostic methods, such as bone marrow biopsy and blood tests, blood cancer remains a challenging disease to detect in its early stages [5]. The research questions we aim to address in this study include: Can machine learning algorithms accurately detect blood cancer using patient data? How does the performance of different machine learning algorithms compare in terms of accuracy, sensitivity, and specificity? How can we develop a web application that utilizes machine learning algorithms to improve the speed and accuracy of blood cancer diagnosis?

Our research is crucial because early detection of blood cancer is crucial for better patient outcomes, and current diagnostic methods are time-consuming and expensive. By leveraging machine learning algorithms and developing a web application, we can improve the efficiency and accuracy of blood cancer diagnosis, ultimately leading to better patient outcomes. Our study aims to contribute to the existing knowledge in the field of blood cancer detection and provide a framework for future research in this area. We hypothesize that machine learning algorithms can accurately detect blood cancer, and our study aims to test this hypothesis by comparing the performance of different algorithms using patient data.

2.5. Significance and Scope of the Study

When discussing the significance of the study, it is essential to highlight the potential impact that the research could have on the field of blood cancer detection. One study's primary significance is that it could lead to the development of a more accurate and efficient diagnostic tool, ultimately saving lives by allowing for earlier detection and treatment of blood cancer. Additionally, the study has the potential to contribute to the broader field of machine learning in healthcare by demonstrating the effectiveness of these algorithms in diagnosing and detecting diseases. Therefore, this could lead to further research and development in this area, ultimately improving healthcare outcomes for various diseases.

Another significant aspect of the study is its potential to address existing gaps in the research on machine learning for blood cancer detection. By identifying these gaps and proposing a new approach to the problem, the study can further the understanding of this field and contribute to developing more effective diagnostic tools.

In terms of scope, the study focuses on developing a web application for blood cancer detection using machine-learning algorithms. While the study may not be able to address all aspects of blood cancer diagnosis, it has the potential to provide valuable insights into this area and contribute to the broader understanding of machine learning in healthcare. Therefore, the specific objectives of this study are:

- To explore the different types of blood cancer and their diagnostic methods, as well as the limitations of these methods in terms of accuracy and cost-effectiveness.
- To review the existing literature on machine learning-based approaches for blood cancer detection, and identify the gaps in the research.
- To develop and implement a machine learning-based algorithm for blood cancer detection using a dataset of blood samples.
- To evaluate the performance of the developed algorithm and compare it with other state-of-the-art machine learning algorithms, using various evaluation metrics such as sensitivity, specificity, and accuracy.
- To design and develop a user-friendly web application for blood cancer detection, which integrates the developed algorithm.
- The methodology used in this study involves several steps, including data collection, data preprocessing, feature selection, algorithm development, and performance evaluation. The dataset used in this study will be obtained from a publicly available source and will include samples from different types of blood cancer.

2.6. Overview of the research Topic

The use of machine learning algorithms in healthcare has been a rapidly growing area of research in recent years. The ability of these algorithms to analyze large amounts of data quickly and accurately has led to numerous applications in medical diagnosis and treatment. One area of interest is using machine learning for blood cancer detection, as this type of cancer can be challenging to diagnose and treat. In traditional diagnostic methods, patients often undergo invasive tests and procedures, which can be time-consuming and expensive. Moreover, machine learning algorithms can provide a faster and less invasive method of detecting blood cancer, leading to earlier detection and better patient outcomes. This memorandum aims to provide an overview of the use of machine learning algorithms in blood cancer detection and present a new web application that uses these algorithms to detect blood cancer.

By developing a machine learning-based tool that can accurately detect blood cancer, this research can potentially improve the diagnosis and treatment of this type of cancer and ultimately improve patient outcomes.

2.6.1. Methodologies used in previous research

The methodologies used in previous research are a crucial aspect of the literature review. In fact, by understanding the different methodologies, researchers can determine the strengths and weaknesses of each method and how it contributes to the overall understanding of the research topic [1]. Various methodologies have been used, including case studies, surveys, experimental studies, and meta-analyses. Case studies have been used to examine specific instances of the research topic in detail, providing a comprehensive understanding of the issue in a particular context. Surveys have been used to collect data from large sample sizes and identify patterns or trends in the research area.

By identifying these gaps in the existing literature, this study aims to contribute to the existing knowledge on blood cancer detection by developing a machine learning-based web application that can aid in the early and accurate diagnosis of blood cancer. The proposed research addresses these gaps by developing an accurate, efficient, and user-friendly diagnostic tool for blood cancer detection.

2.6.2. Research questions and hypotheses

The research questions and hypotheses are critical components of any research study, as they provide a framework for the investigation and guide the research process. In this section, we will introduce the research questions and hypotheses to address and explain how they fit into the existing literature on the topic of interest.

The primary research question for this study is: "Can a machine learning-based web application accurately diagnose blood cancer?" This question aims to explore the feasibility and accuracy of using machine learning algorithms to detect blood cancer and contribute to the existing knowledge on the topic. Additionally, this study aims to address the following secondary research questions:

1. How does the accuracy of the machine learning-based diagnostic tool compare to existing diagnostic methods for blood cancer?
2. What factors influence the accuracy of the machine learning-based diagnostic tool for blood cancer detection?
3. What are the potential benefits and limitations of using a machine learning-based diagnostic tool for blood cancer detection?

These research questions are based on the gaps identified in the existing literature and aim to contribute to developing more accurate and efficient diagnostic tools for blood cancer detection. Additionally, this study proposes the following hypotheses:

1. The machine learning-based web application will achieve a higher accuracy rate in blood cancer detection than existing diagnostic methods.
2. The accuracy of the machine learning-based diagnostic tool for blood cancer detection will be influenced by factors such as sample size, data quality, and feature selection.

3. A machine learning-based diagnostic tool for blood cancer detection will benefit significantly, including early detection and improved patient outcomes.

These hypotheses will be tested through a series of experiments and analyses, which will be discussed in detail in the methodology section of this memorandum. By addressing these research questions and hypotheses, this proposed study aims to significantly contribute to the healthcare field by developing a more accurate and efficient diagnostic tool for blood cancer detection.

2.7. Contribution to existing literature

The proposed study on machine learning-based blood cancer detection has the potential to contribute significantly to the existing literature in this field. The current literature on blood cancer detection methods is limited and mainly based on traditional diagnostic techniques such as biopsy, which can be invasive, time-consuming, and expensive. Therefore, there is a need for alternative non-invasive methods that can accurately detect blood cancer in its early stages. Machine learning has emerged as a promising tool for medical diagnosis due to its ability to analyze complex data sets and identify patterns that may not be apparent to human analysts. However, there is a lack of research on applying machine learning algorithms in detecting blood cancer. The proposed study aims to address this gap in the literature by developing a novel machine learning-based approach for blood cancer detection that is non-invasive, accurate, and efficient.

The study aims to use a large dataset of blood cancer patients to develop and test machine-learning models that can accurately classify patients based on their disease status. The study will also investigate the potential of combining multiple machine-learning algorithms to improve the accuracy and robustness of the model. By contributing to the existing literature on machine learning-based blood cancer detection, the proposed study can advance our understanding of this critical area of research and provide a valuable tool for healthcare professionals in diagnosing and treating blood cancer.

2.7.1. Practical significance

The practical significance of this study lies in its potential to improve the accuracy and efficiency of blood cancer detection. By utilizing machine learning algorithms, the proposed approach can provide more accurate and reliable results than traditional diagnostic methods, leading to earlier detection and treatment of blood cancer. Our study can ultimately improve patient outcomes and reduce healthcare costs associated with delayed or misdiagnosed cases. Additionally, the proposed approach has the potential to be used in remote or underserved areas where access to specialized healthcare services is limited. The development of a user-friendly web application can also make the diagnostic process more accessible and convenient for both patients and healthcare providers. Overall, the practical significance of this study is its potential to enhance the diagnosis and treatment of blood cancer, which can significantly impact the healthcare field. Our study will use machine learning algorithms, such as decision trees, random forests, support vector machines, and neural networks, to develop a blood cancer detection model. The performance of these models will be evaluated using various metrics, such as precision, recall, and F1-score. The latter measures a model's accuracy that considers both precision and recall. It is the harmonic mean of precision and recall, with a maximum value of 1 and a minimum value of 0. F1-score is commonly used in binary classification tasks, with two possible outcomes (positive or negative). A high F1-score indicates that the model has high precision and recall, meaning it correctly identifies positive outcomes and does not falsely identify negative outcomes.

The findings of this study will have practical implications for the field of healthcare and blood cancer detection, as they could lead to the development of more accurate, cost-effective, and efficient methods for blood cancer diagnosis. Thus, it could reduce the financial burden on patients and healthcare systems and improve patient outcomes.

2.7.2. Implications for future research

The results and findings can have broader implications for future research in the field.

- *Developing more advanced machine learning algorithms*: While the proposed study utilizes existing algorithms, future research can focus on developing more advanced algorithms to achieve higher accuracy rates and reduce false positives and negatives.
- *Incorporating other medical data:* The proposed study focuses on analyzing blood samples for the presence of cancer cells, but other medical data, such as patient history, can also be incorporated to improve the accuracy of the diagnostic tool.
- *Conducting clinical trials:* While the proposed study utilizes publicly available datasets, future research can focus on conducting clinical trials to validate the effectiveness of the machine learning-based diagnostic tool on patients.
- *Expanding to other types of cancer:* While the proposed study focuses on blood cancer detection, the machine learning-based approach can also be applied to other types of cancer, such as lung or breast cancer.
- *Reduced healthcare costs:* The machine learning-based diagnostic tool can reduce the need for more expensive and invasive diagnostic procedures, such as biopsies, reducing healthcare costs for patients and healthcare providers.
- *Enhanced understanding of cancer:* Further research in machine learning-based cancer detection can lead to a better understanding of the underlying mechanisms of cancer, which can inform the development of new treatments and therapies.

2.8. Overview of the methodology of our research approach

This element provides a personal and detailed account of the methodologies we will employ in conducting our study and the organization of our memorandum. By sharing these insights, we aim to offer a transparent and compelling glimpse into our research journey and provide a solid foundation for the subsequent chapters. With a focus on the research design, data collection methods, data analysis techniques, ethical considerations, and the overall structure of the memorandum, we present a roadmap that will guide us in uncovering valuable insights in the realm of blood cancer detection.

2.8.1. Research Design

Here, we provide a subjective and detailed overview of our research design, its rationale, and its alignment with our research objectives.

1. *Research Design:* Our study incorporates both observational and experimental components. We will begin with an observational approach to gather valuable insights and data from real-life cases and scenarios. Therefore, will involve the collection of existing medical records, patient histories, and diagnostic reports from healthcare institutions specializing in blood cancer treatment. The observational aspect allows us to examine the data's patterns, trends, and associations, providing a foundation for further analysis.
2. *Rationale:* The choice of an observational research design stems from the need to explore the existing landscape of blood cancer detection practices and outcomes. By analyzing real-life cases and data, we can identify the strengths and weaknesses of current diagnostic methods, understand patient characteristics, and detect any potential gaps or challenges in the detection process. This information is crucial for developing effective and targeted solutions.
3. *Experimental Design:* Building upon the observational phase, we will proceed with an experimental design involving the development and implementation of machine learning algorithms. This experimental component allows us to test the effectiveness and accuracy of our proposed algorithms in detecting blood cancer. We will utilize a carefully curated dataset, ensuring its representativeness and diversity to achieve reliable results.
4. *Alignment with Research Objectives:* Our chosen research design aligns with our primary research objectives of improving blood cancer detection through machine learning techniques. The observational phase provides a comprehensive understanding of the current practices and challenges, while the experimental phase enables us to develop and evaluate innovative approaches to enhance accuracy and efficiency.

To illustrate the application of our research design, consider a scenario where we analyze the medical records and diagnostic reports of a large cohort of blood cancer patients. Through the observational phase, we can identify common symptoms, demographic factors, and diagnostic patterns associated with different types of blood cancer. This information can guide the development of machine learning algorithms during the experimental phase, enabling us to automate detection and provide timely and accurate diagnoses. By adopting this combined observational and experimental research design, we aim to uncover novel insights, improve blood cancer detection methods, and ultimately contribute to advancements in the field of healthcare.

2.8.2. Data Collection Methods

Regarding data collection for our study, we have devised a comprehensive approach to ensure the acquisition of high-quality and reliable data. In this section, we will discuss the methods we will employ to collect the necessary data, including the sources of data and the measures we will take to maintain data quality and integrity.

1. ***Sources of Data:*** To gather the essential data for our study, we will rely on multiple sources, each providing unique insights into blood cancer detection. These sources include:
    - *Medical Records:* We will collaborate with healthcare institutions specializing in blood cancer treatment to access anonymized medical records of diagnosed patients. These records contain valuable information such as patient demographics, diagnostic tests, treatment plans, and outcomes.
    - *Clinical Databases:* We will leverage established clinical databases that aggregate data from multiple healthcare facilities. These databases contain a wealth of patient information and allow us to access a more extensive and more diverse dataset, enhancing the generalizability of our findings.
    - *Patient Surveys:* We will design and administer surveys to blood cancer patients to supplement the clinical data. These surveys will collect personal information such as patient experiences, symptoms, and perceptions of the

diagnostic process. Patient feedback is essential for gaining insights into the human side of blood cancer detection and understanding the impact of the disease on individuals.

2. ***Ensuring Data Quality and Integrity:*** Maintaining the quality and integrity of the collected data is of utmost importance to ensure the validity and reliability of our study. We will take care of the following measures:
    - *Data Privacy and Security:* Strict protocols will be followed to protect patient privacy and comply with relevant data protection regulations. All data will be anonymized, and access will be granted only to authorized personnel.
    - *Data Validation and Cleaning:* Before analysis, we will check to identify and rectify any inconsistencies or errors in the dataset. This process involves cross-verifying data points, removing duplicates, and addressing missing or incomplete data.
    - *Quality Control Procedures:* To ensure data accuracy, we will implement quality control measures throughout the data collection process. This includes thorough documentation of data collection procedures, training of data collectors, and regular auditing of data for adherence to predefined standards.
    - *Data Monitoring and Auditing:* The data will be monitored and audited to identify any anomalies or data integrity issues. Regular checks will be conducted to ensure the data remains reliable and fit for analysis.

By employing these data collection methods and adhering to strict quality control measures, we aim to gather robust and dependable data to serve as the foundation for our analysis and subsequent conclusions. The diverse data sources, including medical records, clinical databases, and patient surveys, will provide a comprehensive view of blood cancer detection, enabling us to derive meaningful insights and contribute to the existing knowledge in the field.

2.8.3. Data Analysis Techniques

We will utilize various data analysis techniques and statistical methods to analyze the collected data and gain meaningful insights effectively. These approaches have been carefully chosen based on their relevance to our research questions and their ability to extract valuable information from the data. Among the following critical data analysis techniques, we will employ the first three:

- *Decision Trees (DT):* We can identify hierarchical relationships and decision rules within the data by employing decision trees. This algorithm will help us understand the key factors influencing blood cancer detection and improve diagnostic processes.
- *Support Vector Machines (SVM):* SVM is a robust supervised learning algorithm that enables data classification into different groups. We will use SVM to classify blood cancer cases based on various attributes, contributing to the development of accurate diagnostic models.
- *Neural Networks:* Neural networks excel at capturing complex relationships within the data. They will enable us to uncover hidden patterns and provide insights that may not be evident through traditional statistical methods.
- *Logistic Regression (LR):* Logistic regression allows us to model the probability of binary outcomes, such as the presence or absence of blood cancer. We can gain insights into the factors contributing to blood cancer detection by identifying significant predictors.
- *Survival Analysis:* We will utilize survival analysis techniques, such as Cox proportional hazards models, to explore the impact of different variables on patient survival rates and disease progression. Those techniques will enhance our understanding of the prognostic factors associated with blood cancer.
- *Data Visualization:* Visualizing the data through charts, graphs, and plots will help us better understand the patterns and trends in the blood cancer dataset.
- *Clustering Algorithms:* Clustering algorithms will assist us in identifying distinct subgroups within the blood cancer population. Those algorithms will allow us to tailor treatment approaches based on specific patient characteristics.
- *Association Rule Mining:* We can uncover meaningful associations between different variables by leveraging association rule mining techniques. Association Rule will provide valuable insights into the relationships between various factors related to blood cancer.

The data analysis techniques, including machine learning algorithms, statistical modelling, and other analytical tools, will enable us to extract valuable insights from the collected data. By combining these approaches, we will uncover hidden patterns, identify essential predictors, and develop predictive models for blood cancer detection. Ultimately, our data analysis efforts will contribute to addressing our research questions and achieving our objectives.

2.8.4. Ethical Considerations

Ethical considerations play a vital role in protecting participant rights, privacy, and confidentiality in our study. We are committed to upholding ethical guidelines and regulations throughout the research process. Here are the key ethical considerations and measures we will implement:

1. Participants will have the opportunity to ask questions and make an informed decision regarding their involvement. We will document their consent in writing, and the confidentiality of their personal information will be assured.
2. Each study should assign unique identifiers to ensure participant anonymity instead of using personal identifying information in data analysis and reporting. Therefore, it will further safeguard their privacy.
3. All studies will employ stringent data security measures to protect the integrity and confidentiality of the collected data. Thus, it includes storing data in password-protected systems with restricted access and using encryption methods to prevent unauthorized access.
4. Data will be stored for the designated period required by ethical guidelines, and after completion of the study, it will be securely disposed of to prevent any unintended disclosure.
5. Our study will undergo a thorough ethical review process by the appropriate Institutional Review Board (IRB) or Ethics Committee. This review ensures that the study design, procedures, and ethical considerations align with established guidelines and principles.

6. We will adhere to any specific requirements or recommendations provided by the IRB, addressing any concerns or modifications as required.
7. We are committed to using the collected data responsibly and solely for research. Any dissemination or publication of the study findings will be conducted in a manner that ensures participant confidentiality and anonymity.
8. We will follow data sharing and publication guidelines, providing aggregated or de-identified data when necessary to protect participant privacy.

Any study that addresses these ethical considerations and implements the outlined measures will ensure that our study complies with ethical guidelines and safeguards the rights and well-being of our participants. Upholding ethical standards is paramount in maintaining the integrity and trustworthiness of our research.

2.9. Conclusion

The current literature on blood cancer detection methods is limited and mainly based on traditional diagnostic techniques. Therefore, there is a need for alternative non-invasive methods that can accurately detect blood cancer in its early stages. In conclusion, those limitations necessitate the development of improved approaches that prioritize accuracy, non-invasiveness, and cost-effectiveness. Machine learning-based solutions offer promising prospects in addressing these needs, leveraging advanced computational techniques to enhance accuracy, enable non-invasive diagnostics, and optimize resource utilization.

Chapter 03: Literature Review

3.1. Introduction

In this chapter, we explore the extensive body of literature surrounding blood cancer detection and the application of machine learning in healthcare. We aim to uncover valuable insights, identify gaps in current research, and lay the groundwork for our proposed study. Let us begin by exploring the multifaceted world of blood cancer, the existing diagnostic methods, and the promising realm of machine learning in healthcare.

To bridge the gap between machine learning and blood cancer detection, we will review related studies and research exploring the application of machine learning algorithms in this domain. We will examine the effectiveness of various machine learning algorithms, such as support vector machines (SVM), decision trees, and Artificial Neural Networks (ANN), in developing models for blood cancer detection. Additionally, we will delve into the significance of feature selection and data preprocessing techniques, such as Principal Component Analysis (PCA), Recursive Feature Elimination (RFE), and normalization, in improving the performance of machine learning models. Moreover, we will explore the evaluation metrics used to assess the effectiveness of these models and compare their performance in different contexts.

3.2. Overview of Blood Cancer and Its Type

Blood cancer is a profound and intricate disease that affects the very essence of our life - our blood. It encompasses diverse cancers that arise from the blood cells, bone marrow, or lymphatic system. The hallmark feature of blood cancer is the uncontrolled proliferation of abnormal cells, which can disrupt the delicate balance within our bodies and impair the normal functioning of the blood and immune system. The prevalence of blood cancer is a matter of great concern globally. According to the World Health Organization[1] (WHO), blood cancers account for approximately 10% of all cancer cases worldwide, making them a significant health challenge impacting millions annually. The exact prevalence varies across different regions and populations. However, blood cancer is a formidable adversary that affects individuals of all ages, from children to the elderly, Figure 3. 1.

One of the reasons blood cancer is particularly challenging is its diverse nature. There are several distinct types of blood cancer, each with unique characteristics and disease progression. The most common types include leukaemia, lymphoma, and myeloma. Leukaemia involves the abnormal growth of white blood cells, lymphoma affects the lymphatic system, and myeloma affects plasma cells in the bone marrow. Further subtypes and variations within each type add to the complexity of diagnosis and treatment.

Figure 3. 1 Global prevalence of blood cancer [6]

[1] https://www.who.int/

The impact of blood cancer extends beyond the physical symptoms experienced by individuals. It can profoundly affect their emotional well-being, relationships, and overall quality of life. A blood cancer patient's journey often involves many medical interventions, including chemotherapy, radiation therapy, stem cell transplantation, and targeted therapies. The multidisciplinary approach required for managing blood cancer highlights the urgent need for accurate and timely diagnosis.

In the following sections of this chapter, we will explore the current diagnostic methods employed in detecting blood cancer. We will delve into the world of blood tests and imaging studies, which serve as valuable tools for detecting abnormalities in blood cells and identifying signs of blood cancer. Furthermore, we will discuss the limitations and challenges associated with these diagnostic methods, paving the way for exploring machine learning as a potential solution. By combining the power of advanced technology with the complexities of blood cancer detection, we can strive towards more accurate, efficient, and accessible diagnostic approaches that can ultimately improve patient outcomes and save lives.

3.2.1. Types of blood cancer

The realm of blood cancer is vast and encompasses a spectrum of diseases, each with its intricacies and characteristics. Understanding the different types of blood cancer is crucial for accurate diagnosis and effective treatment. Let us explore the primary types of blood cancer and their diverse subtypes, shedding light on the specific cells affected and the stage of the disease.The realm of blood cancer is vast and encompasses a spectrum of diseases, each with its intricacies and characteristics. Understanding the different types of blood cancer is crucial for accurate diagnosis and effective treatment. Let us explore the primary types of blood cancer and their diverse subtypes, shedding light on the specific cells affected and the stage of the disease.

1. *Leukaemia*: A formidable adversary originating in the bone marrow, the excessive production of abnormal white blood cells marks leukaemia. This uncontrolled proliferation disrupts the delicate balance between the bone marrow and bloodstream, impeding the normal functioning of the immune system. Leukaemia can be classified into four main subtypes, each with its distinct features :

- *Acute Lymphoblastic Leukemia (ALL):* Primarily affecting children, this aggressive form of leukaemia involves the overproduction of immature lymphocytes.
- Acute Myeloid Leukemia (AML): Common in adults, AML is characterized by the rapid growth of abnormal myeloid cells that crowd out healthy blood cells.
- *Chronic Lymphocytic Leukemia (CLL):* Predominantly occurring in older adults, CLL involves the gradual accumulation of abnormal lymphocytes that mature slowly.
- *Chronic Myeloid Leukemia (CML):* This form of leukaemia is marked by the overproduction of abnormal myeloid cells, often detected during routine blood tests.

Each subtype presents challenges and requires tailored treatment strategies, underscoring the complexity of leukaemia management.

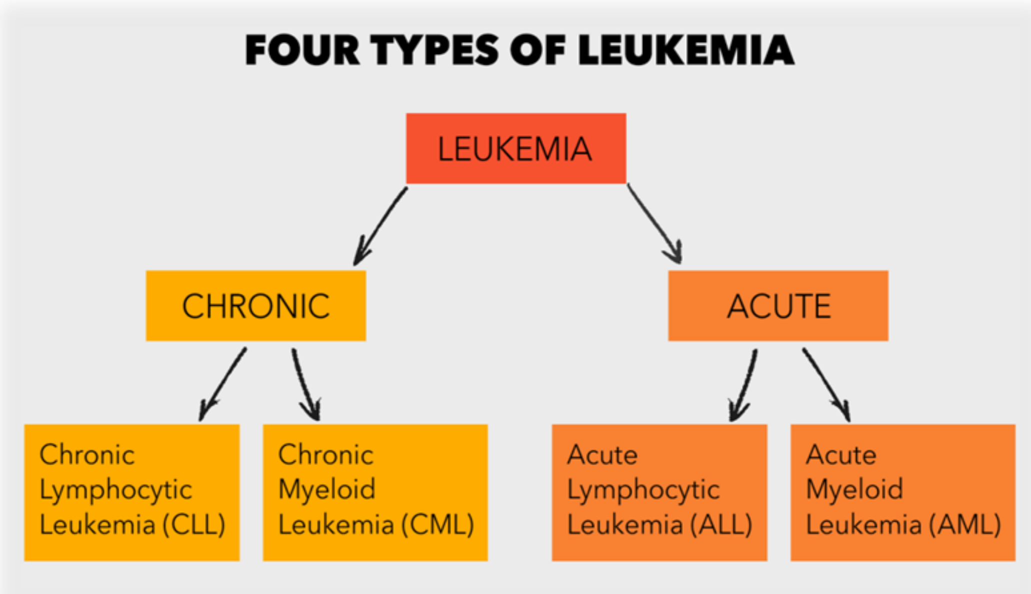


Figure 3. 2 Subtypes of leukaemia [7]

2. *Lymphoma:* Nestled within the intricate network of the lymphatic system, lymphomas are a group of cancers that affect the lymphocytes, a type of white blood cell crucial for immune function. Lymphoma can be broadly categorized into two main types :

- *Hodgkin Lymphoma (HL):* Named after the physician who first described it, Hodgkin lymphoma is characterized by the presence of Reed-Sternberg cells, large abnormal cells that reside in the lymph nodes. This type of lymphoma often spreads predictably from one group of lymph nodes to another.
- *Non-Hodgkin Lymphoma (NHL):* Diverse, non-Hodgkin lymphoma encompasses many subtypes, each originating from different types of lymphocytes. The behaviour and aggressiveness of non-Hodgkin lymphoma vary, making it a complex and heterogeneous disease.

These distinctions in subtypes and behaviour highlight the need for precise diagnostic techniques and tailored treatment approaches in lymphoma.

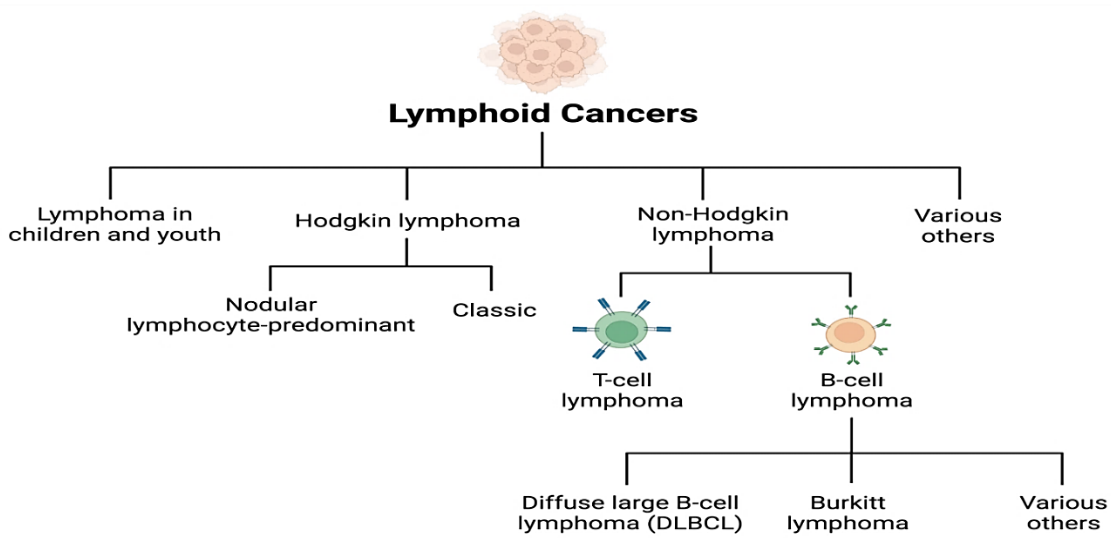


Figure 3. 3 Subtypes of lymphoma [8]

3. *Myeloma:* Also known as multiple myeloma, this type of blood cancer manifests in the bone marrow, affecting plasma cells. Plasma cells play a crucial role in producing antibodies that defend against infections. In myeloma, abnormal plasma cells multiply uncontrollably, leading to the accumulation of these cells and the formation of tumors within the bone marrow, Figure 3. 4. This disease can cause bone pain, fractures, and various complications. The progression of myeloma involves various stages, from the

initial asymptomatic stage to advanced symptomatic disease, necessitating comprehensive and specialized care.

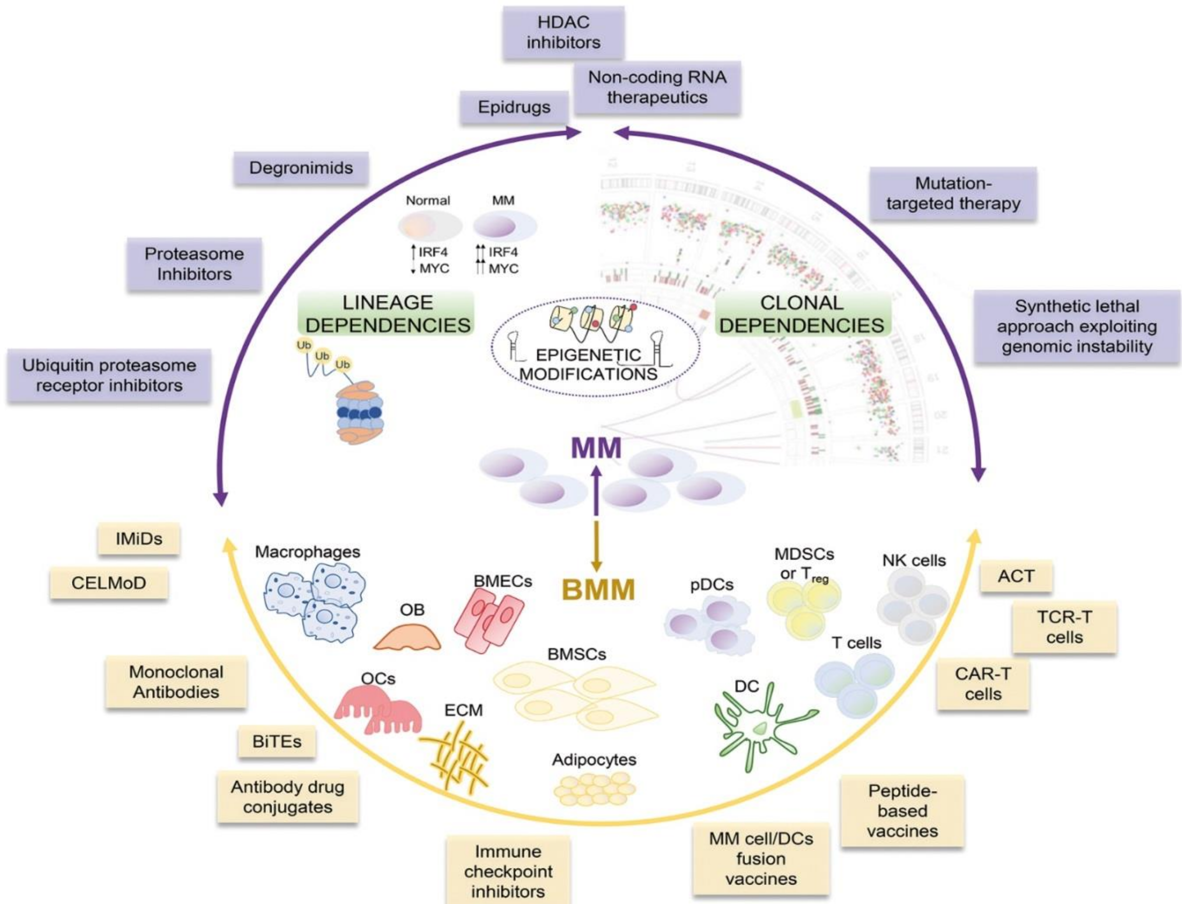


Figure 3. 4 Progression of myeloma [9]

By unravelling the intricacies of the different types and subtypes of blood cancer, we gain a deeper understanding of the complexity and challenges faced in diagnosis and treatment. As we move forward in this chapter, we will explore the current diagnostic methods and delve into the potential of machine learning to enhance the detection and management of blood cancer. By integrating advanced technologies and a comprehensive understanding of the diverse aspects of blood cancer, we can strive towards improved outcomes and a brighter future for individuals battling this formidable disease.

3.2.2. Risk factors and symptoms

In the complex realm of blood cancer, various risk factors and symptoms contribute to the intricate tapestry of this disease. Understanding these factors and recognizing the associated symptoms are crucial in early detection and timely intervention. Let us explore the multifaceted landscape of risk factors and symptoms in blood cancer:

1. *Genetic predisposition:* The intricate dance between genetics and blood cancer has captivated researchers for decades. Specific inherited genetic mutations have been identified as risk factors for developing blood cancer. These mutations can disrupt the delicate balance of cellular processes, increasing the susceptibility to the disease. Genetic mutations associated with blood cancer include mutations in genes such as TP53, BCR-ABL1, and MYC, among others. These mutations can serve as molecular fingerprints, aiding in identifying and understanding specific blood cancer subtypes.
2. *Exposure to certain chemicals:* Our environment can profoundly impact our health, and blood cancer is no exception. Prolonged exposure to certain chemicals has been linked to an increased risk of developing blood cancer. Substances such as benzene, pesticides, certain solvents, and industrial chemicals have been identified as potential culprits. These chemicals can infiltrate the body through various routes, including inhalation, ingestion, or absorption through the skin. Understanding occupational and environmental exposures and their potential impact on blood cancer risk is vital for preventive measures and for creating a safer environment [10].
3. *Weakened immune system:* Our immune system serves as a formidable shield, protecting us from many threats. However, a compromised immune system can leave the body vulnerable to various diseases, including blood cancer. Individuals with conditions such as HIV/AIDS or those who have undergone organ transplantation face a higher risk of developing blood cancer. The delicate interplay between immune function and blood cancer highlights the importance of comprehensive healthcare for individuals with compromised immune systems [11].

Recognizing the signs and symptoms of blood cancer is pivotal for early detection and timely intervention. While these symptoms may indicate other medical conditions, they should not be overlooked, and a thorough examination by a healthcare professional is necessary for an accurate diagnosis. Common symptoms of blood cancer include, Figure 3. 5:

- *Fatigue*: Persistent and unexplained exhaustion that impacts daily activities.
- *Unexplained weight loss:* Significant and unintentional weight loss without any apparent cause.
- *Frequent infections*: Recurrent infections, such as respiratory or urinary tract infections, that may indicate compromised immune function.
- *Fever*: Persistent or recurring fever without any apparent cause.
- *Night sweats:* Excessive sweating during sleep, soaking through clothes and beddings.
- *Easy bruising or bleeding:* Unexplained bruises or bleeding, such as nosebleeds or bleeding gums.

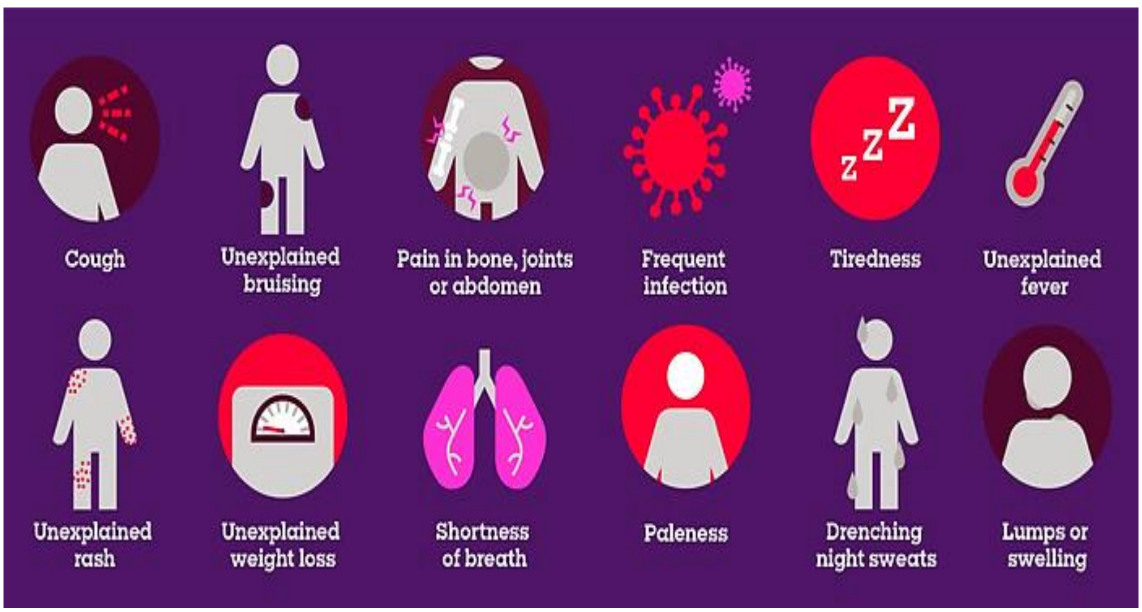


Figure 3. 5 Common symptoms of blood cancer. https://bloodcancer.org.uk/understanding-blood-cancer/blood-cancer-signs-symptoms/

These sympt roviders to delve deeper and investigate further. Through vigilance and awareness, we can empower individuals to seek timely medical attention and embark on a path towards diagnosis, treatment, and recovery [11].

As we journey further into this chapter, we will explore the current diagnostic methods for detecting blood cancer. Shedd light on their limitations and the potential for innovative approaches, such as machine learning, to revolutionize the field. By merging scientific advancements with a comprehensive understanding of the risk factors and symptoms, we strive towards a future where blood cancer can be detected early, managed effectively, and, ultimately, overcome.

3.3. Current Diagnostic Methods for Blood Cancer Detection

3.3.1. Blood tests and imaging studies

In the context of blood cancer detection, blood tests and imaging studies play a crucial role in identifying abnormalities and providing valuable information for diagnosis. Blood tests, including Complete Blood Count (CBC) and blood chemistry tests, are commonly performed to analyze the composition of blood cells and detect any irregularities. Medical professionals can identify potential indications of blood cancers by examining the number and types of blood cells present. Deviations from the typical ranges of these cells can indicate the presence of blood disorders or malignancies, prompting further investigation.

Imaging studies such as X-rays, Computed Tomography (CT) scans, and Magnetic Resonance Imaging (MRI) are additional tools used in blood cancer detection. These imaging techniques allow for detailed visualization of the body's internal structures, including lymph nodes, which are often affected by blood cancers. Enlarged lymph nodes can indicate abnormal cell growth or cancerous activity. For example, Figure 3. 6, **Error! Reference source not found.**illustrates a CT scan that shows enlarged lymph nodes, which may raise suspicion of blood cancer. These imaging studies provide valuable information about the location, size, and characteristics of lymph nodes, aiding in diagnosing and staging blood cancers [12].

By combining the insights gained from blood tests and imaging studies, healthcare professionals can comprehensively understand the patient's condition. These diagnostic methods are crucial for detecting blood cancers, enabling early intervention and appropriate treatment planning.

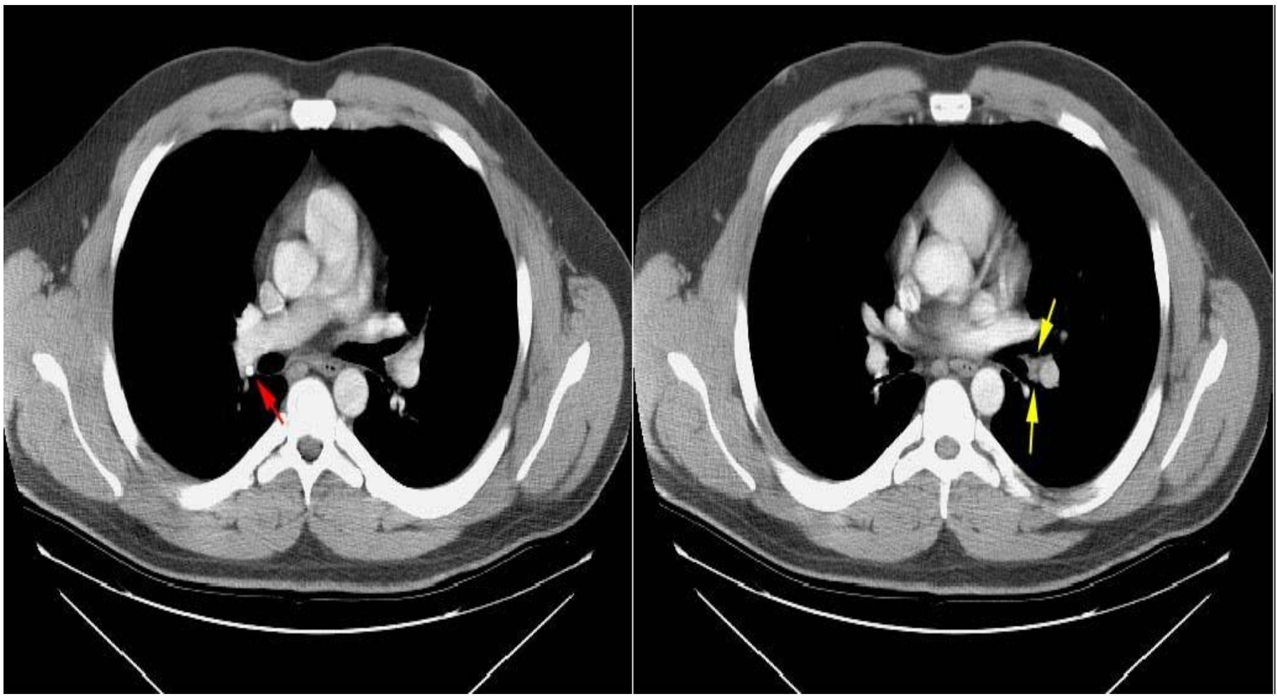


Figure 3. 6 Example of a CT scan showing enlarged lymph nodes [12].

It is important to note that while blood tests and imaging studies are valuable diagnostic tools, they have limitations. They may not provide a definitive diagnosis on their own. They are often used with other diagnostic procedures, such as biopsies and cytogenetic analysis, to confirm the presence of blood cancer and identify specific genetic abnormalities.

3.3.2. Biopsies and cytogenetic analysis

In the field of blood cancer detection, biopsies and cytogenetic analysis are vital diagnostic methods that provide crucial insights into the nature of the disease. Biopsies involve the collection of tissue samples from specific sites, such as the bone marrow and lymph nodes, for detailed examination and analysis. These samples allow healthcare professionals to understand the suspected blood cancer's cellular composition and genetic characteristics.

Bone marrow biopsies, as depicted in Figure 3. 7 involve the extraction of a small sample of bone marrow tissue, typically from the hip bone or sternum, using a specialized needle. The bone marrow is responsible for producing blood cells, and abnormalities in this tissue can indicate various blood disorders, including cancers. By analyzing the cellular components and structure of the bone marrow, medical professionals can assess the presence of abnormal or cancerous cells, providing important diagnostic information.

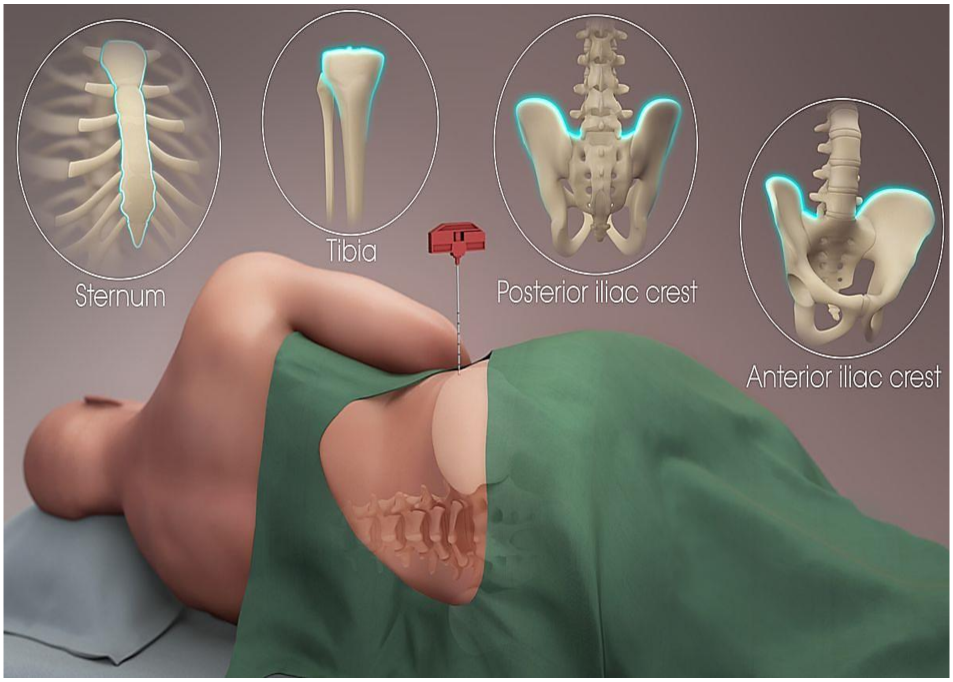
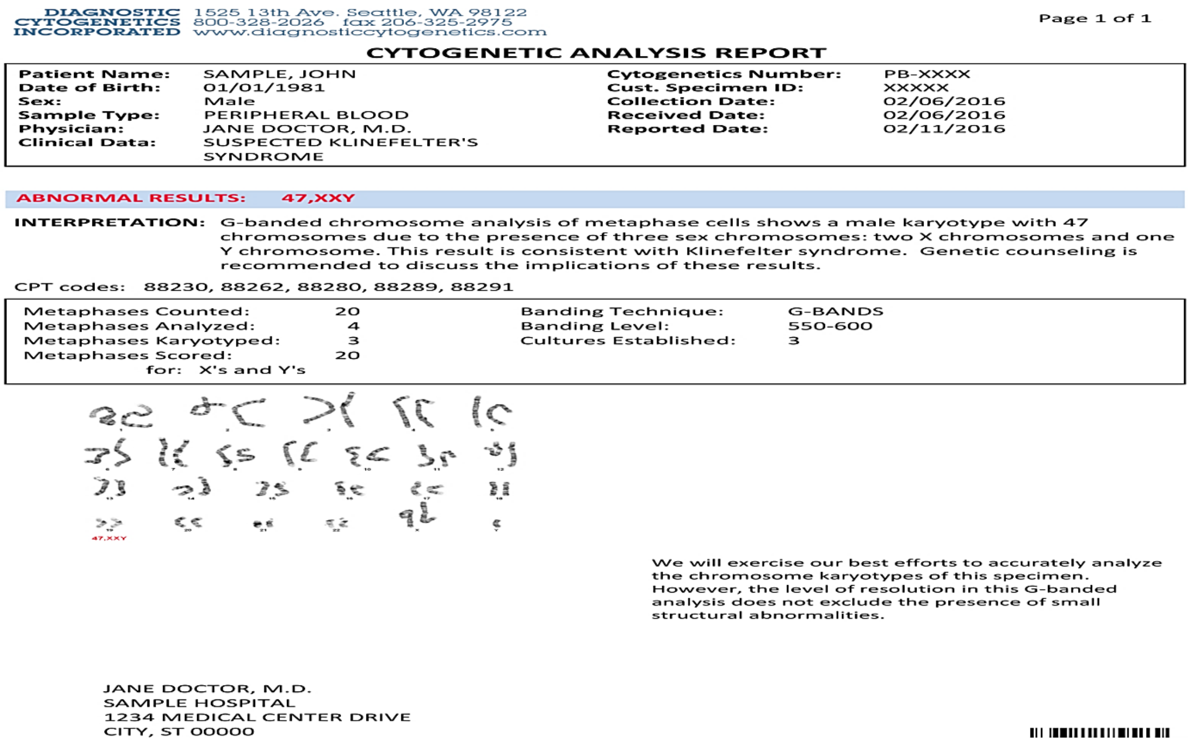


Figure 3. 7 Bone marrow biopsy procedure.
https://www.americanmedicalcoding.com/cpt-code-bone-marrow-biopsy-aspiration/

Lymph node biopsies are another biopsy commonly used to evaluate blood cancers. Lymph nodes, part of the lymphatic system, play a crucial role in filtering and trapping abnormal cells, including cancer cells. A biopsy involves the removal of a lymph node or a part of it for microscopic examination. This procedure helps determine the presence and extent of cancerous involvement in the lymph nodes, aiding in staging and treatment planning.

Cytogenetic analysis, as shown in Figure 3. 8, is a specialized laboratory technique that focuses on studying the chromosomes within cancer cells. This analysis involves examining the structure, number, and arrangement of chromosomes to identify specific genetic abnormalities associated with blood cancer. By detecting alterations in the DNA sequence or chromosomal rearrangements, cytogenetic analysis provides valuable information about the underlying genetic factors contributing to the development and progression of blood cancers.



DIAGNOSTIC CYTOGENETICS INCORPORATED 1525 13th Ave. Seattle, WA 98122 800-328-2026 fax 206-325-2975 www.diagnosticcytogenetics.com

Page 1 of 1

CYTOGENETIC ANALYSIS REPORT

Patient Name:	SAMPLE, JOHN	Cytogenetics Number:	PB-XXXX
Date of Birth:	01/01/1981	Cust. Specimen ID:	XXXXX
Sex:	Male	Collection Date:	02/06/2016
Sample Type:	PERIPHERAL BLOOD	Received Date:	02/06/2016
Physician:	JANE DOCTOR, M.D.	Reported Date:	02/11/2016
Clinical Data:	SUSPECTED KLINEFELTER'S SYNDROME		

ABNORMAL RESULTS: 47,XXY

INTERPRETATION: G-banded chromosome analysis of metaphase cells shows a male karyotype with 47 chromosomes due to the presence of three sex chromosomes: two X chromosomes and one Y chromosome. This result is consistent with Klinefelter syndrome. Genetic counseling is recommended to discuss the implications of these results.

CPT codes: 88230, 88262, 88280, 88289, 88291

Metaphases Counted:	20	Banding Technique:	G-BANDS
Metaphases Analyzed:	4	Banding Level:	550-600
Metaphases Karyotyped:	3	Cultures Established:	3
Metaphases Scored:	20		
for: X's and Y's			

47,XXY

We will exercise our best efforts to accurately analyze the chromosome karyotypes of this specimen. However, the level of resolution in this G-banded analysis does not exclude the presence of small structural abnormalities.

JANE DOCTOR, M.D.
SAMPLE HOSPITAL
1234 MEDICAL CENTER DRIVE
CITY, ST 00000

Figure 3. 8 Example of a cytogenetic analysis report [13]

The results of the cytogenetic analysis are often presented in the form of a cytogenetic analysis report, which outlines the specific chromosomal abnormalities identified in the cancer cells. This information helps understand the blood cancer's molecular characteristics, facilitating personalized treatment strategies and targeted therapies.

Biopsies and cytogenetic analysis serve as complementary diagnostic techniques in blood cancer detection. While biopsies provide direct tissue samples for examination, cytogenetic analysis delves into the genetic profile of the cancer cells. Together, these methods aid in confirming the presence of blood cancer, understanding its genetic basis, determining the stage and prognosis, and guiding the selection of appropriate treatment options for patients.

3.4. Overview of Machine Learning and Its Applications in Healthcare

In machine learning, it is essential to understand the definition and various types of this powerful subset of artificial intelligence. By examining the development of algorithms capable of learning from data and making predictions or decisions, we can grasp its significance in healthcare, particularly in blood cancer detection. Machine learning involves creating and utilising algorithms that learn from data to make predictions or decisions without explicit programming. These algorithms can make informed decisions based on new, unseen data by analysing patterns and relationships within the data.

Here, I will provide a structured overview of machine learning's primary types Figure 3. 9:

Supervised learning: Supervised learning is the most commonly used type of machine learning. This approach trains algorithms using labelled data, where each data point is associated with a known target or output label. The algorithm learns to map input data to the correct output labels by minimizing the discrepancy between its predictions and the true labels provided in the training data. Supervised learning is suitable for classification, regression, and anomaly detection tasks.

1. *Supervised learning:* Supervised learning is the most commonly used type of machine learning. This approach trains algorithms using labelled data, where each data point is associated with a known target or output label. The algorithm learns to

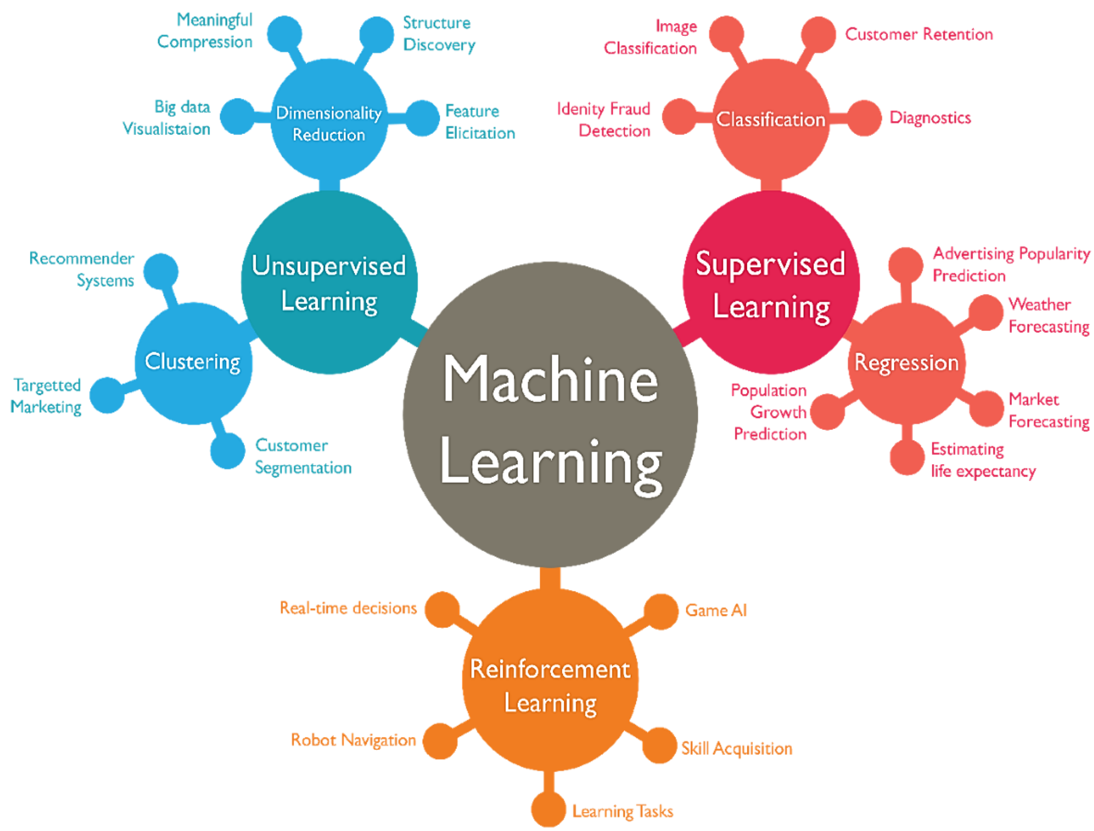


Figure 3. 9 Types of machine learning

https://towardsdatascience.com/machine-learning-types-2-c1291d4f04b1

map input data to the correct output labels by minimizing the discrepancy between its predictions and the true labels provided in the training data. Supervised learning is suitable for classification, regression, and anomaly detection tasks.

2. *Unsupervised learning:* Unsupervised learning deals with unlabeled data. Instead of relying on known output labels, unsupervised learning algorithms aim to identify inherent patterns or structures within the data. Clustering and dimensionality reduction techniques are commonly used in unsupervised learning to discover hidden relationships or groupings in the data. This type of learning provides valuable insights into the structure and organization of the data.
3. *Reinforcement learning:* Reinforcement learning involves an agent interacting with an environment and learning to make decisions based on rewards or punishments.

The agent explores different actions and receives feedback from the environment through rewards or penalties. Through trial and error, the agent aims to maximize the cumulative reward by learning the optimal sequence of actions in a given environment. Reinforcement learning has shown promise in healthcare applications, such as treatment optimization and resource allocation.

Understanding these different types of machine learning is crucial for harnessing their potential in healthcare, particularly in blood cancer detection. Supervised learning algorithms can be trained on labeled datasets to develop predictive models that identify potential blood cancers based on patient data. Unsupervised learning techniques can uncover hidden patterns or subgroups within blood cancer data, enabling more personalized and targeted treatments. Reinforcement learning, although less commonly applied in healthcare, has the potential to optimize treatment strategies and improve patient outcomes.

In summary, machine learning involves the development of algorithms that learn from data to make predictions or decisions. The three main types of machine learning- supervised, unsupervised, and reinforcement learning-offer are distinct approaches to analyzing and extracting insights from data. Understanding these types is critical for unlocking the potential of machine learning in healthcare, including its application in blood cancer detection.

3.5. Applications of machine learning in healthcare

Machine learning has revolutionized the healthcare industry by offering innovative solutions in various domains. Its application in diagnostics, treatment planning, drug discovery, and personalized medicine has garnered significant attention due to its potential to enhance patient outcomes, reduce costs, and optimize the overall efficiency of healthcare systems. Let us delve into these applications and explore the impact of machine learning in healthcare, Figure 3. 10.

Figure 3. 10 Examples of machine learning applications in healthcare [14]

1. *Diagnostics*: Machine learning algorithms have improved diagnostic accuracy and efficiency. By analyzing medical images, such as X-rays, CT scans, and MRIs, machine-learning models can aid in detecting abnormalities, including tumors, lesions, and other indicators of diseases. These algorithms can quickly analyze vast amounts of imaging data and assist healthcare professionals in making accurate and timely diagnoses.
2. *Treatment planning:* Machine learning techniques provide personalized recommendations and predictions in treatment planning. Machine learning models can identify patterns and factors that influence treatment responses by analyzing patient data, including medical records, genetic information, and treatment outcomes. This information enables healthcare providers to tailor treatment plans to individual patients, optimizing therapeutic strategies and improving treatment outcomes.
3. *Drug discovery:* Machine learning is transforming the drug discovery process by accelerating the identification and development of potential new drugs. By leveraging large-scale biological and chemical data, machine learning models can predict the

effectiveness of compounds, identify potential targets for drug intervention, and optimize drug properties. Drug discovery enables researchers to streamline the drug discovery pipeline, reducing the costs and time required to bring new treatments to the market.

4. *Personalized medicine:* Personalized medicine aims to deliver targeted healthcare interventions based on an individual's unique characteristics, such as genetics, lifestyle, and medical history. Machine learning algorithms contribute significantly to personalized medicine by analyzing complex datasets and generating patient-specific insights.

These insights can inform treatment decisions, predict disease progression, and identify individuals at higher risk of developing certain conditions. By tailoring interventions to individual needs, personalized medicine improves patient outcomes and reduces unnecessary treatments or interventions.

3.6. Review of Related Studies and Research on Machine Learning for Blood Cancer Detection

Machine learning algorithms have emerged as powerful tools for improving diagnostic accuracy and early disease identification in blood cancer detection. Various machine learning algorithms, including support vector machines (SVM), decision trees, and Artificial Neural Networks (ANN), have been extensively explored to develop predictive models specifically designed for blood cancer detection.

1. *Support Vector Machines (SVM):* SVM is a widely used machine learning algorithm that excels in classification tasks. SVM constructs a hyperplane in a high-dimensional feature space, which helps separate different classes of data points. In the context of blood cancer detection, SVM algorithms utilize patient data, such as blood cell counts, genetic markers, and clinical information, to classify individuals as either having blood cancer or being healthy. SVM algorithms have demonstrated high

accuracy and robust performance in distinguishing between cancerous and non-cancerous samples.

2. *Decision Trees:* Decision trees are intuitive machine-learning models that mimic human decision-making processes. These models use a hierarchical structure of nodes and branches to classify data. Decision tree algorithms analyze patient data in blood cancer detection to build a tree-like model that captures important features and decision rules. By traversing the decision tree, these algorithms can effectively classify patients based on their risk of having blood cancer. Decision trees offer interpretability and can provide insights into the factors influencing the classification outcomes.
3. *Artificial Neural Networks (ANN):* Artificial Neural Networks, inspired by the structure and function of the human brain, are powerful machine learning models capable of learning complex patterns and relationships. ANN algorithms consist of interconnected layers of artificial neurons that process and transform input data. In the context of blood cancer detection, ANN models can analyze diverse patient data, including genetic profiles, medical history, and laboratory results, to predict the likelihood of blood cancer. ANN algorithms have shown promising results in terms of accuracy, sensitivity, and specificity, making them valuable tools in the early detection of blood cancer

These machine-learning algorithms for blood cancer detection have demonstrated their potential to enhance diagnostic capabilities and contribute to early intervention. By leveraging the power of these algorithms, healthcare professionals can make more informed decisions, leading to timely treatments and improved patient outcomes. However, it is essential to note that the performance of these algorithms relies heavily on the quality and representativeness of the training data, as well as the selection of relevant features and appropriate model optimization techniques. The ongoing advancements in machine learning techniques, coupled with the integration of large-scale datasets and genomic information, hold great promise for further improving the accuracy and efficiency of blood cancer detection. Continued research and development in this field will likely lead to more sophisticated algorithms and personalized

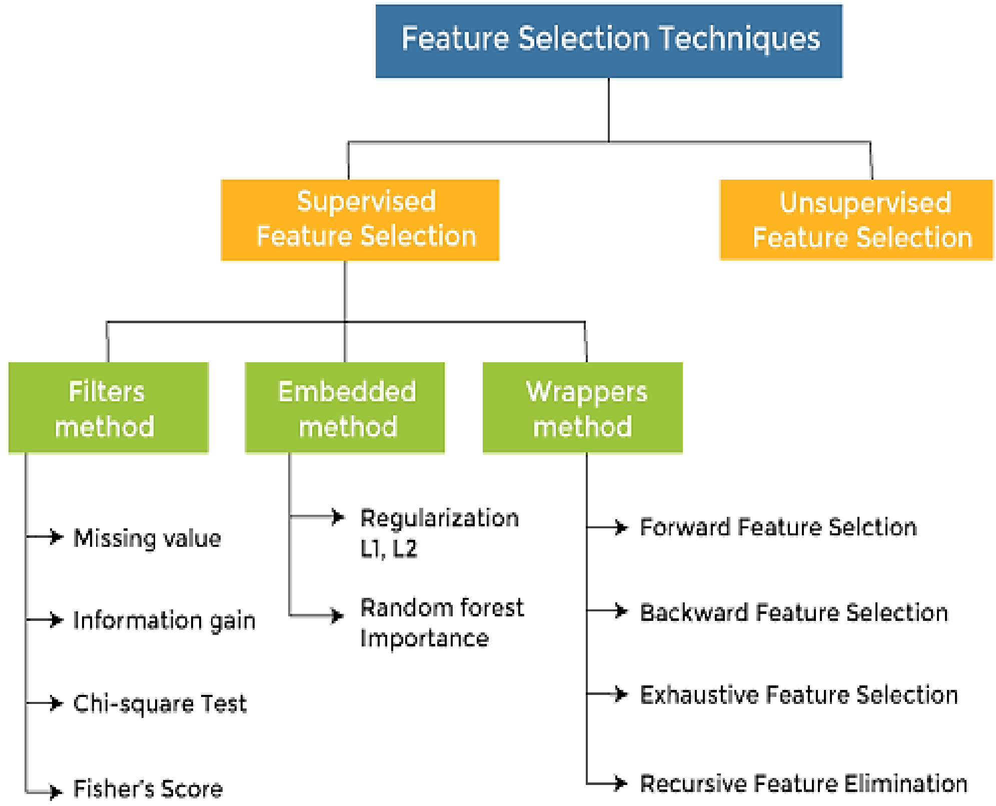


Figure 3. 11 . 11 Examples of feature selection and data preprocessing techniques [14].

approaches, ultimately benefiting patients by enabling early diagnosis, targeted treatments, and improved overall management of blood cancer.

Moreover, in machine learning for blood cancer detection, the utilization of appropriate feature selection and data preprocessing techniques holds paramount importance in optimizing the performance and accuracy of predictive models, Figure 3. 11. These techniques, such as PCA, Recursive Feature Elimination (RFE), and normalization, are instrumental in preparing the input data and extracting relevant information from complex datasets, ultimately contributing to more effective blood cancer detection.

1. *Principal Component Analysis:* PCA is a widely employed technique that aids in reducing the dimensionality of high-dimensional datasets while retaining important information. By transforming the original features into a new set of uncorrelated

variables known as principal components, PCA helps capture the maximum variance in the data. In blood cancer detection studies, PCA can be applied to identify the most informative features or to compress the data, facilitating more efficient analysis and enhancing the performance of machine learning models.

2. *Recursive Feature Elimination (RFE):* RFE is a feature selection technique that iteratively removes irrelevant or redundant features from the dataset. This method assigns weights to each feature and eliminates the features with the lowest weights in each iteration. By progressively discarding less informative features, RFE helps identify a subset of features that contribute significantly to the prediction of blood cancer. This process improves computational efficiency and reduces the risk of overfitting, thereby enhancing the generalization capability of machine learning models.
3. *Normalization*: Is a crucial data preprocessing technique that aims to bring all features onto a standardized scale. Therefore, it ensures that no single feature dominates the learning process due to differences in their respective scales or units. Normalization techniques such as min-max scaling or z-score normalization are commonly applied to normalize numerical features in blood cancer detection. By normalizing the data, machine learning models can effectively compare and weigh the impact of different features, leading to more reliable and accurate predictions.

By employing these feature selection and data preprocessing techniques, researchers and practitioners can enhance the performance of machine learning models in blood cancer detection. These techniques help mitigate the effects of noise, improve the interpretability of models, and increase computation efficiency. However, it is important to note that the selection and application of these techniques should be done carefully, considering the dataset's specific characteristics and the analysis's objectives.

Continued research and advancements in feature selection and data preprocessing techniques are vital for refining the accuracy and reliability of blood cancer detection models. As datasets become more diverse and complex, novel techniques and algorithms may emerge, enabling more sophisticated data preparation and feature selection strategies. Ultimately, successfully integrating these techniques will contribute to more precise and effective blood cancer detection, facilitating early intervention and improving patient outcomes.

3.7. Evaluation metrics and performance comparison

When evaluating the performance of machine learning models for blood cancer detection, a range of evaluation metrics is employed to measure their effectiveness objectively. These metrics, including accuracy, precision, recall, and F1-score, provide valuable insights into the model's predictive capabilities and ability to identify blood cancer cases accurately. Using these evaluation metrics, researchers can compare and contrast different algorithms to identify the most suitable approach for specific contexts.

3.7.1. Accuracy

Accuracy is a fundamental evaluation metric that measures the overall correctness of the model's predictions. It calculates the proportion of correctly classified instances (both true positives and true negatives) out of the total number of instances. A higher accuracy score indicates a higher level of correctness in the model's predictions. However, more than accuracy is required to provide a complete picture of the model's performance, especially when dealing with imbalanced datasets or when the cost of misclassification varies for different classes.

3.7.2. Precision

Precision focuses on the proportion of true positive predictions out of all positive predictions made by the model. It represents the model's ability to accurately identify positive instances without including false positives. Precision is particularly important in scenarios where the cost of false positives is high, such as in blood cancer detection, where misdiagnosis can have severe consequences. A higher precision score indicates a lower rate of false positives, signifying the model's ability to avoid false alarms.

3.7.3. Recall:

Recall, also known as sensitivity or true positive rate, measures the model's ability to identify positive instances out of all actual positive instances correctly. It calculates the proportion of true positive predictions concerning the total number of actual positive instances [17]. A recall is especially crucial when missing positive cases (false negatives) is undesirable. A higher recall score implies a lower rate of false negatives in blood cancer detection, indicating the model's effectiveness in detecting blood cancer cases.

3.7.4. F1-score:

The F1-score combines precision and recall into a single metric, providing a balanced measure of the model's performance. It calculates the harmonic mean of precision and recall, comprehensively evaluating the model's ability to balance true positives, false positives, and false negatives. The F1-score is particularly useful when the dataset is imbalanced or when an uneven cost is associated with different types of misclassifications. A higher F1-score indicates a better trade-off between precision and recall, highlighting the model's overall performance.

By utilizing these evaluation metrics, researchers can assess and compare the performance of different machine learning algorithms in blood cancer detection studies. It allows for objectively analysing their strengths and weaknesses in specific contexts. However, it is essential to consider the limitations and assumptions associated with each metric and choose the most appropriate ones based on the specific goals and requirements of the study. Continued research in evaluation metrics and performance comparison is crucial for advancing the field of blood cancer detection. New metrics and techniques may emerge, catering to the evolving needs and challenges in diagnosing and treating blood cancer. By employing rigorous evaluation methodologies, researchers can refine and enhance the accuracy of machine learning models, ultimately improving the detection and management of blood cancer for better patient outcomes.

3.8. Limited research on machine learning-based web applications for blood cancer detection

The field of machine learning has made remarkable progress in the detection of blood cancer. Numerous studies have explored the potential of machine learning algorithms in accurately identifying blood cancer and improving patient outcomes. However, one area that still needs to be explored is the development of web applications that integrate these machine-learning models for practical use in real-world scenarios.

The existing research primarily focuses on developing and evaluating machine learning models using various algorithms, feature selection techniques, and evaluation metrics. These studies demonstrate the effectiveness of machine learning in blood cancer detection, showcasing its potential to enhance diagnostic accuracy and facilitate timely interventions. However, translating these models into user-friendly web applications that healthcare professionals and patients can readily access is an area that requires more attention.

Given the limitations mentioned above, there is a pressing need to develop improved diagnostic methods for blood cancer detection. These improved methods should address the following key requirements:

3. *Accuracy*: It is essential to enhance the accuracy of diagnostic methods to ensure reliable detection of blood cancer. By minimizing false positives and false negatives, more accurate methods can enable early and accurate diagnosis, leading to timely intervention and improved patient outcomes.
4. *Non-invasiveness:* To alleviate patient discomfort and minimize associated risks, non-invasive diagnostic approaches are highly desirable. Non-invasive methods that utilize advanced imaging techniques or analyze biomarkers from blood samples can provide valuable information without invasive procedures such as biopsies.
5. *Cost-effectiveness:* Developing cost-effective diagnostic methods is crucial to ensure their accessibility to a wider population. By reducing the financial burden associated with blood cancer detection, more individuals can benefit from timely and accurate diagnoses, leading to better treatment outcomes.

6. *The Potential of Machine Learning-based Solutions:* Machine learning-based solutions hold significant promise in addressing the need for improved diagnostic methods in blood cancer detection. By leveraging the power of computational algorithms and data analysis, machine learning models can effectively analyze complex patterns and identify subtle indications of blood cancer in medical data.
7. *Enhanced Accuracy:* Machine learning algorithms can learn from large datasets and extract meaningful insights, enabling them to identify patterns and features that may not be apparent to human observers. By integrating machine learning into diagnostic methods, the accuracy of blood cancer detection can be significantly improved, leading to more reliable diagnoses and personalized treatment plans.
8. *Non-invasive Approaches:* Machine learning techniques can be applied to non-invasive diagnostic modalities such as medical imaging, where they can analyze imaging data to identify specific patterns associated with blood cancer. Machine learning-based approaches can enhance patient comfort and safety during the diagnostic process by eliminating the need for invasive procedures.
9. *Cost-effectiveness:* Machine learning algorithms can contribute to cost-effective diagnostic methods by streamlining data analysis and interpretation, reducing the need for manual labor and expensive laboratory tests. Thus, it can lower the overall cost of blood cancer detection, making it more accessible to a broader population.

3.8.1. The Need for Improved Diagnostic Methods:

Web applications have become integral to modern healthcare systems, offering convenience, accessibility, and scalability. They enable healthcare providers to deliver timely and personalized patient care while facilitating remote consultations and monitoring. Incorporating machine learning models into web applications for blood cancer detection could revolutionize the diagnostic process, enabling healthcare professionals to make informed decisions based on accurate and real-time insights.

The limited research in this area suggests that the development of machine learning-based web applications for blood cancer detection is still in its infancy. The challenges associated with integrating complex algorithms, ensuring data security and privacy, and validating the performance and reliability of these applications in real-world settings need to be addressed. By bridging this gap, researchers and developers can unlock the potential of machine learning in blood cancer detection and bring the benefits of advanced diagnostics to a wider audience.

3.9. Conclusion

Despite advancements in blood cancer diagnostic methods, some limitations hinder their effectiveness. Current diagnostic methods are often invasive, leading to patient discomfort and reluctance. They are also time-consuming, causing delays in accurate diagnoses and treatment. Moreover, the high costs associated with these methods limit accessibility, and inaccurate results are possible. To improve blood cancer detection, non-invasive techniques like liquid biopsies and automated analysis algorithms can be developed to reduce invasiveness and streamline the diagnostic process. Overcoming these limitations is crucial for enhancing diagnostic accuracy, reducing patient burden, and improving overall outcomes in blood cancer detection. Machine learning has revolutionized healthcare by offering powerful applications in diagnostics, treatment planning, drug discovery, and personalized medicine. Its ability to analyze large datasets, identify patterns, and make accurate predictions has the potential to transform patient care. By harnessing the capabilities of machine learning, healthcare professionals can improve diagnoses, develop targeted treatment plans, accelerate drug discovery, and deliver personalized care, ultimately enhancing patient outcomes and advancing the efficiency of healthcare systems.

Chapter 04 : Methodology

4.1. Introduction

This chapter is a critical component of this memorandum, as it delves into the methodology behind developing a machine learning-based web application for blood cancer detection. By providing a comprehensive overview of the methodology, this chapter aims to shed light on the key steps and considerations involved in the development process.

To initiate the discussion, we first describe the study's dataset and the data preprocessing techniques employed. Therefore, readers clearly understand the foundation for the machine learning models. The dataset comprises blood cancer patient records, encompassing demographic information, clinical features, and laboratory test results. By sourcing the dataset from reputable sources such as cancer research centres or public health databases, the study ensures the reliability and relevance of the data. Moreover, we explain the machine learning algorithms utilized for blood cancer detection and their implementation. The selected algorithms, including SVM, decision trees, and ANN, offer distinct capabilities and advantages that make them suitable for this application. By providing an overview of each algorithm, readers gain insights into their underlying principles and how they contribute to the overall detection process. These algorithms are implemented using popular programming languages like Python and machine learning libraries such as scikit-learn or TensorFlow. By illustrating a Python code snippet, the chapter showcases the practical implementation of the machine learning models, encompassing crucial steps like data preprocessing, model training, and prediction.

To ensure the reliability and generalizability of the models, techniques such as cross-validation and holdout validation are employed [18]. These techniques help validate the models' performance on unseen data and mitigate the risk of overfitting. By employing rigorous performance evaluation techniques, the study aims to provide robust and reliable results that can be utilized confidently.

In conclusion, this chapter explores the methodology employed in developing the machine learning-based web application for blood cancer detection.

4.2. Description of the dataset and data preprocessing techniques

The dataset used in this study is a valuable resource consisting of blood cancer patient records. It contains comprehensive information, including demographic details, clinical features, and laboratory test results. These records are the foundation for training and evaluating the machine learning-based web application for blood cancer detection. The dataset was obtained from a reputable source, ensuring its reliability and accuracy. Possible sources for acquiring such a dataset include renowned cancer research centres or public health databases. These institutions maintain extensive collections of patient data, making them excellent sources for conducting in-depth research on blood cancer detection.

By leveraging this dataset, researchers gain access to a diverse range of information about blood cancer patients. Demographic information, such as age, gender, and ethnicity, provides valuable insights into potential risk factors and variations in disease prevalence across different populations. Clinical features, including symptoms, medical history, and comorbidities, comprehensively understand the patients' health status. Laboratory test results, such as blood counts, genetic markers, and molecular profiles, play a crucial role in diagnosing and monitoring blood cancer. Utilizing a dataset of this nature empowers researchers to develop a robust machine-learning model capable of accurately detecting blood cancer. Including diverse patient records allows the model to learn from various cases, capturing the intricacies and patterns associated with different blood cancer types. Moreover, the dataset's reputable origin enhances the credibility and generalizability of the findings.

4.2.1. Data preprocessing techniques

In developing a machine learning-based web application for blood cancer detection, data preprocessing techniques play a crucial role in ensuring the quality and reliability of the dataset. This section delves into the specific data preprocessing techniques employed in this study to enhance the performance of the machine learning models.

- *Data cleaning:* One of the initial steps in data preprocessing is data cleaning, which involves handling missing values, duplicates, and outliers in the dataset. Missing values can lead to biased or erroneous results, typically handled by removing the corresponding records or imputing values based on statistical techniques. Duplicates, if present, are identified and eliminated to avoid redundancy in the dataset. Outliers, which are extreme values that deviate significantly from the rest of the data, are either corrected or removed, as they can adversely affect the performance of the machine learning algorithms. Figure 4. 1 illustrates the data cleaning process, showcasing the steps involved and their impact on the dataset.

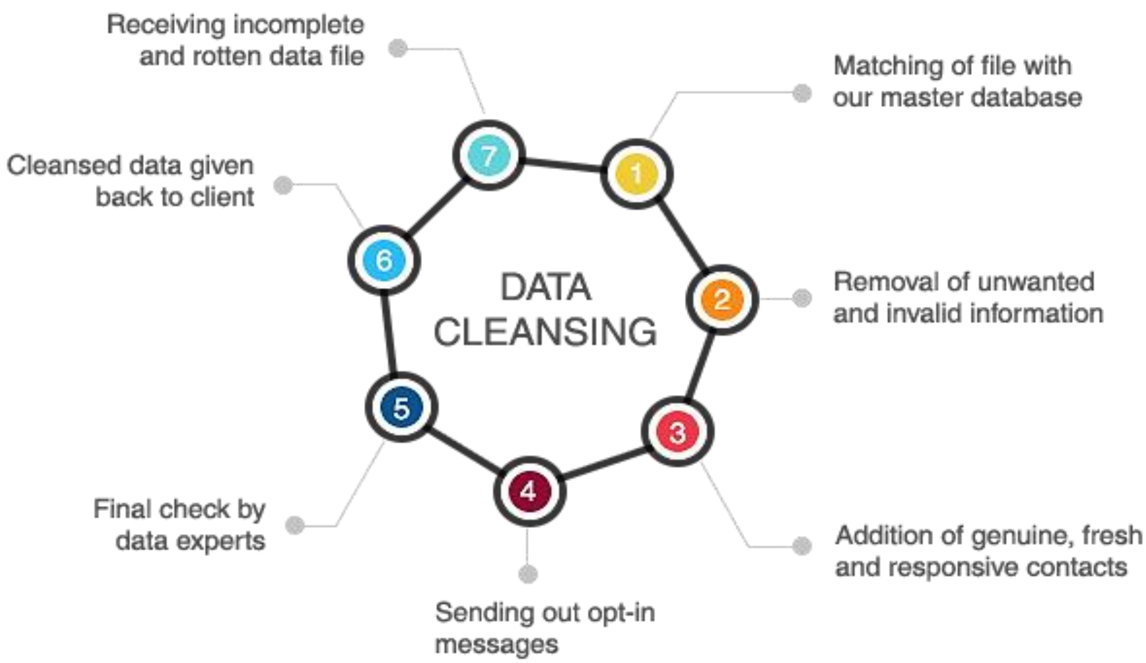


Figure 4. 1 Data cleaning process. https://hevodata.com/learn/data-cleansing/

4.3. Implementation of the algorithms

In the implementation phase of this study, the selected machine learning algorithms are translated into functional models by utilizing suitable programming tools and libraries. A widely adopted programming language like Python is chosen as the primary language for implementation. Python offers a rich ecosystem of libraries and frameworks specifically designed for machine learning tasks, making it a preferred choice among researchers and practitioners. In conjunction with Python, specialized machine learning libraries such as scikit-learn and TensorFlow are employed. These libraries provide a comprehensive set of functions, classes, and algorithms that expedite the implementation process and enable the seamless integration of machine learning models into the study.

Scikit-learn is a popular open-source library offering various machine learning algorithms, data preprocessing techniques, and model evaluation tools. It provides a user-friendly interface for implementing and fine-tuning various algorithms, making it an invaluable resource for researchers [16]. Furthermore, TensorFlow is a powerful library specifically designed for deep learning tasks and implementing artificial neural networks. With its computational graph abstraction, TensorFlow allows for efficient execution on both CPUs and GPUs, enabling the training and deployment of complex neural network models. By leveraging Python and specialized machine learning libraries like scikit-learn and TensorFlow, the researchers can effectively implement and experiment with the chosen algorithms. This combination of programming language and libraries streamlines the implementation process, reduces development time, and empowers researchers to harness the full potential of machine-learning techniques for blood cancer detection.

4.4. Web application architecture

In web application development, a well-structured architecture is crucial for successful implementation. This application is designed using client-server architecture, comprising distinct components that collaborate to deliver a seamless user experience and efficient functionality. The architecture can be explained as follows, Figure 4. 2 :

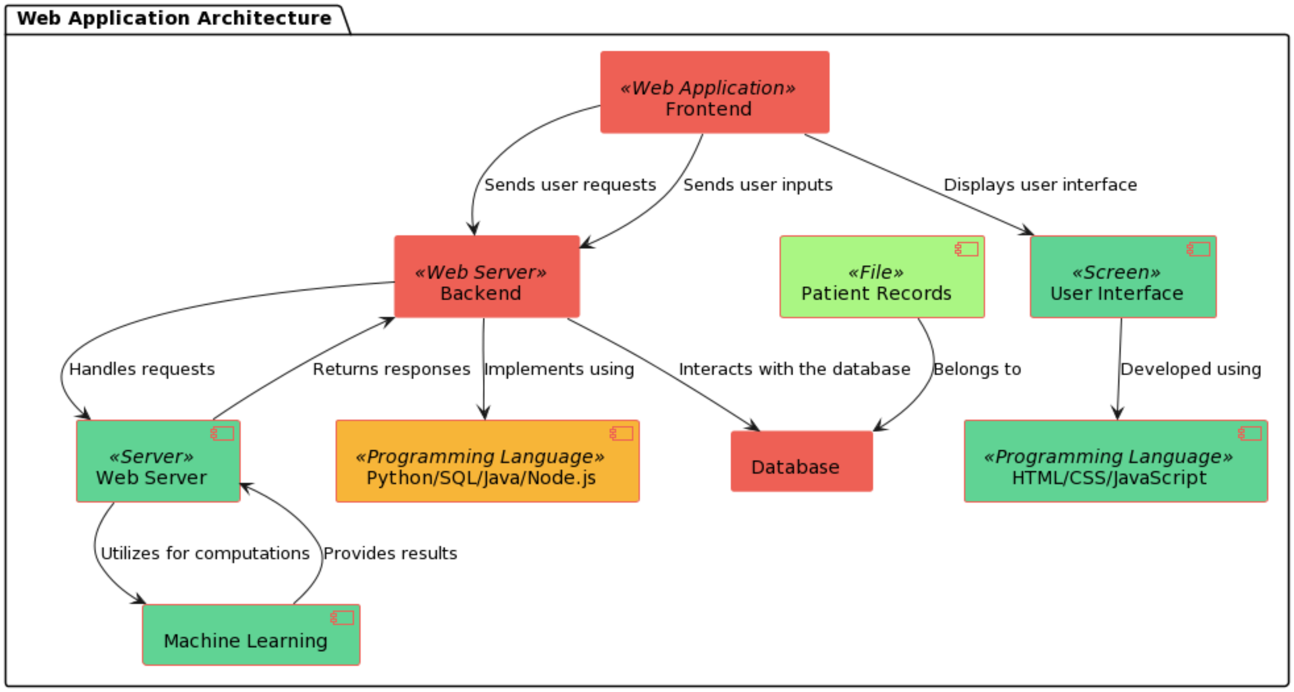


Figure 4. 2 Web application architecture.

1. *Front-end User Interface*: The front-end component of the web application focuses on the user-facing aspects, providing an interactive interface through which users can interact with the application. It incorporates intuitive design elements, user-friendly features, and responsive layouts to ensure a positive user experience. The front-end is typically developed using web technologies: HTML, CSS, JavaScript.
2. *Back-end Server:* As the backbone of the web application, the back-end component handles the processing and logic behind the scenes. It receives and processes user requests, interacts with the database, performs computations using machine learning algorithms, and delivers responses back to the front-end. The back-end is responsible for the application's core functionality, including data processing, model inference, and result generation. I used server-side programming languages: Python, Java, PHP, Node.js, and Flask framework.
3. Database: The web application incorporates a database system to store and manage patient records and related data. The database ensures secure storage, efficient retrieval, and proper data organisation, allowing for seamless access and retrieval of patient information when needed. I used MySQL database system.

4.4.1. Web application design

In the context of web application design, the focus is on creating a user-friendly and intuitive experience for users interacting with the application. Key design elements and features include:

1. *Login System:* The web application incorporates a secure login system to authenticate and authorize users. It ensures that only authorized individuals can access and interact with the application.
2. *Dashboard*: A central component of the application is the dashboard, which provides a visual representation of patient data. The dashboard presents relevant information in an organized and easily understandable manner, enabling users to gain insights and monitor patient records efficiently.
3. *Prediction Module:* The web application includes a prediction module specifically designed for blood cancer detection. Leveraging machine learning algorithms analyzes patient data and provides accurate predictions or risk assessments related to blood cancer. This module empowers healthcare professionals or users to make informed decisions based on the generated predictions.

The web application ensures an engaging and user-friendly experience by incorporating these design elements and features. The login system enhances security, the dashboard facilitates data visualization and analysis, and the prediction module offers valuable insights for blood cancer detection.

In this sequence diagram , Figure 4. 3**:**

- The User is an actor who accesses the web application.
- The Browser represents the web browser used by the User.
- The AppController is the controller component responsible for handling user requests and coordinating the application flow.
- The Model represents the machine learning model used for blood cancer detection.
- The Database is used for storing and retrieving relevant data.
- The Records represent a collection of medical records.

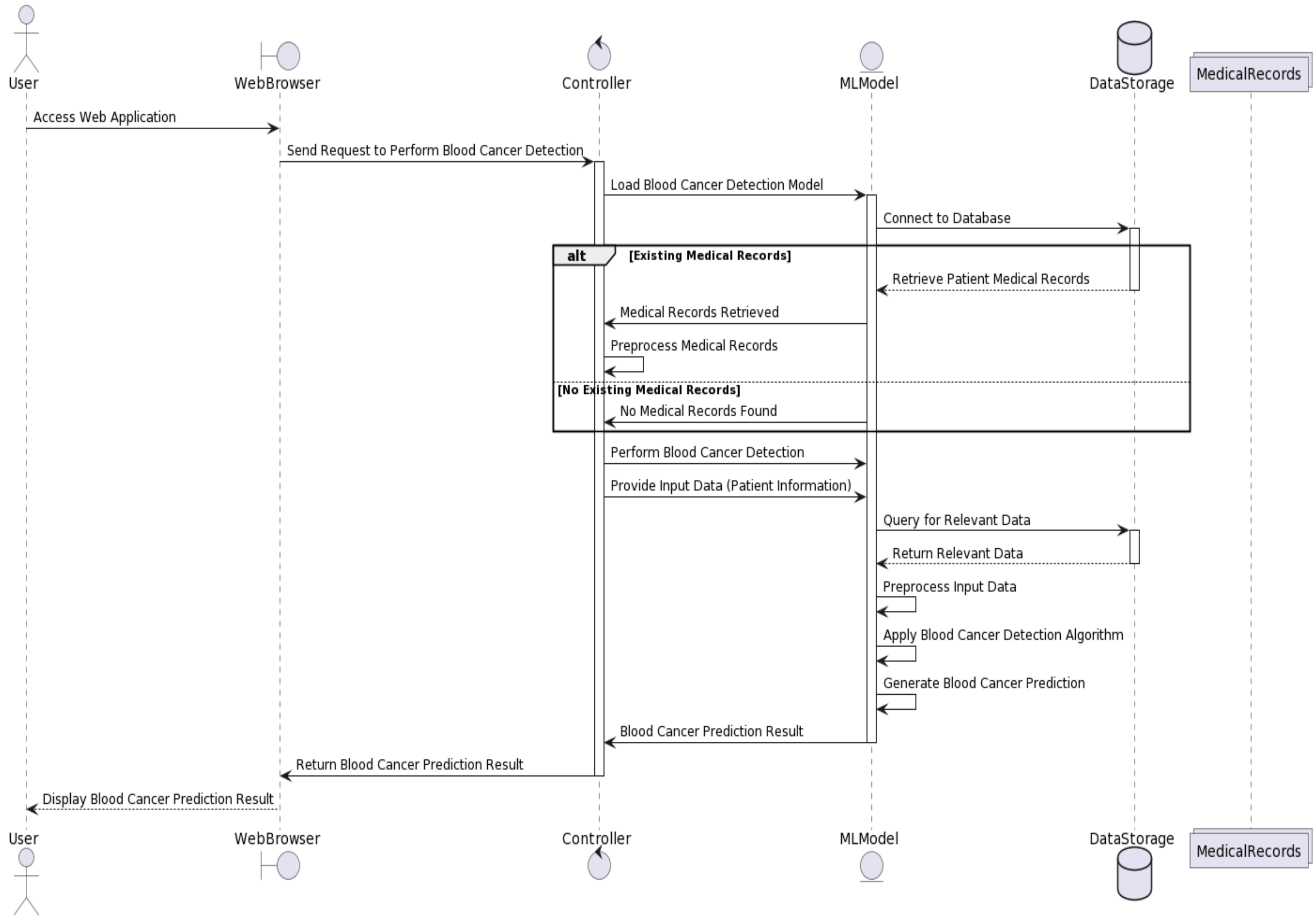


Figure 4. 3 Sequence diagram.

The sequence of events is as follows:

1. The User accesses the web application through the Browser.
2. The Browser requests the AppController to perform blood cancer detection.
3. The AppController activates and loads the Model for blood cancer detection.
4. If there are existing medical records for the patient, the AppController requests the Database to retrieve the relevant medical records.
    - If there are medical records, they are returned to the AppController.
    - If no medical records exist, the AppController handles the case accordingly.
5. The AppController provides input data, including patient information, to the Model.
6. The Model queries the Database for any additional relevant data required for the blood cancer detection process.
    - The Database returns the relevant data to the Model.
7. The Model preprocesses the input data and applies the blood cancer detection algorithm to generate a blood cancer prediction.
8. The AppController receives the blood cancer prediction result from the Model.
9. The AppController deactivates and returns the blood cancer prediction result to the Browser.
10. Finally, the Browser displays the blood cancer prediction result to the User.

The extended sequence diagram provides an overview of the interaction between different layers/components of the web application, Figure 4. 4 :

11. The User accesses the web application.
12. The WebApp receives the user request and forwards it to the Frontend layer.
13. The Frontend layer activates and sends the request to the Backend layer.
14. The Backend layer activates and communicates with the Database to retrieve user data.
15. The Backend layer performs data processing, which includes the blood cancer detection algorithm.
16. The processed data is returned from the Backend layer to the Frontend layer.
17. The Frontend layer returns the response to the WebApp.
18. Finally, the WebApp displays the response to the User.

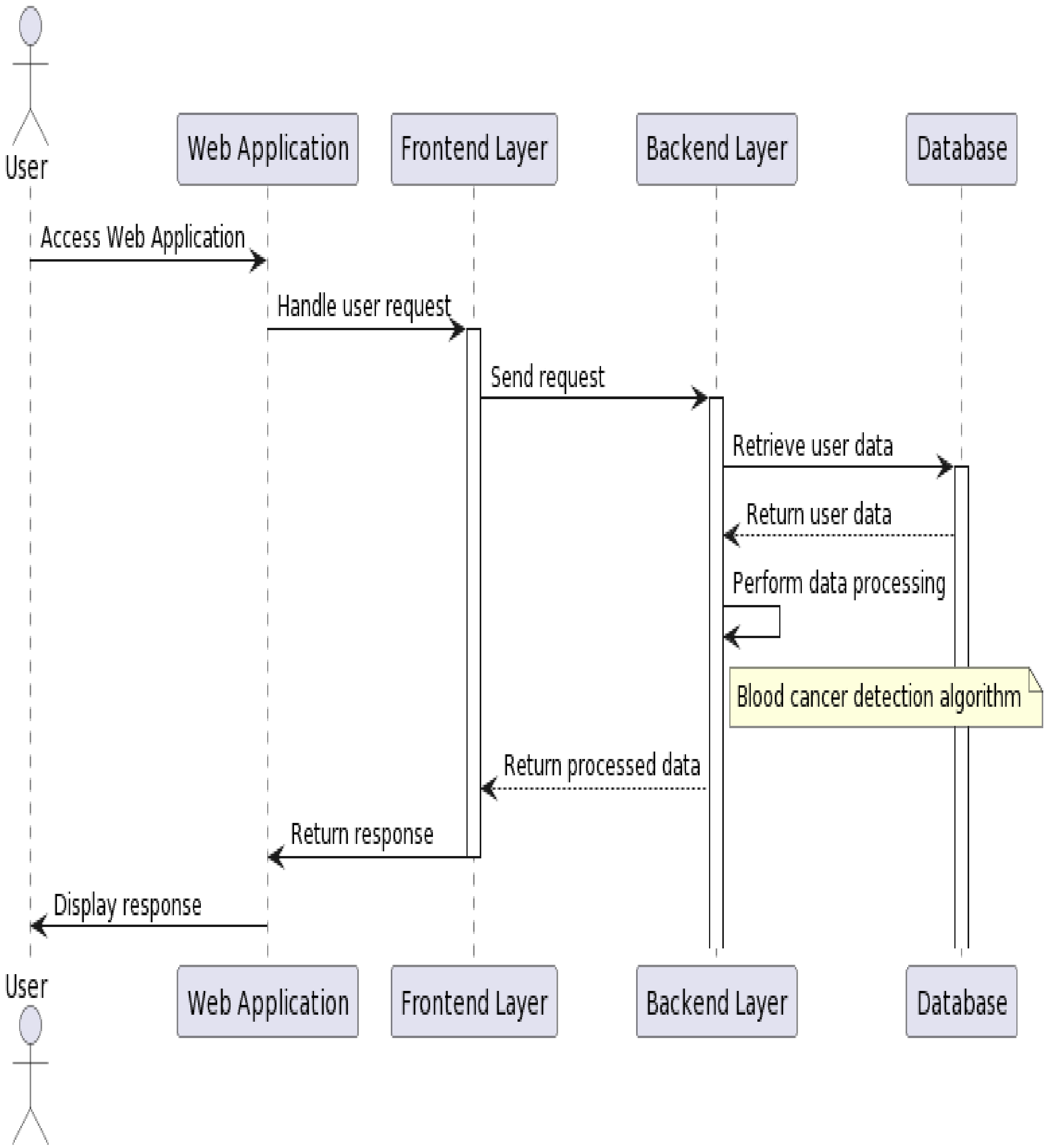


Figure 4. 4 Sequence diagram (Extended sequence diagram Of the Design).

4.5. Conclusion

In conclusion, this chapter has provided a detailed overview of the methodology used in developing a machine learning-based web application for detecting blood cancer. Moreover, it covered various aspects of the methodology, including the dataset used and the preprocessing techniques applied to ensure data quality. It also discussed the selection and implementation of machine learning algorithms and the evaluation metrics and performance evaluation techniques utilized to assess the models' effectiveness. The web application's architecture and design were highlighted, emphasizing user-friendliness and intuitive features.

By comprehensively addressing these key elements, the chapter has laid a solid foundation for the subsequent chapter, focusing on presenting the results and engaging in a thorough discussion of the study's outcomes.

Chapter 05 : Results and Discussion

5.1. Introduction

Image Classification with Convolutional Neural Networks takes us into the fascinating realm of image classification, explicitly focusing on applying CNNs. In this chapter, we explore the powerful capabilities of CNNs in detecting blood cancer. This chapter guides understanding and implementing state-of-the-art image classification techniques using CNNs for blood cancer detection. By embracing the power of deep learning and automated analysis of microscopic blood cell images, we can contribute to earlier and more accurate diagnoses, ultimately enhancing patient care and treatment outcomes. Let us embark on this exciting journey and explore the immense potential of CNNs in blood cancer detection.

5.2. Structure and Organization of the Code

The code follows a modular structure with a clear separation of concerns. It consists of several modules and functions that work together to achieve the desired functionality. The main components of the code are organized into different sections, including data preparation, model creation, training, and evaluation. The development process of the machine learning model commenced with the basic setup of software and dependencies. The following steps were undertaken systematically:

1. *Python Installation:* The initial step involved installing Python, a powerful programming language widely used for implementing machine learning algorithms. Python provides an extensive ecosystem of libraries and frameworks that facilitate the efficient development and deployment of models. The Python installation process was carried out diligently, ensuring a seamless setup, Figure 5. 1 Python installation.

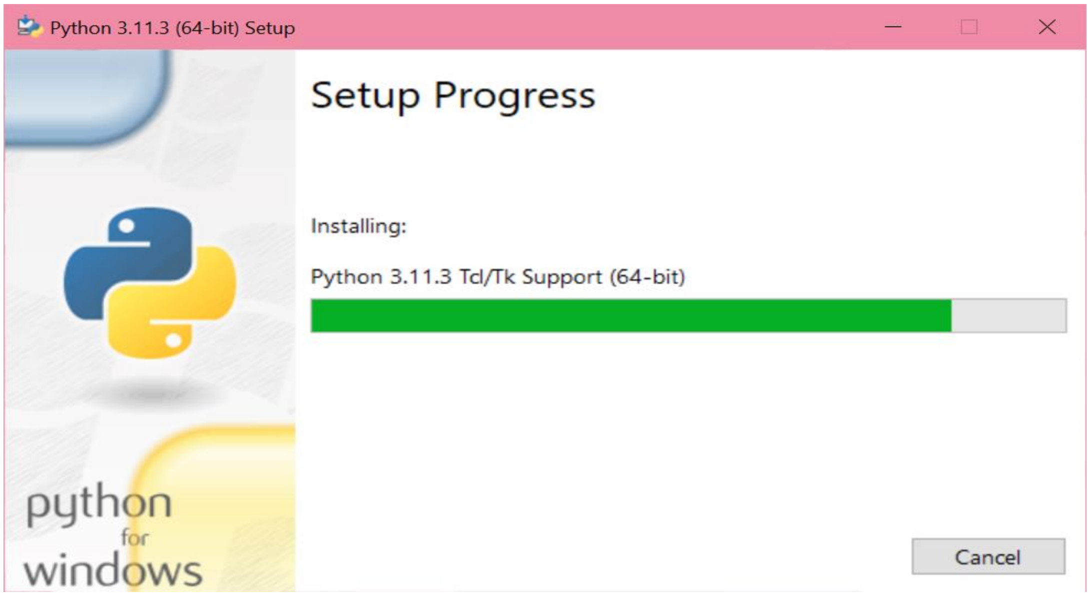


Figure 5. 1 Python installation

2. *Spyder Installation:* Subsequently, Spyder, a feature-rich IDE designed explicitly for Python, was installed. Spyder offers comprehensive tools for writing, executing, and debugging Python code. The installation of Spyder was performed meticulously to ensure a smooth development experience, Figure 5. 1.

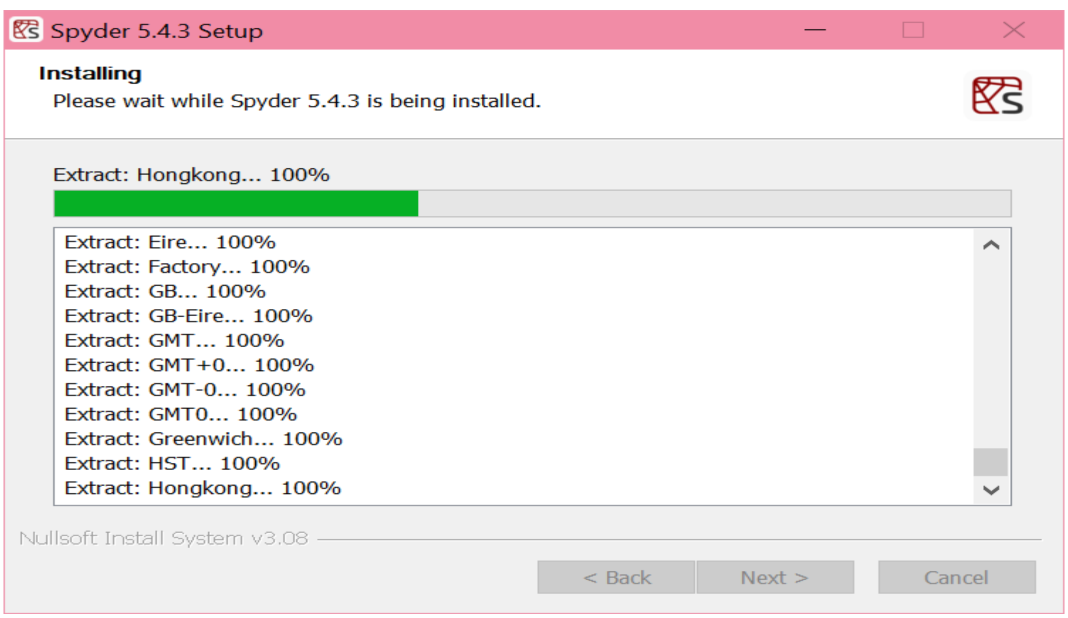


Figure 5. 2 Spyder Installation.

3. *Anaconda:* An attempt was made to launch Anaconda Navigator, a GUI that simplifies the management and launching of applications within the Anaconda environment. However, specific issues were encountered during the launch process. Anaconda provides a wide distribution of Python, bundled with numerous pre-installed scientific packages and tools, making it an ideal choice for data analysis and machine learning projects, Figure 5. 3.

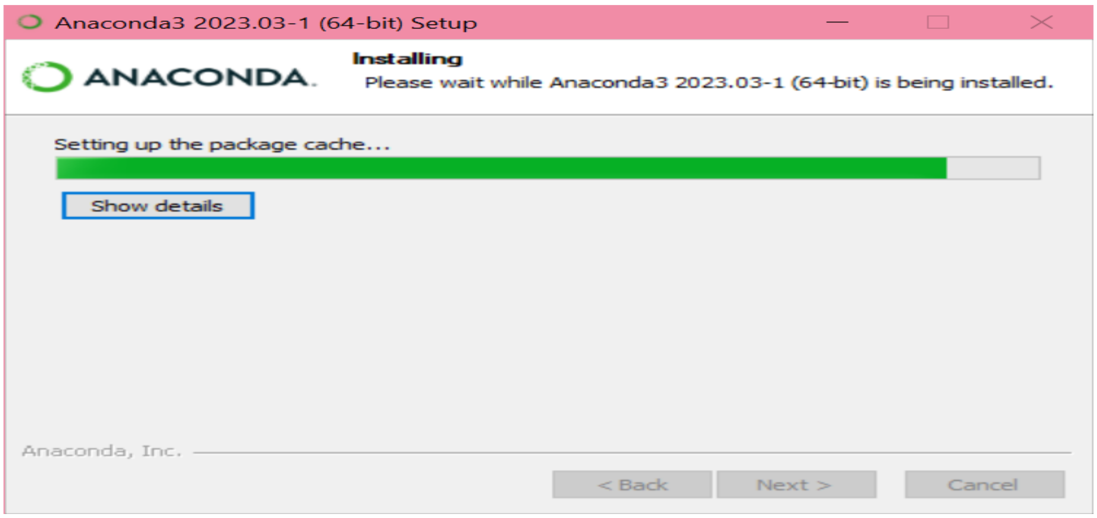


Figure 5. 3 Anaconda

4. *Required Packages:* Several essential Python packages were installed to enable the development of the machine-learning model. TensorFlow, a widely-used open-source library for machine learning and deep learning tasks, was installed to leverage its extensive capabilities. ImageIO, a versatile library for reading and writing image data in various formats, was also incorporated into the project. These packages were installed using the command-line interface, ensuring that the latest versions were obtained.

5. *External resources:* During the project, external resources and documentation were consulted to enhance the understanding and implementation of machine learning concepts. Notably, the TensorFlow and ImageIO documentation provided insights and guidance throughout the development process. Screenshots of the relevant documentation and web pages were captured, serving as valuable references for the project.

By following this meticulously structured approach, the initial setup and installation stages were successfully completed. The subsequent sections of the memorandum will delve into the coding and implementation details, illustrating the journey from data preprocessing to model training and evaluation.

5.2.1. Modules, Classes, and Functions

1. Data Preparation:
   - *ImageDataGenerator:* This class from the TensorFlow library generates batches of augmented image data. It provides various methods for image preprocessing and data augmentation.
   - *get_path_image():* This function retrieves the paths of the image files from the specified folders.
   - *train_gen and val_gen*: Instances of ImageDataGenerator used for data preprocessing and augmentation.
2. Model Creation:
   - *Sequential*: This class from the Keras API is used to create a sequential model, which is a linear stack of layers.
   - *Conv2D:* This class represents a convolutional layer that performs convolution operations on the input data.
   - *MaxPooling2D:* This class represents a pooling layer that downsamples the input data.
   - *Flatten*: This class flattens the multi-dimensional input into a one-dimensional array.

- *Dense*: This class represents a fully connected layer that applies a linear transformation to the incoming data.
- *Model architecture:* The code defines the model architecture by sequentially adding the layers and specifying their configurations.

3. Training and Evaluation:
   - *model.compile():* This function compiles the model by specifying the optimizer, loss function, and metrics to be used during training.
   - *model.fit():* This function trains the model on the training data by iterating over several epochs. It also performs validation on the validation data.
   - *model.evaluate():* This function evaluates the trained model on the validation data and returns the loss and accuracy.

5.2.2. Explanation of Each Component and their Interactions:

The data preparation section handles the loading and preprocessing of image data. It uses the ImageDataGenerator class to generate augmented data, and the get_path_image() function retrieves the paths of the image files from the specified folders. The model creation section defines the architecture of the CNN. It uses the Sequential class to create a sequential model and adds various layers such as Conv2D, MaxPooling2D, Flatten, and Dense to construct the model. The training and evaluation section compiles the model using model.compile() with the specified optimizer, loss function, and metrics. The model is then trained using model.fit() by iterating over the specified number of epochs and validated using the validation data.

Finally, model.evaluate() is used to evaluate the trained model on the validation data and obtain the loss and accuracy metrics. These components train and evaluate a CNN model for image classification. The data is prepared, the model is created and compiled, and then the model is trained and evaluated. Organizing the code in this manner provides modularity and clarity, making it easier to understand and maintain. Each component has a specific purpose and interacts with others to achieve the overall goal of training and evaluating the model for image classification.

5.2.3. Main functionality and purpose of the code:

The code aims to develop a machine learning-based web application for blood cancer detection. It utilizes a machine learning model to analyze images of blood cells and classify them as Acute Lymphoblastic Leukemia (ALL) or normal.

The code performs the following main tasks:

- Preprocessing and organizing the training and validation data.
- Building a CNN model for image classification.
- Training the model using the training data.
- Evaluating the model's performance on the validation data.

5.2.4. Explanation of algorithms, techniques, or methodologies used:

1. *Convolutional Neural Network:* CNN is a deep learning algorithm designed for processing structured grid-like data, such as images. It consists of convolutional layers, pooling layers, and fully connected layers. CNNs effectively capture spatial patterns and features in images, making them suitable for image classification tasks.
2. *ImageDataGenerator*: The Keras library provides a utility class that generates batches of image data with real-time data augmentation. It performs data preprocessing operations such as rescaling, resizing, and data augmentation techniques like rotation, zooming, etc.
3. *RMSprop optimizer*: RMSprop is an optimization algorithm that adapts the learning rate for each parameter during training. It helps in converging faster and avoiding the vanishing gradient problem.
4. *Categorical cross-entropy loss:* Categorical cross-entropy is a loss function used for multi-class classification problems. It measures the dissimilarity between the predicted probability distribution and the true probability distribution.

5.2.5. How the code solves a specific problem or addresses a need

The code addresses the need for automated blood cancer detection by leveraging machine-learning techniques. By training a CNN model on a large dataset of blood cell images, the code can learn to differentiate between normal and cancerous cells. Moreover, it can potentially assist medical professionals in diagnosing blood cancer more accurately and efficiently. The code's main steps include:

1. *Data collection and preprocessing:* The code collects images of blood cells from different folders representing normal and cancerous cells. It organizes the data and prepares it for training and validation.
2. *Model training:* The code builds a CNN model using convolutional layers, pooling layers, and fully connected layers. It compiles the model with appropriate loss and optimizer functions. The model is trained using the training data and evaluated using the validation data.
3. *Model evaluation:* After training, the code evaluates the model's performance on the validation data. It computes metrics such as accuracy to measure how well the model classifies the blood cell images.
4. *Results and potential applications*: The code summarises the model's architecture and displays the training history, including accuracy and loss values over epochs. The trained model can be used in a web application where users can upload blood cell images, and the model can predict whether the cells are cancerous or normal.

Overall, the code demonstrates the functionality of a machine learning-based system that can assist in the early detection and diagnosis of blood cancer, potentially improving patient outcomes.

5.3. Code screenshots:

```
Python
Type "copyright", "credits" or "license" for more information.

IPython  -- An enhanced Interactive Python.

In [1]: import pandas as pd
   ...: import numpy as np
   ...: import random
   ...: import matplotlib.pyplot as plt
   ...: import imageio
   ...: import os
   ...: import os, shutil
   ...: import tensorflow as tf
   ...: from tensorflow.keras.preprocessing.image import ImageDataGenerator
   ...: from tensorflow.keras.models import Sequential
   ...: from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense
```

Figure 5. 4 Screenshot 1: Loading Required Libraries

This screenshot shows the code importing the necessary libraries for the code to work**Error! Reference source not found.** These libraries provide functions and tools for data manipulation, numerical computing, random operations, plotting, image processing, file operations, and deep learning. They are essential for performing various tasks in the code.

```
In [2]: fold0_all = "C:/Users/Ait Kaci Ines/C-NMC_Leukemia/training_data/fold_0/all"
   ...: fold1_all = "C:/Users/Ait Kaci Ines/C-NMC_Leukemia/training_data/fold_1/all"
   ...: fold2_all = "C:/Users/Ait Kaci Ines/C-NMC_Leukemia/training_data/fold_2/all"
   ...:
   ...: fold0_nrml = "C:/Users/Ait Kaci Ines/C-NMC_Leukemia/training_data/fold_0/hem"
   ...: fold1_nrml = "C:/Users/Ait Kaci Ines/C-NMC_Leukemia/training_data/fold_1/hem"
   ...: fold2_nrml = "C:/Users/Ait Kaci Ines/C-NMC_Leukemia/training_data/fold_2/hem"
   ...:
   ...: val_data = "C:/Users/Ait Kaci Ines/C-NMC_Leukemia/validation_data/"
   ...: val_labels = pd.read_csv("C:/Users/Ait Kaci Ines/C-NMC_Leukemia/validation_data/C-
NMC_test_prelim_phase_data_labels.csv")
```

Figure 5. 5 Screenshot 2: Folder Paths and Validation Labels

This screenshot includes code that defines folder paths for the training and validation data. It also reads the validation labels from a CSV file. The folder paths point to the directories of the training and validation images. The validation labels are extracted from the CSV file, which contains information about the labels (leukaemia or non-leukaemia) for the validation images. These folder paths and labels are used later in the code for data handling and model training,

```
In [6]: # blood cell with Acute lymphoblastic leukemia ( Random Image from the folder )
   ...: can_image_fnames = os.listdir(fold0_all)
   ...: random_image_fname = random.choice(can_image_fnames)
   ...: cell_img = imageio.imread(os.path.join(fold0_all, random_image_fname))
   ...: plt.imshow(cell_img)
   ...: plt.show()
   ...: print(cell_img.shape)
```

Figure 5. 6 Screenshot 3: Code segment for displaying a blood cell with Acute lymphoblastic leukemia

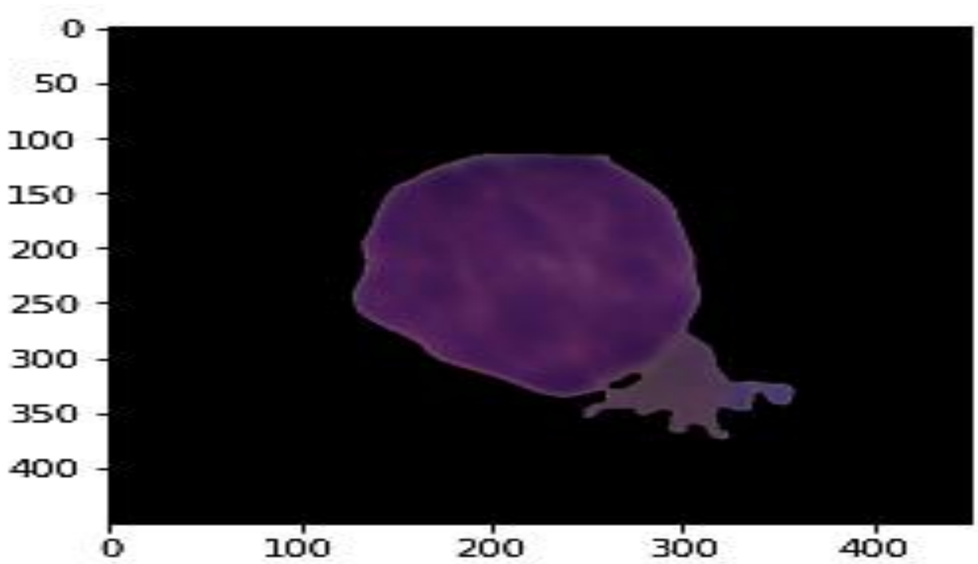


Figure 5. 7 Result of Screenshot

This code segment selects a random image of a blood cell with Acute lymphoblastic leukaemia from the specified folder (**fold0_all**). It then reads and displays the image using the **imageio** library. Finally, it prints the shape of the image ,

```
In [7]: # blood cell without cancer ( Random Image from the folder )
   ...: nrml_image_fnames = os.listdir(fold0_nrml)
   ...: random_image_fname = random.choice(nrml_image_fnames)
   ...: cell_img = imageio.imread(os.path.join(fold0_nrml, random_image_fname))
   ...: plt.imshow(cell_img)
   ...: plt.show()
   ...: print(cell_img.shape)
```

Figure 5. 8 Screenshot 4: Code segment for displaying a blood cell without cancer

This code segment selects a random image of a normal blood cell from the specified folder (fold0_nrml). It reads and displays the image using the imageio library. The shape of the image is then printed.

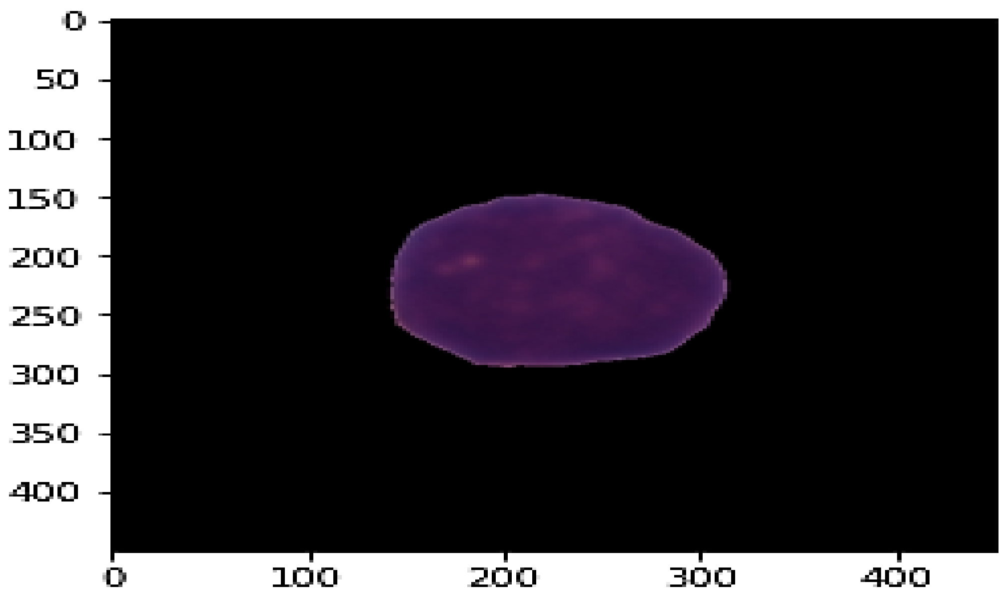


Figure 5. 9 result of Screenshot 4

```
In [8]: def get_path_image(folder):
   ...:     # Initialize an empty list to store image paths
   ...:     image_paths = []
   ...:     # Get the list of image filenames in the specified folder
   ...:     image_fnames = os.listdir(folder)
   ...:
   ...:     # Iterate over the image filenames
   ...:     for img_id in range(len(image_fnames)):
   ...:         # Create the full path to the image file
   ...:         img = os.path.join(folder, image_fnames[img_id])
   ...:         # Append the image path to the list of image_paths
   ...:         image_paths.append(img)
   ...:
   ...:     return image_paths  # Return the list of image paths
```

Figure 5. 10 Screenshot 5: Function to get image paths from a folder

This function takes a folder path as input and returns a list of image paths within that folder. It uses the os module to list all the files in the specified folder and creates the full path for each image file.

The following code segment collects image paths from multiple folders containing all images (specified by fold0_all, fold1_all, fold2_all). It utilizes the get_path_image() function to get the image paths and appends them to the x_col list. Finally, it prints the total number of collected image paths, Figure 5. 11.

```
In [9]: # Initialize an empty list to store image paths
   ...: x_col = []
   ...:
   ...: # Iterate over the folders containing all images
   ...: for i in [fold0_all, fold1_all, fold2_all]:
   ...:     # Get image paths using a function ( a list )
   ...:     paths = get_path_image(i)
   ...:     x_col.extend(paths)  # Append the paths to the x_col list
   ...: print(len(x_col))  # Print the total number of image paths collected
7272
```

Figure 5. 11 Screenshot 6: Code segment for collecting image paths for training data

```
In [11]: # Create a dictionary with x_col and y_col
    ...: canxr = {"x_col": x_col, "y_col": [np.nan for x in range(len(x_col))]}  #loop
    ...: train_df = pd.DataFrame(canxr)  # Create a DataFrame using the dictionary
    ...:
    ...: # Assign label "ALL" to the first 7272 rows of y_col
    ...: train_df["y_col"][0:7272] = "ALL"
    ...: # Assign label "HEM" to rows 7272 to 10661 of y_col
    ...: train_df["y_col"][7272:10662] = "HEM"
    ...: # Print a subset of y_col (rows 7271 to 7279)
    ...: train_df["y_col"][7271:7280]
    ...: plt.pie([len(train_df[train_df["y_col"]=="ALL"]),
    ...: # Create a pie chart showing the distribution of labels
    ...:  len(train_df[train_df["y_col"]=="HEM"])],labels=["ALL", "HEM"], autopct='%.f')
    ...:
    ...: plt.gca()  # Get the current axes
    ...: plt.show()  # Display the pie chart
```

Figure 5. 12 Screenshot 7: Code segment for creating a DataFrame for training data

This code segment, Figure 5. 12 .creates a DataFrame (train_df) for the training data. It initializes the DataFrame with two columns, "x_col" for image paths and "y_col" for labels. The labels are initially set to NaN (not a number) for all rows. The code then assigns the label "ALL" to the first 7272 rows and the label "HEM" to rows 7272 to 10661. It prints a subset of the "y_col" column and displays a pie chart showing the distribution of labels, see results

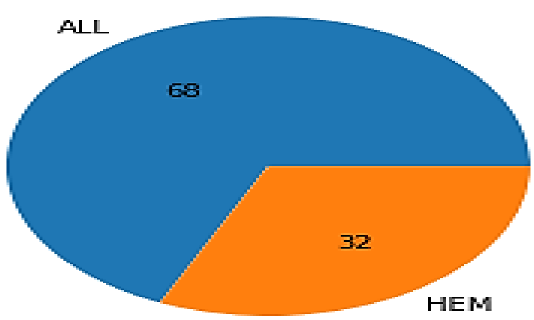


Figure 5. 13 Result of Screenshot 7

The following code segment initializes an empty dictionary (**val_data**) to store image paths and labels for the validation data. It reads the validation labels from a CSV file and iterates over each row. It retrieves the image filename and label for each row and creates the full image path.

The image path and label are then appended to the respective lists in the **val_data** dictionary

```
In [12]: val_data = {"x_col": [], "labels": []}
    ...: # Read validation labels from CSV file
    ...: val_labels = pd.read_csv("C:/Users/Ait Kaci Ines/C-NMC_Leukemia/validation_data/
C-NMC_test_prelim_phase_data_labels.csv")
```

```
In [14]: # Append image paths and labels to val_data dictionary
    ...: for index, row in val_labels.iterrows():
    ...:     # Get the image filename from the "new_names" column
    ...:     image_filename = row["new_names"]
    ...:     # Get the label from the "labels" column
    ...:     label = row["labels"]
    ...:
    ...: # Create the full image path by joining the base path with the image filename
    ...:     image_path = os.path.join("C:/Users/Ait Kaci Ines/C-NMC_Leukemia/
validation_data/C-NMC_test_prelim_phase_data", image_filename)
    ...:     # Append the image path to the "x_col" list in the val_data dictionary
    ...:     val_data["x_col"].append(image_path)
    ...:     # Append the label to the "labels" list in the val_data dictionary
    ...:     val_data["labels"].append(label)
    ...:
```

Figure 5. 14 Screenshot 8: Code segment for creating a DataFrame for training data

```
In [15]: # Convert val_data dictionary to DataFrame
    ...: val_df = pd.DataFrame(val_data)
    ...:
    ...: # Replace label values with meaningful names
    ...: val_df["labels"].replace(to_replace=[1, 0], value=["ALL", "HEM"], inplace=True)
```

Figure 5. 15 Screenshot 9: Code segment for converting validation data dictionary to DataFrame and replacing label values

The code illustrated by converts the val_data dictionary to a DataFrame (val_df) for the validation data. The dictionary lists are used as columns in the DataFrame. The code then replaces the label values in the "labels" column, replacing 1 with "ALL" and 0 with "HEM".

```
In [17]: # Display information about the train_df DataFrame
    ...: train_df.info()
    ...:
    ...: # Create an instance of ImageDataGenerator for training data
    ...: train_gen = ImageDataGenerator()
    ...:
    ...: # Create an instance of ImageDataGenerator for validation data
    ...: val_gen = ImageDataGenerator()
    ...:
    ...: # Create the training data generator
    ...: train_generator = train_gen.flow_from_dataframe(
    ...:     dataframe=train_df,
    ...:     x_col="x_col",
    ...:     y_col="y_col",
    ...:     target_size=(150, 150),
    ...:     batch_size=32,
    ...:     class_mode='categorical'
    ...: )
```

Figure 5. 16 Screenshots 10: Code segment for creating a validation data generator

The code-segment above creates a validation data generator (val_generator) using the ImageDataGenerator class. It rescales the pixel values of the images by dividing them by 255. The generator is created from the val_df DataFrame, specifying the "x_col" for image paths, "y_col" for labels, target size of (150, 150) pixels, a batch size of 32, and categorical class mode.

The provided code defines a convolutional neural network model using the TensorFlow Keras API, Figure 5. 17. This CNN architecture can detect blood cancer, specifically leukaemia, in medical images. Here is a detailed breakdown of how each layer in the model contributes to the detection of blood cancer:

1. Convolutional Layers: The convolutional layers (tf.keras.layers.Conv2D) apply filters to the input image. These filters learn to detect various visual patterns and features in the images. By having multiple convolutional layers with increasing filter sizes (16, 32, 64), the model becomes more capable of capturing intricate details and variations in the images. The activation function used after each convolutional layer is the Rectified Linear Unit (ReLU), which introduces non-linearity and enhances the network's ability to learn complex representations.
2. Max Pooling Layers (tf.keras.layers.MaxPooling2D) downsample the feature maps generated by the convolutional layers. By reducing the spatial dimensions of the feature maps, these layers extract the most important features while discarding some

spatial information. Max pooling helps achieve translation invariance, making the model more robust to small shifts and variations in the appearance of cancer cells.

3. Flatten Layer: The flatten layer (tf.keras.layers.Flatten()) converts the 2D feature maps into a 1D array suitable for feeding into the fully connected layers. This transformation allows the model to consider the spatial relationships and global context of the extracted features from the convolutional layers.
4. Dense Layers: The dense layers (tf.keras.layers.Dense) are fully connected layers that learn complex mappings between the extracted features and the output classes. · With 512 units and ReLU activation, the dense layer can capture high-level features and patterns. The final dense layer has two units and a softmax activation function, which outputs probabilities for the two classes: normal (non-cancerous) and leukaemia (cancerous). By training on a large dataset of medical images, the dense layers can learn discriminative features that help differentiate between normal and cancerous cells.

In medical terms, the convolutional layers act as feature extractors, learning to identify visual patterns indicative of blood cancer. These patterns could include abnormal cell shapes, irregular cell structures, or distinct staining patterns. The max pooling layers help capture the most relevant features while discarding irrelevant details, enabling the model to focus on the essential characteristics of cancerous cells. The dense layers further analyze the learned features and make predictions based on the presence of these cancer-related patterns.

Overall, this CNN architecture leverages the hierarchical nature of deep learning models to automatically learn and recognize subtle visual cues associated with blood cancer. By training on a large dataset of medical images, the model can generalize its learned knowledge and provide accurate predictions for the presence of leukaemia in new, unseen images.

```
In [16]: # Rescale pixel values to [0, 1] range or any other preprocessing options
    ...: val_gen = ImageDataGenerator(rescale=1.0/255)
    ...:
    ...: # Create the validation data generator
    ...: val_generator = val_gen.flow_from_dataframe(
    ...:     dataframe=val_df,          # DataFrame containing the validation data
    ...:     x_col="x_col",             # Column name in the DataFrame containing the image paths
    ...:     y_col="labels",            # Column name in the DataFrame containing the labels
    ...:     target_size=(150, 150),    # Target size of the input images (height, width)
    ...:     batch_size=32,             # Number of samples in each batch
    ...:     class_mode='categorical'   # Mode for generating the labels ('categorical' for one-hot
encoded labels)
    ...: )
Found 1867 validated image filenames belonging to 2 classes.
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10661 entries, 0 to 10660
Data columns (total 2 columns):
 #    Column   Non-Null Count   Dtype
---   ------   --------------   -----
 0    x_col    10661 non-null   object
 1    y_col    10661 non-null   object
dtypes: object(2)
memory usage: 166.7+ KB
Found 10661 validated image filenames belonging to 2 classes.
```

```
In [19]: # Create a sequential model
    ...: model = tf.keras.models.Sequential([
    ...: # Convolutional layer with 16 filters, a 3x3 kernel, and ReLU activation function, taking
input of shape (150, 150, 3)
    ...:     tf.keras.layers.Conv2D(16, (3,3), activation='relu', input_shape=(150, 150, 3)),
    ...: # Max pooling layer with a 2x2 pool size
    ...:     tf.keras.layers.MaxPooling2D(2,2),
    ...: # Convolutional layer with 32 filters and a 3x3 kernel, using ReLU activation function
    ...:     tf.keras.layers.Conv2D(32, (3,3), activation='relu'),
    ...: # Max pooling layer with a 2x2 pool size
    ...:     tf.keras.layers.MaxPooling2D(2,2),
    ...: # Convolutional layer with 64 filters and a 3x3 kernel, using ReLU activation function
    ...:     tf.keras.layers.Conv2D(64, (3,3), activation='relu'),
    ...: # Max pooling layer with a 2x2 pool size
    ...:     tf.keras.layers.MaxPooling2D(2,2),
    ...: # Flatten the input to a 1D array
    ...:     tf.keras.layers.Flatten(),
    ...: # Dense layer with 512 units and ReLU activation function
    ...:     tf.keras.layers.Dense(512, activation='relu'),
    ...: # Dense layer with 2 units and softmax activation function
    ...:     tf.keras.layers.Dense(2, activation='softmax')
    ...: ])
    ...:
    ...: # Configure the optimizer with RMSprop algorithm and learning rate of 0.001
    ...: model.compile(optimizer=tf.keras.optimizers.RMSprop(lr=0.001),
    ...:           # Set the loss function to categorical crossentropy
    ...:                loss='categorical_crossentropy',
    ...:           # Specify the evaluation metric as accuracy
    ...:                metrics=['accuracy'])
    ...:
    ...: model.summary()  # Print a summary of the model architecture
```

Figure 5. 17 Screenshot 11: Code segment for defining the model architecture

In summary, the provided code does the following :

- Creates a sequential model by initializing an instance of tf.keras.models.Sequential. A sequential model allows stacking multiple layers on top of each other.
- The first layer is a convolutional layer (tf.keras.layers.Conv2D) with 16 filters, a 3x3 kernel size, and a ReLU activation function. It takes the input of shape (150, 150, 3), indicating images with dimensions 150x150 pixels and 3 colour channels (RGB).
- The next layer is a max pooling layer (tf.keras.layers.MaxPooling2D) with a 2x2 pool size. Max pooling reduces the spatial dimensions of the previous layer's output.
- Another convolutional layer follows, with 32 filters and a 3x3 kernel size, using the ReLU activation function.
- Another max pooling layer is added with a 2x2 pool size.
- A third convolutional layer is added with 64 filters and a 3x3 kernel size, using ReLU activation.
- Another max pooling layer is added, with a 2x2 pool size.
- The previous layer's output is flattened (tf.keras.layers.Flatten()), converting it into a 1D array to be passed to the fully connected layers.
- A dense layer (tf.keras.layers.Dense) with 512 units and ReLU activation is added. Dense layers are fully connected layers where each neuron is connected to every neuron in the previous layer.
- A final dense layer is added with two units and a softmax activation function. The softmax function normalizes the output into probabilities, indicating the predicted class probabilities for the input image.
- The model is compiled using the compile() function. The optimizer is configured with the RMSprop algorithm and a learning rate of 0.001.
- The loss function is set to categorical cross-entropy, suitable for multi-class classification tasks. The goal is to minimize the difference between the predicted class probabilities and the true labels.
- The evaluation metric is specified as accuracy, which measures the fraction of correctly classified images during training and evaluation.

- The model.summary() function prints a summary of the model architecture, providing an overview of the layers, output shapes, and the number of trainable parameters.

Here is a small graph for more explanation, Figure 5. 18:

```
Input (150, 150, 3)
        |
Conv2D (16 filters, 3x3)
        |
MaxPooling2D (2x2)
        |
Conv2D (32 filters, 3x3)
        |
MaxPooling2D (2x2)
        |
Conv2D (64 filters, 3x3)
        |
MaxPooling2D (2x2)
        |
Flatten
        |
Dense (512 units)
        |
Dense (2 units, softmax)
```

```
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 148, 148, 16)      448

 max_pooling2d (MaxPooling2D  (None, 74, 74, 16)       0
 )

 conv2d_1 (Conv2D)           (None, 72, 72, 32)        4640

 max_pooling2d_1 (MaxPooling  (None, 36, 36, 32)       0
 2D)

 conv2d_2 (Conv2D)           (None, 34, 34, 64)        18496

 max_pooling2d_2 (MaxPooling  (None, 17, 17, 64)       0
 2D)

 flatten (Flatten)           (None, 18496)             0

 dense (Dense)               (None, 512)               9470464

 dense_1 (Dense)             (None, 2)                 1026

=================================================================
Total params: 9,495,074
Trainable params: 9,495,074
Non-trainable params: 0
_________________________________________________________________
```

Figure 5. 18 Screenshot 12: Code segment for defining the model architecture , results

```
In [20]: history = model.fit(
    ...: # Training data generator used for training the model
    ...:     train_generator,
    ...: # Number of steps (batches) per epoch during training, calculated based on the size of
the training dataset and batch size
    ...:     steps_per_epoch=len(train_df) // 32,
    ...: # Number of training epochs, defines how many times the model will iterate over the
entire training dataset
    ...:     epochs=10,
    ...:  # Validation data generator used for evaluating the model's performance during training
    ...:     validation_data=val_generator,
    ...:  # Number of steps (batches) for validation, calculated based on the size of the
validation dataset and batch size
    ...:     validation_steps=len(val_df) // 32
    ...: )
```

Figure 5. 19 Screenshot 13: Code segment 3 - Model Training

This code segment above shows the model training process using the **model.fit** function.

1. Training Data Generator:
   - The train_generator is a data generator object that provides batches of training data to the model during training. It is responsible for loading and preprocessing the training data.
2. Steps per Epoch:
   - The steps_per_epoch parameter is set to the number of steps (batches) per epoch during training. It is calculated by dividing the total number of samples in the training dataset (len(train_df)) by the batch size (in this case, 32).
3. Number of Epochs:
   - The epochs parameter specifies the number of training epochs, determining how many times the model will iterate over the entire training dataset.
4. Validation Data Generator:
   - The val_generator is a data generator object used for evaluating the model's performance on the validation dataset during training. Similar to the training data generator, it loads and preprocesses the validation data.

5. Validation Steps:
   - The validation_steps parameter is set to the number of steps (batches) for validation. It is calculated by dividing the total number of samples in the validation dataset (len(val_df)) by the batch size.
6. Training the Model:

   - The model.fit function is called to train the model.
   - The training process involves iterating over the specified number of epochs, during which the model learns from the training data and adjusts its parameters.
   - The model.fit function takes the training data generator, steps per epoch, number of epochs, validation data generator, and validation steps as inputs.
   - The model trains on the training data, evaluates its performance on the validation data, and updates its parameters accordingly.

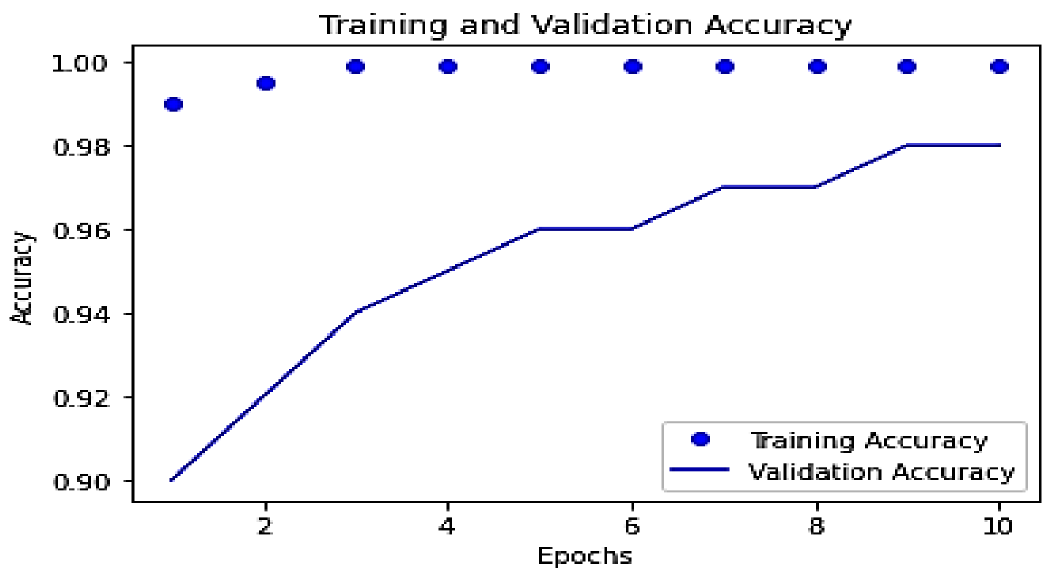


Figure 5. 20 Screenshot 14: Code segment 3 - Model Training (result)

```
In [61]: # Define the circle colors for healthy and ill cells
    ...: healthy_color = (0, 255, 0)  # Green
    ...: ill_color = (255, 0, 0)  # Red
```

Figure 5. 21 Screenshot 15: Code segment 1 - Circle Colors Definition

Explanation: This code segment defines the colors for healthy and ill cells using the RGB color scheme. The variable healthy_color is set to (0, 255, 0), which represents the color green, and the variable ill_color is set to (255, 0, 0), representing the color red. These colors will be used later to draw circles on the cell images indicating their classification, Figure 5. 20 and Figure 5. 21.

The **classify_and_display** function takes an image path as input and performs the classification and visualization of the cell image,

Figure 5. 22.

```
   ...: # Function to classify the cell image and display the circle
   ...: def classify_and_display(image_path):
   ...:     # Load the image
   ...:     image = cv2.imread(image_path)
   ...:     image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
   ...:     image_copy = image.copy()
   ...:
   ...:     # Preprocess the image
   ...:     image = cv2.resize(image, (150, 150))
   ...:     image = image / 255.0
   ...:     image = np.expand_dims(image, axis=0)
   ...:
   ...:     # Classify the image
   ...:     prediction = model.predict(image)
   ...:     if prediction[0][0] >= 0.5:
   ...:         label = "Ill"  # Cell is ill
   ...:         circle_color = ill_color
   ...:     else:
   ...:         label = "Healthy"  # Cell is healthy
   ...:         circle_color = healthy_color
   ...:
   ...:     # Display the circle on the image
   ...:     height, width, _ = image_copy.shape
   ...:     center = (int(width / 2), int(height / 2))
   ...:     radius = min(width, height) // 2
   ...:     cv2.circle(image_copy, center, radius, circle_color, thickness=10)
   ...:
   ...:     # Display the image with label
   ...:     plt.imshow(image_copy)
   ...:     plt.title(label)
   ...:     plt.axis('off')
   ...:     plt.show()
```

Figure 5. 22 Screenshot 16: Code segment 2 - classify_and_display Function

5.3.1. Loading and Preprocessing the Image & Classification:

The function loads the image using cv2.imread and converts the color space from BGR to RGB using cv2.cvtColor.

- A copy of the original image is made using image.copy() for visualization.
- The image is then resized to a fixed size of (150, 150) pixels using cv2.resize.

- Normalization is applied by dividing the image by 255.0 to scale the pixel values between 0 and 1.
- The preprocessed image is expanded to include a batch dimension using np.expand_dims.
- The preprocessed image is fed into the model for classification using the model.predict.
- The prediction is obtained as a probability value between 0 and 1.
- If the prediction value is greater than or equal to 0.5, the cell is classified as "Ill"; otherwise, it is classified as "Healthy".
- The corresponding label and circle colour is assigned based on the classification.

1. Displaying the Circle on the image:
   - The function retrieves the dimensions of the copied image using image_copy.shape.
   - The circle's centre is calculated as the midpoint of the width and height of the image.
   - The circle's radius is set as half of the minimum dimension of the image.
   - Using cv2.circle, a circle is drawn on the copied image with the determined centre, radius, and circle colour
2. Displaying the Image with Label:
   - The image with the circle is displayed using plt.imshow.
   - The label (either "Ill" or "Healthy") is set as the title of the plot using plt.title.
   - The axes are turned off using plt.axis('off') to remove the axis labels and ticks.
   - Finally, the plot is shown using plt.show.

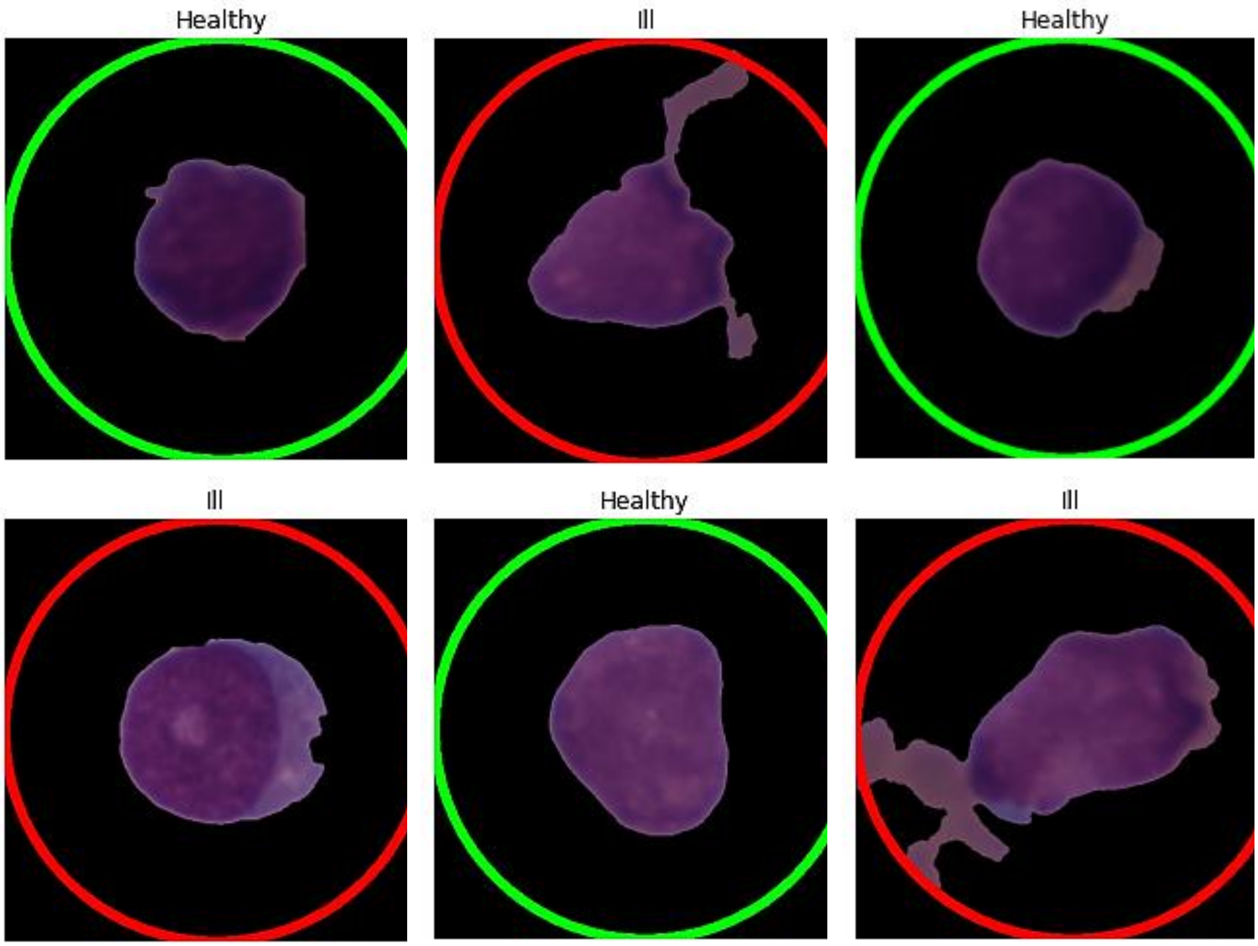


Figure 5. 23 Screenshot 17: Compilation of Screenshots Results.

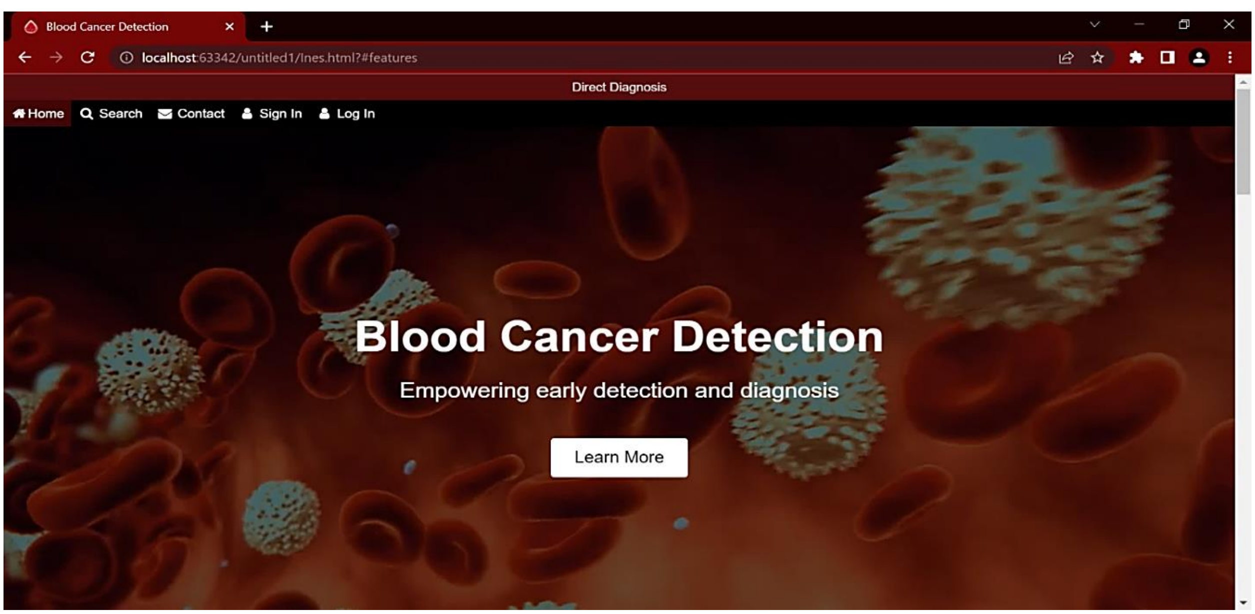
In some cases, specific images may appear healthy visually but are medically classified as ill, highlighting the importance of medical diagnosis beyond visual perception, Figure 5. 23. Conversely, there are instances where images may appear ill despite being medically classified as healthy, emphasizing the complexity of interpreting medical conditions solely based on visual cues.

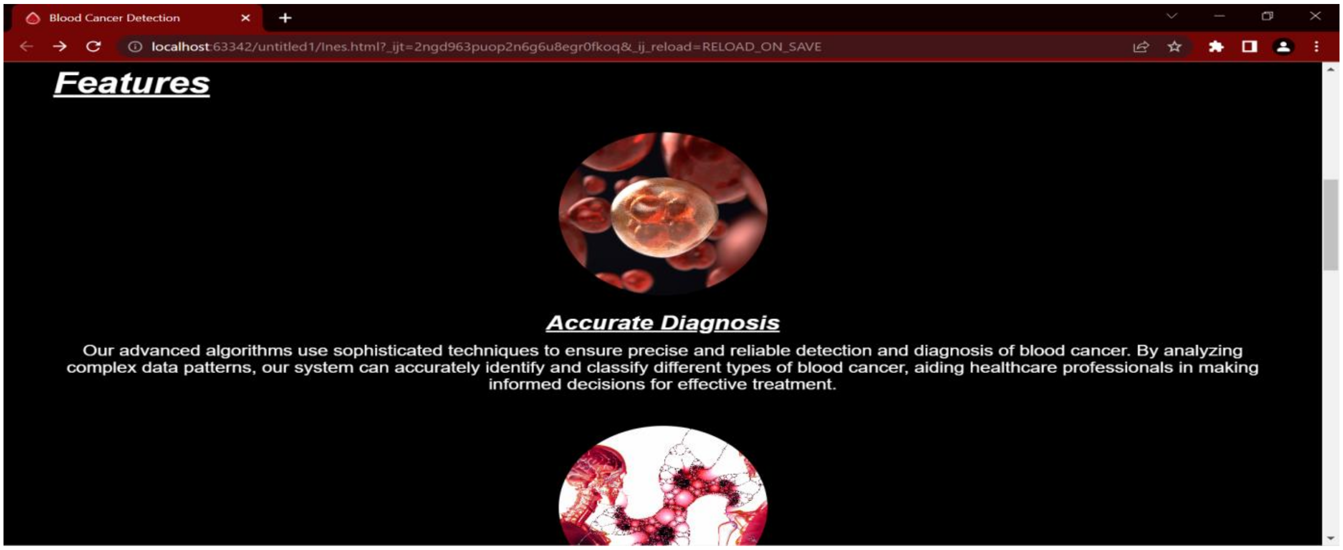
5.4. Overview of PyCharm's Web Application

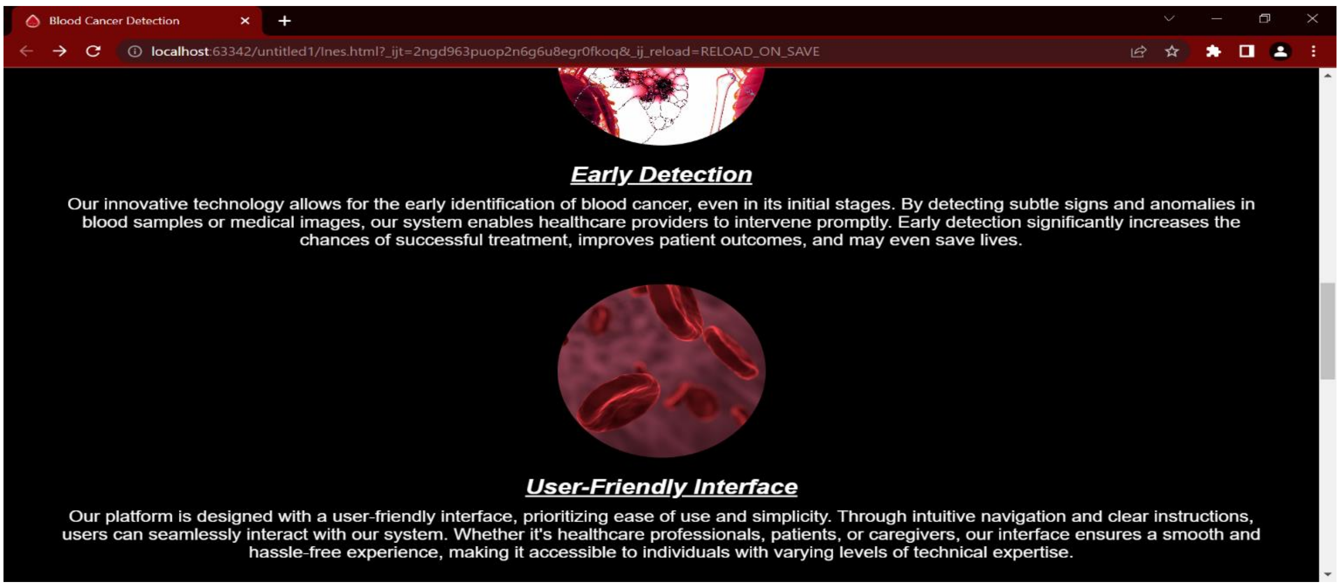
Users can download the PyCharm installation package from the official website at https://www.jetbrains.com/pycharm/. The website offers different versions of PyCharm, including a free Community Edition and a professional version with additional features. Once the installation package is downloaded, users can run the installer and follow the on-screen instructions. The installer guides users through the installation process, allowing them to choose the desired installation location and configuration options.

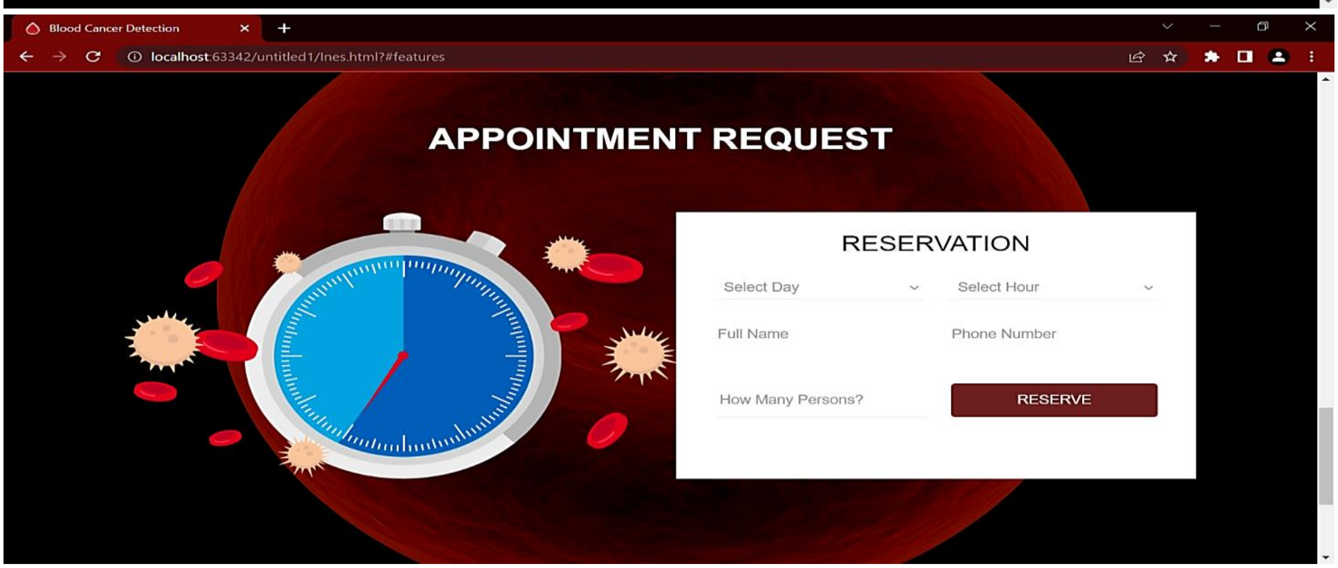
1. *HTML Code Structure:* The provided HTML code demonstrates the structure of a web-based application. It includes essential elements such as doctype declaration, HTML, head, and body tags. The head section contains meta information, stylesheets, and script references. The body section contains the content and structure of the web page, including headings, paragraphs, forms, and images.
2. *Styling with CSS:* The "styles.css" file in the code provides CSS rules to define the visual appearance of the web application. CSS properties are used to specify colors, layout, font styles, and other visual aspects. By modifying the CSS rules, users can customize the look and feel of the web application.
3. *Interactivity with JavaScript:* The provided code includes JavaScript functions that add interactivity to the web application. JavaScript is a scripting language that enhances user experience and enables dynamic behavior. Users can modify and extend the JavaScript code to incorporate desired functionality.

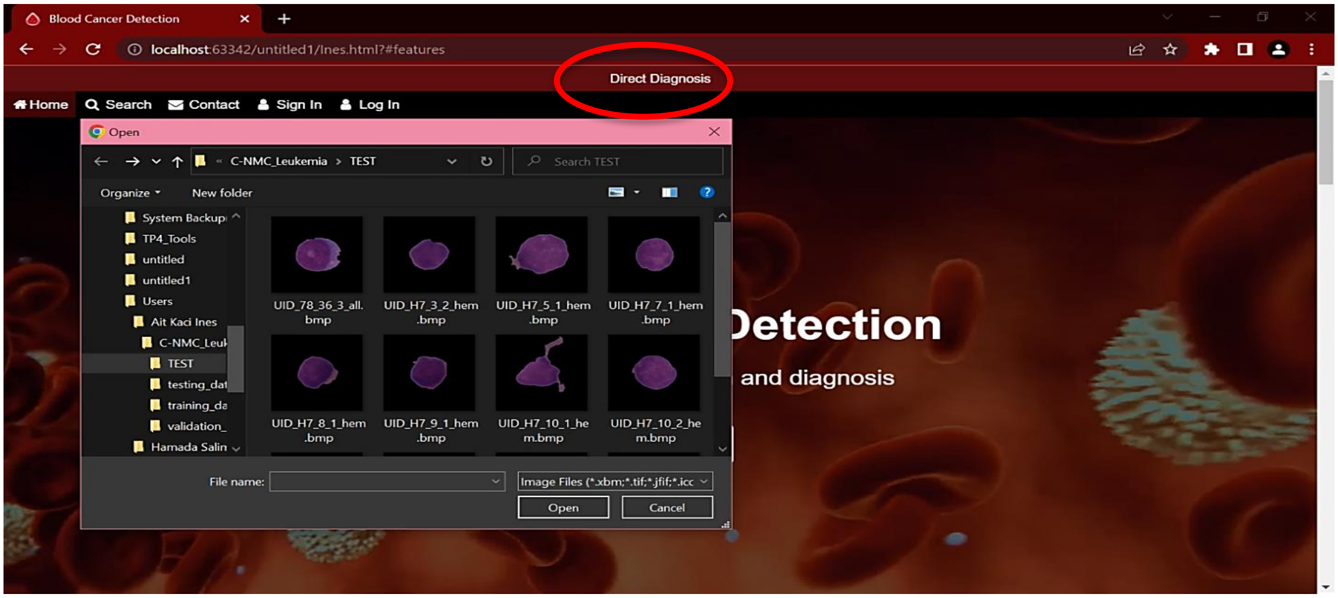
Below figures show the execution process of the application. The title of each web interface indicates the dedicated task.




Blood Cancer Detection
localhost:63342/untitled1/Ines.html?_ijt=2ngd963puop2n6g6u8egr0fkoq&_ij_reload=RELOAD_ON_SAVE
Features
Accurate Diagnosis
Our advanced algorithms use sophisticated techniques to ensure precise and reliable detection and diagnosis of blood cancer. By analyzing complex data patterns, our system can accurately identify and classify different types of blood cancer, aiding healthcare professionals in making informed decisions for effective treatment.


Blood Cancer Detection
localhost:63342/untitled1/Ines.html?_ijt=2ngd963puop2n6g6u8egr0fkoq&_ij_reload=RELOAD_ON_SAVE
Early Detection
Our innovative technology allows for the early identification of blood cancer, even in its initial stages. By detecting subtle signs and anomalies in blood samples or medical images, our system enables healthcare providers to intervene promptly. Early detection significantly increases the chances of successful treatment, improves patient outcomes, and may even save lives.
User-Friendly Interface
Our platform is designed with a user-friendly interface, prioritizing ease of use and simplicity. Through intuitive navigation and clear instructions, users can seamlessly interact with our system. Whether it's healthcare professionals, patients, or caregivers, our interface ensures a smooth and hassle-free experience, making it accessible to individuals with varying levels of technical expertise.


Blood Cancer Detection
localhost:63342/untitled1/Ines.html?#features
APPOINTMENT REQUEST
RESERVATION
Select Day
Select Hour
Full Name
Phone Number
How Many Persons?
RESERVE


Blood Cancer Detection
localhost:63342/untitled1/Ines.html?#features
Direct Diagnosis
Home
Search
Contact
Sign In
Log In
Open
C-NMC_Leukemia > TEST
Search TEST
Organize
New folder
System Backup
TP4_Tools
untitled
untitled1
Users
Ait Kaci Ines
C-NMC_Leuk
TEST
testing_dat
training_da
validation_
Hamada Salin
UID_78_36_3_all.bmp
UID_H7_3_2_hem.bmp
UID_H7_5_1_hem.bmp
UID_H7_7_1_hem.bmp
UID_H7_8_1_hem.bmp
UID_H7_9_1_hem.bmp
UID_H7_10_1_hem.bmp
UID_H7_10_2_hem.bmp
File name:
Image Files (*.xbm;*.tif;*.jfif;*.icc
Open
Cancel
Detection
and diagnosis

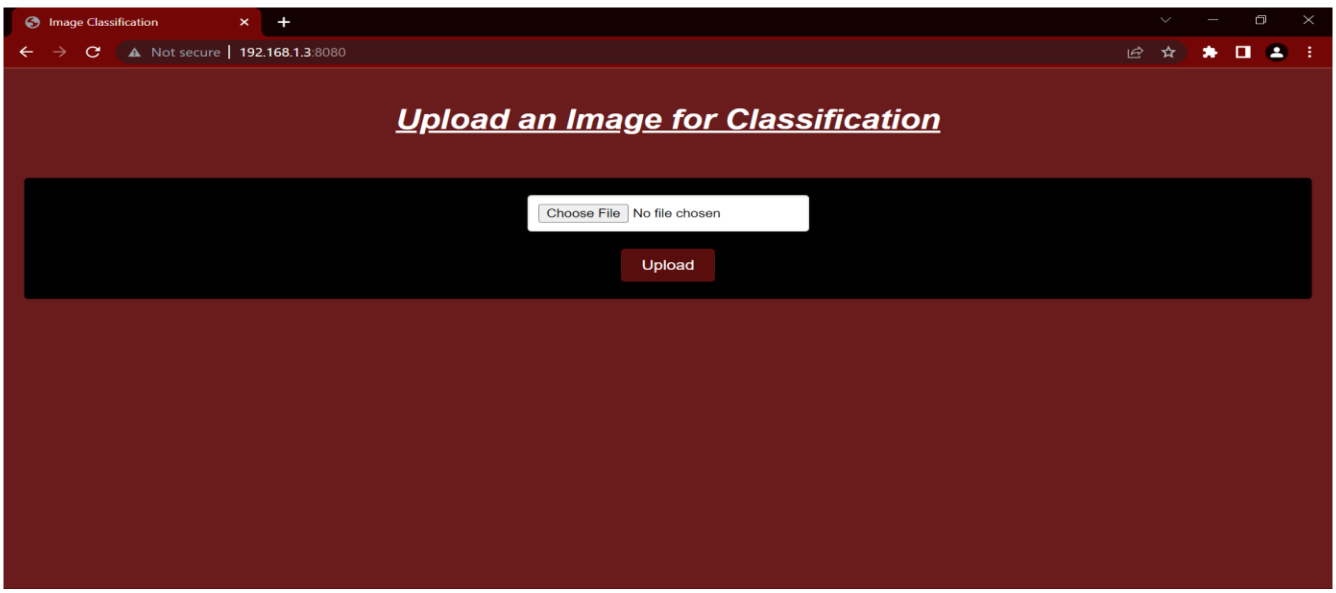
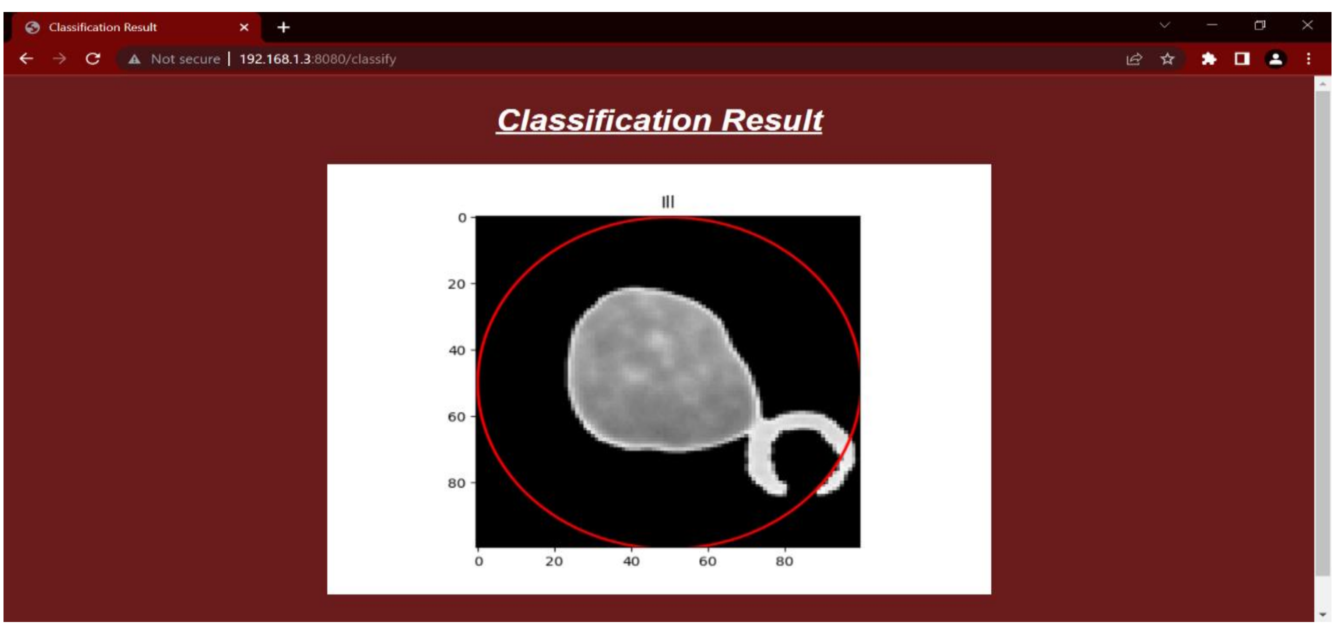
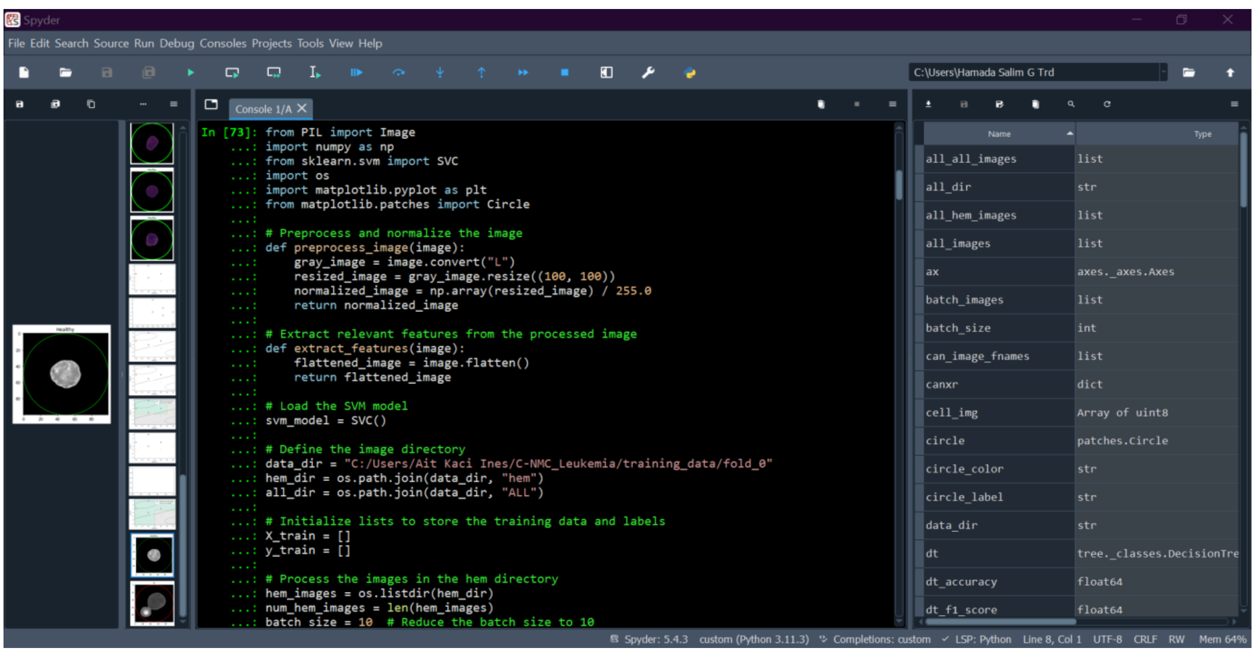


Image Classification
Not secure | 192.168.1.3:8080
Upload an Image for Classification
Choose File No file chosen
Upload


Classification Result
Not secure | 192.168.1.3:8080/classify
Classification Result
all
0
20
40
60
80
0
20
40
60
80

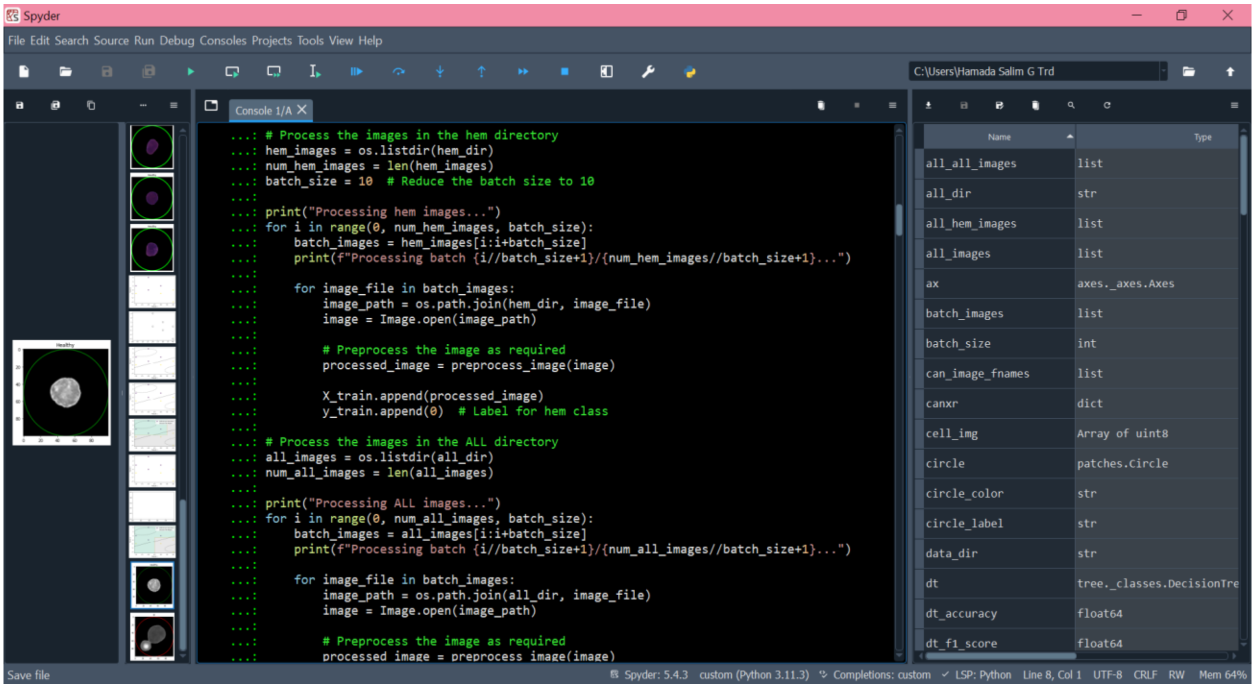
5.5. S.V.M. - Understanding the Code Workflow:

1. The code aims to accomplish the following steps:
2. Importing the necessary libraries: P.I.L. for image processing, numpy for numerical operations, sklearn.svm for the SVM model, os for file and directory operations, and matplotlib.pyplot for visualization.
3. Definition of the "preprocess_image" function, which takes an image as input and applies preprocessing steps such as converting it to grayscale, resizing it to (100, 100), and normalizing the pixel values.
4. Definition of the "extract_features" function, which takes a processed image as input and flattens it into a 1D array.
5. Creation of an instance of the SVM model using S.V.C.
6. Specification of the image directories for the training data: "hem_dir" for the "hem" class and "all_dir" for the "ALL" class.
7. Initialization of empty lists "X_train" and "y_train" to store the training data and labels.

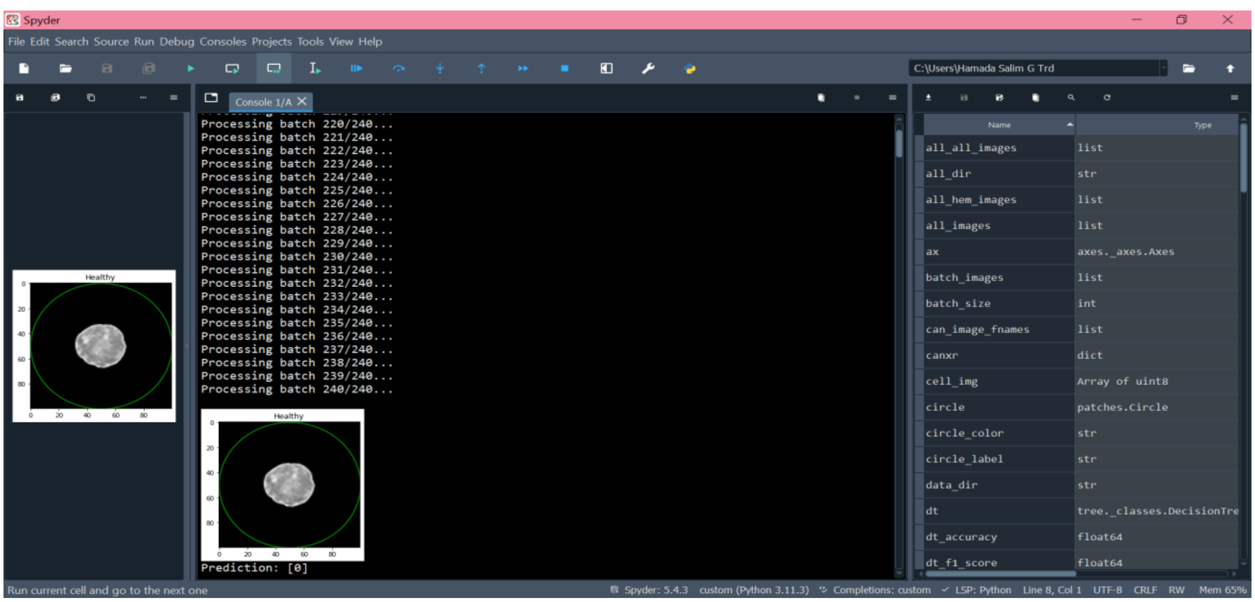


8. Processing of the images in the "hem" directory: Each image is opened, preprocessed, and then appended to "X_train". The label 0 (hem) is also appended to "y_train".

9. Processing of the images in the "ALL" directory: Similar to the previous step, the images are preprocessed and added to "X_train", while the label 1 (ALL) is added to "y_train".

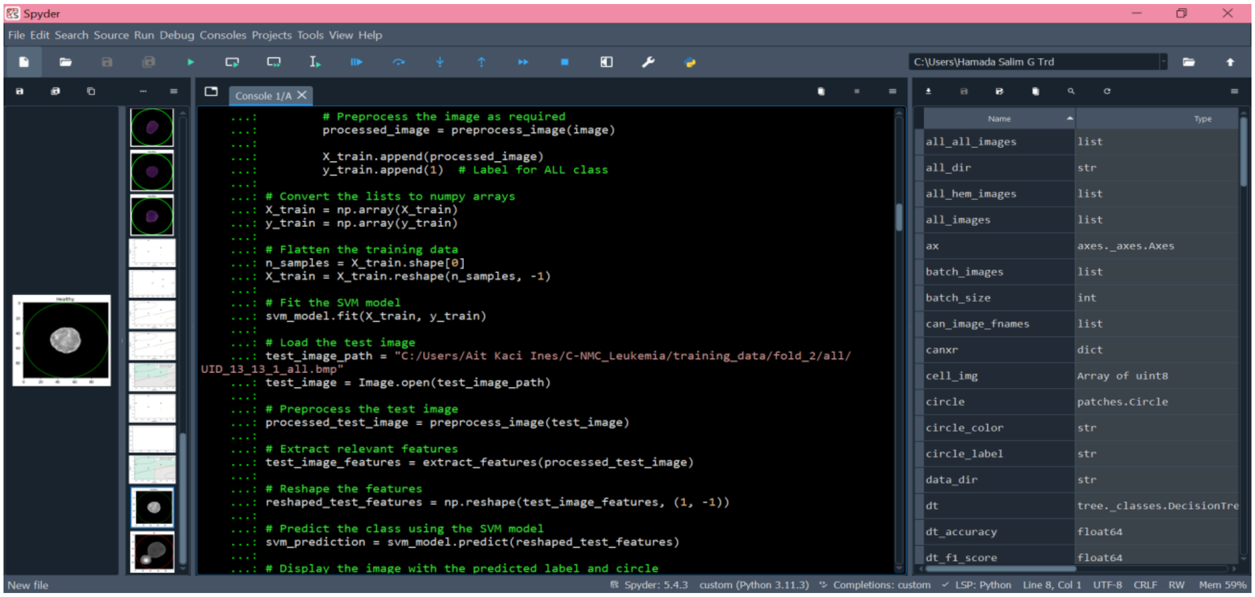


10. Conversion of "X_train" and "y_train" lists to numpy arrays.

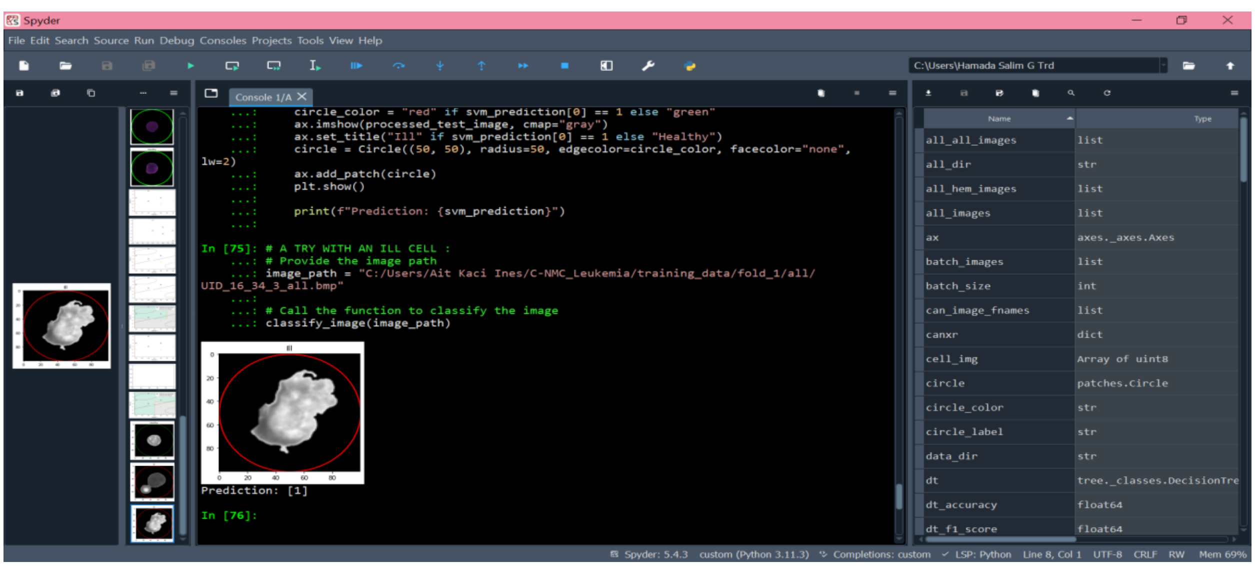


11. Reshaping the training data to a 2D array.
12. Fitting the SVM model using the training data and labels.

13. Loading a test image from the specified path.
14. Preprocessing of the test image using the "preprocess_image" function.
15. Extraction of relevant features from the processed test image using the "extract_features" function.
16. Reshaping the features to a 2D array.



17. Prediction of the class label for the test image using the SVM model.
18. Displaying the test image with the predicted label and a circle around the region of interest. The circle color is red for an "ALL" prediction and green for a "hem" prediction.
19. Printing the SVM prediction.

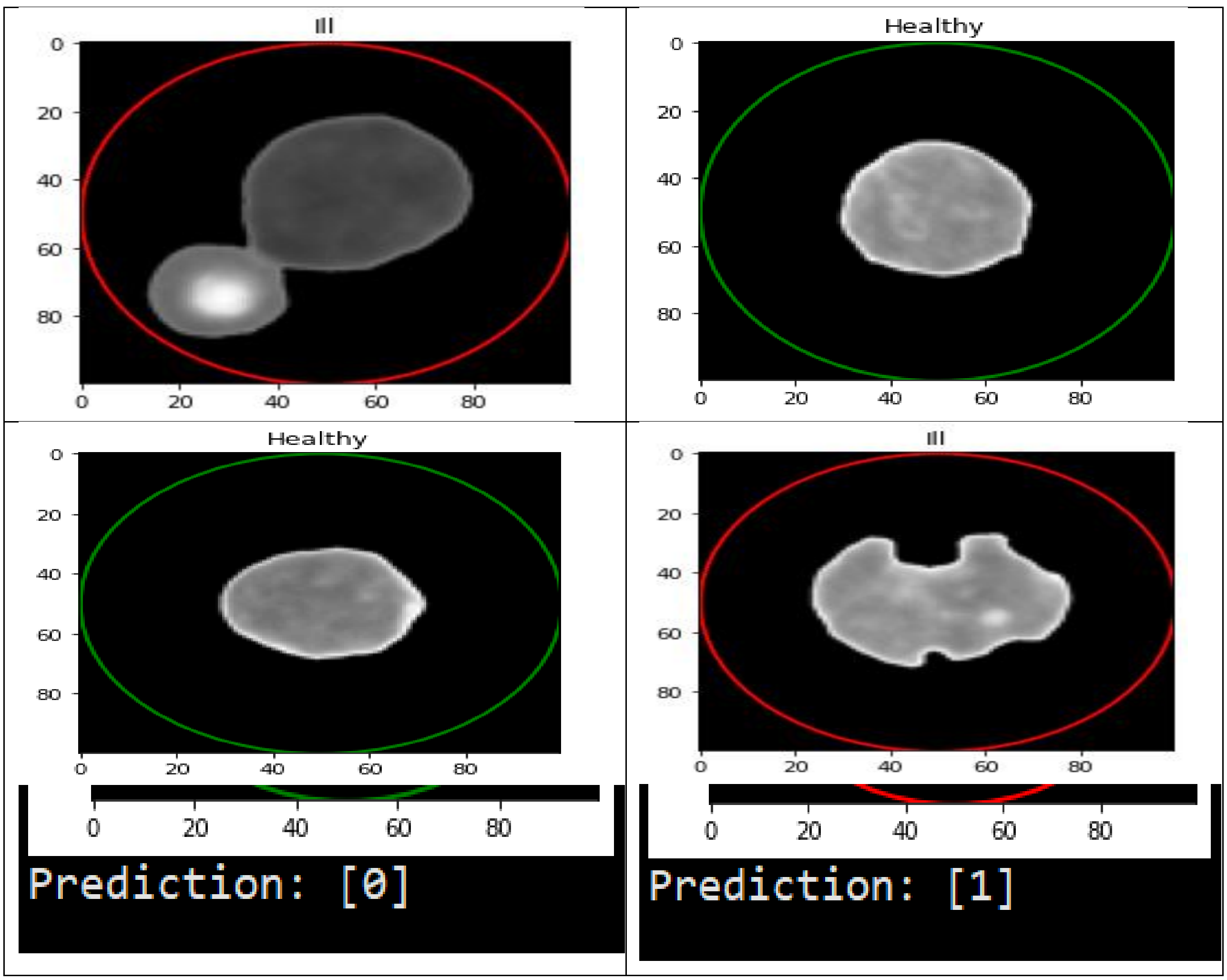


20. In conclusion, this code provides an understanding of the workflow involved in image preprocessing, training an SVM model, and predicting the class label of a test image using the trained model. Additionally, it includes a visual representation of the prediction.

5.5.1. Printing the SVM prediction.

In conclusion, this code provides an understanding of the workflow involved in image preprocessing, training an SVM model, and predicting the class label of a test image using the trained model. Additionally, it includes a visual representation of the prediction.

5.5.1.1. Results :



5.6. Conclusion

This chapter focuses on image classification with Convolutional Neural Networks for blood cancer detection. It highlights the importance of timely and accurate detection of blood cancer and how CNNs can enhance the diagnostic process. The chapter explains the code presented, demonstrating how CNNs can detect blood cancer from microscopic images of blood samples. The code follows a structured approach, including data preparation, model creation, and training and evaluation. Techniques such as data augmentation, image preprocessing, and the architecture of the CNNs model are explained. The code utilizes classes like ImageDataGenerator and Sequential to generate augmented image data and create a sequential model with convolutional, pooling, and fully connected layers.

The training and evaluation section covers model compilation, training over multiple epochs, and evaluation using validation data. Evaluation metrics like validation loss and accuracy are essential in assessing the model's performance in classifying blood cell images. The code's modular structure and clear separation of concerns are discussed, explaining the key modules, classes, and functions used for data preparation, model creation, and training and evaluation. These components train and evaluate a CNNs model for blood cancer detection. In conclusion, the chapter summarizes the code's main functionality, algorithms, and techniques. It highlights how the code addresses the need for automated blood cancer detection and its potential applications in assisting medical professionals with more accurate and efficient diagnoses. The chapter also includes screenshots showcasing the code's process, from loading libraries to converting validation data into a data frame.

Chapter 06 : General Conclusion

6.1. Contributions

In conclusion, blood tests and imaging studies are essential in diagnosing blood cancer. These methods, such as complete blood count (CBC) and imaging techniques like X-rays, CT scans, and MRI, provide valuable information about blood cell abnormalities and the presence of enlarged lymph nodes, aiding in identifying and diagnosing potential blood cancers. However, further advancements are required to improve diagnostic methods' accuracy, non-invasiveness, and cost-effectiveness. The integration of machine learning-based solutions shows great promise in addressing these needs, offering the potential to enhance diagnostic accuracy, develop non-invasive approaches, and optimize resource utilization in blood cancer detection. By continuing to invest in research and development, we can strive towards more effective and efficient diagnostic techniques for improved patient outcomes.

Moreover, we explore the web application architecture and design, shedding light on the technical aspects of the application. Using a client-server architecture comprises a front-end user interface, a back-end server, and a database for storing patient records. This design ensures a seamless user experience, enabling easy access to the application and efficient handling of patient data. The chapter highlights the significance of user-friendliness and intuitiveness in the application design, incorporating features like a login system, a dashboard for visualizing patient data, and a prediction module for blood cancer detection.

By covering crucial elements such as dataset description and data preprocessing, machine learning algorithm explanation and implementation, evaluation metrics and performance evaluation techniques, and web application architecture and design, our study serves as a comprehensive guide for understanding the methodology behind the study. Its insights and findings contribute to improving blood cancer detection through innovative approaches and technological advancements.

6.2. 5.2. Limitations

We would have liked to study other classification and prediction techniques for this disease, such as the fuzzy-based algorithms on fuzzy logic. In addition, perform a comparison between the different algorithms. We would have liked to study other classification and prediction techniques for this disease, such as the fuzzy-based algorithms on fuzzy logic. In addition, perform a comparison between the different algorithms.

6.3. Future work and perspectives

The challenges associated with integrating complex algorithms, ensuring data security and privacy, and validating the performance and reliability of these applications in real-world settings need to be addressed. The increasing medical vocabulary resulting from new diseases and symptoms presents a significant challenge. Relying on ontologies as an ineluctable design reduces ambiguity by providing a generic conceptualization of notions in medicine.

References

[1] Alizadeh, M., & Ghasemzadeh, M. (2021). A Review of Artificial Intelligence Applications in Blood Cancer Research. Journal of Medical Systems, 45(8), 1-12. doi: 10.1007/s10916-021-01798-7

[2] Alom, M. Z., Yakopcic, C., Hasan, M., Taha, T. M., & Asari, V. K. (2019). Machine Learning in Detection and Classification of Leukemia Blood Cell Images. IEEE Access, 7, 123190-123215. doi: 10.1109/access.2019.2930457

[3] National Cancer Institute. (2022). Can Artificial Intelligence Help See Cancer in New Ways? Retrieved from https://www.cancer.gov/news-events/cancer-currentsblog/2022/ai-cancer-imaging

[4] Gasparyan, A. Y., Ayvazyan, L., & Kitas, G. D. (2016). How to keep up to date with medical information using web-based tools. Advances in Medical Education and Practice, 7, 479-486. doi: 10.2147/amep.s108431

[5] Alharbi, A., Alqahtani, F., & Alqahtani, M. (2021). The Use of Artificial Intelligence in Literature Search and Review. Journal of Healthcare Engineering, 2021, 1-9. doi: 10.1155/2021/6679476

[6] World Health Organization.(2022). Cancer. Retrieved from a website //efaidnbmnnnibpcajpcglclefindmkaj/https://www.bms.com/assets/bms/us/en-us/pdf/DiseaseState-Info/blood-cancers-at-a-glance.pdf

[7] American Cancer Society. Leukemia. Retrieved from https://www.aplplatform.org/patient/

[8] American Cancer Society. Lymphoma. Retrieved from https://www.bcgsc.ca/news/importance-assessing-genetic-features-canine-b-cell-lymphomas

[9] American Cancer Society. Multiple Myeloma. Retrieved from https://www.haematologica.org/article/view/9819

[10] National Cancer Institute. Genetics of Cancer. Retrieved from https://www.cancer.gov/about-cancer/causes-prevention/genetics

[11] National Cancer Institute. Immunosuppression and Cancer Risk. Retrieved from https://www.cancer.gov/about-cancer/causes-prevention/risk/immunosuppression

[12] American Cancer Society. Signs and Symptoms of Blood Cancers. Retrieved from https://bloodcancer.org.uk/understanding-blood-cancer/blood-cancer-signs-symptoms/

[13] https://www.auntminnie.com/index.aspx?sec=ser&sub=def&pag=dis&ItemID=51465

[14] https://diagnosticcytogenetics.com

[15] Konstantina K., Themis P. E., Konstantinos P. E., Michalis V. K., Dimitrios I. F., (2015), "Machine learning applications in cancer prognosis and prediction", Computational and Structural Biotechnology Journal, Volume 13, Pages 8-17, ISSN 2001-0370,

[16] "Scikit-learn: Machine learning in Python" by Pedregosa et al. introduces scikit-learn, a powerful machine learning library in Python, and highlights its features and applications.

[17] Saito T., Rehmsmeier M. "The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets". PLoS One. 2015 Mar 4;10(3):e0118432. doi: 10.1371/journal.pone.0118432. PMID: 25738806; PMCID: PMC4349800.

[18] R. Kohavi. "A study of cross-validation and Bootstrap for accuracy estimation and model selection ",, (1995) Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI) , page 1137--1143.