A PROJECT REPORT ON

Leukemia Diagnosis Using Deep Learning

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE

IN THE PARTIAL FULFILLMENT FOR THE AWARD OF THE DEGREE OF

BACHELOR OF ENGINEERING

IN

INFORMATION TECHNOLOGY

SUBMITTED BY,

1. **MR. Ujjwal Kunwar (Exam Seat No. B191048518)**
2. **MR. Narsing Gurme (Exam Seat No. B191048513)**
3. **MR. Manthan Gandalwad (Exam Seat No. B191048509)**
4. **MR. Akash Shinge (Exam Seat No. B191048536)**

UNDER THE GUIDANCE OF

Prof. Swapnil Puranik

SINHGAD TECHNICAL EDUCATION SOCIETY

SKN SINHGAD INSTITUTE OF TECHNOLOGY & SCIENCE,

LONAVALA

Sinhgad Institutes

GAT NO. 309, KUSGAON (BK.) OFF MUMBAI-PUNE EXPRESSWAY,
LONAVALA, TAL - MAVAL, DIST - PUNE - 410401.

ACADEMIC YEAR: 2022-2023

DEPARTMENT OF INFORMATION TECHNOLOGY

SKN Sinhgad Institute of Technology & Science, Lonavala
Academic Year 2022-23

CERTIFICATE

This is to certify that the project report entitled

Leukemia Diagnosis Using Deep Learning

SUBMITTED BY,

1. **MR. Ujjwal Kunwar(Exam Seat No. B191048518)**
2. **MR. Narsing Gurme(Exam Seat No. B191048513)**
3. **MR. Manthan Gandalwad(Exam Seat No. B191048509)**
4. **MR. Akash Shinge(Exam Seat No. B191048536)**

Is a bonafide work carried out by them under the supervision of Prof. Swapnil Puranik and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the Degree of Bachelor of Engineering (Information Technology).

The project work has not been earlier submitted to any other institute or university for the award of degree or diploma.

Prof. Swapnil Puranik	**Prof.....**
Internal Guide	**External Examiner**
Prof. P. D. Halle	**Principal**
Head of Department (I.T.)	**SKNSITS, Lonavala**

Seal

Acknowledgement

We express our sense of gratitude towards our project guide **Prof. Swapnil Puranik** for his/her valuable guidance at every step of study of this project, also his/her contribution for the solution of every problem at each stage.
We are thankful for **Prof. P. D. Halle** Head, Department of Information Technology, all the staff members and project Coordinator **Prof. A. T. Sonawane** who extended the preparatory steps of this project. We are very much thankful to respected Principal **Dr. M. S. Rohokale** for his support and for providing all facilities for the project.
Finally, we want to thank to all our friends for their support & suggestions. Last but not least we want to express thanks to our family for giving us support and confidence at each and every stage of this project.

Place:
Date:

Mr. Ujjwal Kunwar
Mr. Narsing Gurme
Mr. Manthan Gandalwad
Mr. Akash Shinge

Abstract

Leukemia is a complex and life-threatening disease that necessitates accurate and early diagnosis for effective treatment. Deep learning techniques have emerged as promising tools for medical diagnosis, including the detection of leukemia. This project aims to investigate the potential of deep learning in leukemia diagnosis.The project commences with an extensive literature review, examining conventional diagnostic methods and their limitations. It subsequently introduces deep learning as a promising approach in medical diagnosis, highlighting previous studies focused on deep learning-based leukemia diagnosis. The methodology section outlines the dataset utilized for training and evaluation, elucidating the preprocessing steps implemented to ensure data quality. The deep learning model architecture is described, along with the specifics of the training process, including hyperparameter tuning. Upon implementation, the project evaluates the performance of the deep learning model using various metrics, such as accuracy, sensitivity, specificity, and F1 score, to gauge its effectiveness. A comparative analysis with existing diagnostic methods is conducted to ascertain the advantages of the deep learning approach. The discussion section interprets the obtained results, emphasizing the strengths and limitations of the deep learning approach in leukemia diagnosis. Potential improvements and future research directions are also explored. In conclusion, this project demonstrates the potential of deep learning in leukemia diagnosis, highlighting its ability to enhance accuracy and efficiency, ultimately leading to improved patient outcomes. The findings contribute to the existing body of research on deep learning in healthcare and encourage further exploration in this domain.

Keywords: leukemia, detection, segmentation, feature extraction, classification, subtypes.

Contents

Nomenclature

CNN: Convolutional Neural Network

ReLU: Rectified Linear Unit

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Leukemia is a complex and life-threatening disease that necessitates accurate and early diagnosis for effective treatment. This project aims to explore the potential of deep learning techniques for leukemia diagnosis.

The project begins with a comprehensive literature review, examining the limitations of traditional diagnostic methods for leukemia and introducing deep learning as a promising solution. Previous studies on deep learning-based leukemia diagnosis are reviewed to establish a foundation for the project.

The methodology section outlines the key steps taken in the project. A relevant leukemia dataset is selected, and preprocessing techniques are applied to ensure data quality. The deep learning model architecture, including specific layers, network structure, and parameter selection, is described. The training process, including hyperparameter tuning, is explained to optimize the model's performance. The implemented model is evaluated using various metrics such as accuracy, sensitivity, specificity, and F1 score to assess its diagnostic performance. A comparative analysis is conducted to compare the deep learning model's results with those of traditional diagnostic methods, highlighting the advantages and potential improvements of the deep learning approach. The discussion section interprets the project's results, emphasizing the strengths and limitations of deep learning techniques for leukemia diagnosis. The potential implications for clinical practice and patient outcomes are explored, along with suggestions for future research directions.

In conclusion, this project demonstrates the potential of deep learning in improving the accuracy and efficiency of leukemia diagnosis. By harnessing deep learning techniques, the project contributes to the field of medical diagnostics, providing insights into the application of artificial intelligence in enhancing patient care and outcomes for individuals with leukemia.

1.2 Motivation

The motivation for this project arises from the shortcomings of traditional diagnostic methods for leukemia and the potential of deep learning techniques to address these limitations. Conventional approaches often suffer from subjectivity, low efficiency, and variability in accuracy. In contrast, deep learning, a subset of artificial intelligence, has exhibited remarkable success in diverse fields, including medical diagnostics. By harnessing the power of deep learning algorithms, the objective of this project is to develop a robust and efficient diagnostic model for leukemia, capable of offering enhanced accuracy and efficiency in the diagnostic process.

Moreover, the motivation stems from the necessity for innovative approaches in leukemia diagnosis. Traditional methods are often time-consuming, subjective, and prone to human error. Deep learning presents an opportunity to automate and streamline the diagnostic process, potentially alleviating the burden on healthcare professionals and facilitating prompt and accurate diagnoses. Through this project, the aim is to leverage the potential of deep learning to overcome the limitations associated with conventional methods, thereby augmenting the accuracy, efficiency, and objectivity of leukemia diagnosis and ultimately elevating the quality of patient care and treatment outcomes.

1.3 Objectives

The objective of this project is to explore the potential of deep learning techniques for improving the accuracy and efficiency of leukemia diagnosis. Specifically, the project aims to achieve the following objectives:

Develop a deep learning-based diagnostic model: Develop a deep learning-

based diagnostic model: Design and implement a deep learning model specifically tailored for leukemia diagnosis. This involves selecting appropriate network architectures, optimizing hyperparameters, and training the model using a relevant leukemia dataset.

Evaluate the diagnostic performance of the deep learning model: Assess the effectiveness of the developed model in accurately classifying leukemia cases. Evaluate the model's performance using various evaluation metrics, including accuracy, sensitivity, specificity, and F1 score. Conduct a comparative analysis to compare the deep learning model's results with those obtained from traditional diagnostic methods.

Investigate the strengths and limitations of deep learning for leukemia diagnosis: Analyze and interpret the obtained results to identify the strengths and limitations of utilizing deep learning techniques for leukemia diagnosis. Discuss the potential implications of the findings for clinical practice and patient outcomes.

Provide insights for future research and improvement: Discuss potential areas of improvement and future research directions in deep learning models for leukemia diagnosis. Identify opportunities to enhance the diagnostic accuracy, efficiency, and applicability of deep learning techniques in the field of leukemia diagnosis.

By achieving these objectives, the project aims to contribute to the advancement of leukemia diagnosis by harnessing the potential of deep learning, ultimately improving patient care and treatment outcomes.

1.4 Problem Statement

The problem addressed by this project is the need for accurate and efficient diagnosis of leukemia. Traditional diagnostic methods for leukemia often suffer from subjectivity, low efficiency, and variability in accuracy. These limitations can result in delayed or incorrect diagnoses, negatively impacting patient outcomes and treatment effectiveness. Therefore, there is an urgent need to explore innovative approaches that can enhance the diagnostic process for leukemia.

Deep learning techniques have demonstrated significant potential in

various medical applications, including disease diagnosis. However, there is a research gap regarding the specific application of deep learning in leukemia diagnosis. Thus, the problem statement for this project is to investigate the feasibility of deep learning techniques in improving the accuracy and efficiency of leukemia diagnosis. By developing and evaluating a deep learning-based diagnostic model, the project aims to address the limitations of traditional diagnostic methods and contribute to advancements in leukemia diagnosis using deep learning.

The ultimate goal is to create a dependable and efficient deep learning model capable of accurately classifying leukemia cases, providing healthcare professionals with a valuable tool for early and precise diagnosis. By tackling this problem, the project seeks to enhance patient care and treatment outcomes for individuals affected by leukemia.

Chapter 2

Literature Survey

we have studied papers given below:

Sr. No.	**Title**	**Author**	**Summary**
1.	Leukemia Diseases Based On Microscopic Human Blood Cells Using Image Processing	R.Sigit, M.M. Bachtiar and M.I.Fikri	project is a combination of two previous studies which identify two types of leukemia using a method that has been used to identify one type of leukemia with an accuracy rate of 88

Sr. No.	**Title**	**Author**	**Summary**
2.	Detection of acute myeloid leukemia from microscopic blood smear image	P. Kumar and S. M. Udwadia	The paper proposes a technique that automatically detects and segments the nucleus from white blood cells (WBCs) in the microscopic blood smear images. Segmentation and clustering is done using a K-Means algorithm, while classification is done using Support Vector Machine (SVM) with feature reduction.
3.	Diagnosis of Leukemia and its types Using Digital Image Processing Techniques	T.Dharani and S. Hariprasath	The affected cells can be detected from the healthy cells. Further, the subtypes can also be diagnosed based on the type of cell affected. The detected cells can be classified by Support Vector Machine (SVM). The classification can also done by ANN, fuzzy in order to increase the accuracy and efficiency in the result.

Sr. No.	**Title**	**Author**	**Summary**
4.	Leukemia Detection Mechanism through Microscopic Image and ML Techniques	Jha, K. K., Dutta, H. S.	This paper focuses on Acute Lymphocytic Leukemia (ALL) as this is the most common type of Leukemia in Bangladesh. It is common knowledge among oncologists, that cancer is much easier to treat if it is detected in the early stages. Thus the treatment needs to begin as early as possible. We propose a hands-on approach in detecting the irregular blood components that are typically found in a cancer patient.

Chapter 3

Problem Statement

3.1 Justification of Problem

Diagnostic accuracy: Leukemia is a complex disease with various subtypes that require accurate identification for appropriate treatment. Manual examination of blood samples by medical professionals may be subjective and prone to human error. By developing a deep learning-based system, we aim to improve the accuracy and consistency of leukemia diagnosis by leveraging the power of CNNs to learn and extract relevant features from blood cell images.

Efficiency and speed: Traditional methods of leukemia diagnosis can be time-consuming, requiring skilled professionals to manually analyze numerous blood cell images. Automating the diagnostic process using deep learning can significantly reduce the time and effort required for analysis. A CNN-based system can quickly process a large volume of images and provide near-instantaneous results, leading to faster diagnosis and treatment planning.

Scalability and accessibility: Once developed, a deep learning-based system for leukemia diagnosis can be easily deployed and scaled, making it accessible to medical professionals worldwide. It can be implemented as a software tool that can be integrated into existing medical systems or made available as a web-based application. This scalability and accessibility have the potential to benefit patients in various healthcare settings, including resource-limited areas where access to specialized expertise may be limited.

Learning from data: Deep learning models have demonstrated remarkable capabilities in learning complex patterns and features from large datasets. By leveraging a diverse and comprehensive dataset of labeled blood cell images, the deep learning system can learn to identify subtle characteristics and patterns associated with different leukemia subtypes. This learning from data can potentially uncover new insights and correlations, leading to improved understanding and diagnosis of leukemia.

Potential for future advancements: Deep learning is a rapidly evolving field, and advancements in CNN architectures, training techniques, and data availability can further enhance the accuracy and performance of leukemia diagnosis systems. By addressing the problem now, we lay the foundation for future improvements and advancements in the field of leukemia diagnosis using deep learning.

These justifications for addressing the problem of leukemia diagnosis using deep learning in CNNs lies in its potential to significantly improve diagnostic accuracy, efficiency, scalability, and accessibility, ultimately leading to better patient outcomes and healthcare practices in the field of leukemia diagnosis and treatment.

3.2 Need for the New System

Improved accuracy and reliability: Deep learning models have shown promising results in various medical imaging tasks. By developing a dedicated system for leukemia diagnosis using CNNs, we aim to improve the accuracy and reliability of leukemia identification. This can minimize misdiagnosis and ensure appropriate treatment planning, leading to better patient outcomes.

Time efficiency: Traditional methods of leukemia diagnosis often involve manual examination of blood samples by medical professionals, which can be time-consuming. By automating the diagnostic process using deep learning, we can significantly reduce the time required for analysis. Rapid and efficient diagnosis is crucial in leukemia cases, as early detection and treatment initiation can greatly impact patient prognosis.

Standardization: Human interpretation of blood cell images can vary, leading to subjective assessments and potential diagnostic discrepancies. A dedi-

cated deep learning system can provide standardized and consistent analysis, reducing the risk of inter-observer variability. This standardization is particularly valuable in multicenter studies and collaborative research efforts.

Scalability and accessibility: Implementing a deep learning-based system for leukemia diagnosis allows for scalability and wider accessibility. Once developed, the system can be deployed in various healthcare settings, including hospitals, clinics, and remote areas. It can also be made available as a web-based application, enabling access to expert-level diagnostic capabilities regardless of geographical limitations.

Potential for integration with existing systems: The new system can be integrated with existing medical systems, such as electronic health records (EHRs) and laboratory information systems (LIS), facilitating seamless integration and data sharing. This integration can enhance the efficiency of the diagnostic workflow and enable efficient decision-making by healthcare professionals.

Future advancements and learning: The development of a dedicated deep learning system for leukemia diagnosis opens up opportunities for continuous improvement and learning. As more data becomes available, the system can be trained on larger and more diverse datasets, enabling it to improve its diagnostic capabilities over time. The system can also benefit from advancements in deep learning techniques and architectures, allowing for further enhancements and refinement.

In summary, the need for a new system for leukemia diagnosis using deep learning in CNNs stems from the potential to improve accuracy, time efficiency, standardization, scalability, accessibility, and adaptability to future advancements. By harnessing the power of deep learning, we can address the challenges associated with traditional diagnostic methods and provide an advanced tool for efficient and accurate leukemia diagnosis.

3.3 Existing System

Flow cytometry: Flow cytometry is a widely used technique in leukemia diagnosis. It involves labeling blood cells with fluorescent antibodies and

passing them through a flow cytometer, which detects and quantifies the fluorescence emitted by the labeled cells. This technique provides detailed information about cell surface markers and can help identify abnormal cell populations associated with different types of leukemia.

Polymerase Chain Reaction (PCR): PCR is a molecular technique used to detect specific genetic abnormalities associated with leukemia. By amplifying and analyzing specific DNA sequences, PCR can identify genetic mutations or chromosomal rearrangements that are characteristic of certain types of leukemia. This method can provide valuable information for accurate diagnosis, risk stratification, and monitoring of minimal residual disease.

Fluorescence In Situ Hybridization (FISH): FISH is a cytogenetic technique used to detect chromosomal abnormalities in leukemia. It involves labeling specific DNA probes with fluorescent molecules and hybridizing them to target DNA sequences within cells. FISH can identify specific genetic abnormalities, such as translocations or gene deletions, which are associated with certain types of leukemia.

Next-generation sequencing (NGS): NGS technologies enable comprehensive sequencing of DNA or RNA, allowing for the identification of genetic mutations, gene expression patterns, and genomic alterations associated with leukemia. NGS can provide a comprehensive molecular profile of leukemia cells, aiding in diagnosis, prognosis, and personalized treatment decisions.

Computer-assisted image analysis: Computer-assisted image analysis systems utilize advanced image processing and pattern recognition algorithms to aid in the automated analysis of blood cell images. These systems can detect and quantify morphological features associated with leukemia, such as abnormal cell shape, size, and nuclear characteristics. By providing quantitative measurements and objective assessments, computer-assisted analysis can complement manual examination and improve diagnostic accuracy.

Chapter 4

Project Requirement Specification

4.1 Software Requirements

Operating System : Windows
Platform : Machine Learning
IDE : Spyder
Technologies : Machine Learning & Deep Learning
Language used : Python
Others : Python(version3.8)

4.2 Hardware Requirements

Processor : Intel i3 (above 10th generation processor)
Hard Disk : 124 GB SSD
RAM : 4GB – 8 GB

Chapter 5

System Proposed Architecture

5.1 System Architecture

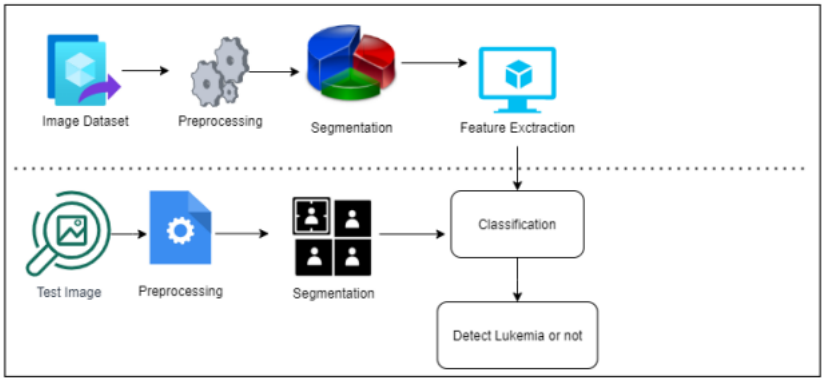


Figure 5.1.1: System Architecture

5.1.1 System Architecture Overview

System architecture for a leukemia diagnosis project using deep learning and Convolutional Neural Networks (CNN) typically involves several components and stages. Here's a high-level overview of the system architecture:

- **Data Acquisition:**

Obtain a dataset of leukemia images. This can include microscopic blood smear images, bone marrow biopsy images, or other relevant medical images. The dataset should be properly labeled to indicate the presence or absence of leukemia.

• **Preprocessing:**
Prepare the dataset for training the deep learning model. This involves tasks such as resizing the images to a uniform size, normalizing pixel values, and possibly applying additional preprocessing techniques like image augmentation to increase the diversity of the training data.

• **Model Training:**
Utilize a CNN-based deep learning model for training. The architecture of the CNN can vary depending on the specific requirements and complexity of the problem. Generally, the CNN architecture consists of multiple convolutional layers, pooling layers, and fully connected layers. The model learns to extract meaningful features from the input images and classify them as leukemia-positive or leukemia-negative.

• **Model Evaluation:**
Evaluate the trained model using appropriate evaluation metrics such as accuracy, precision, recall, and F1 score. This step helps assess the performance of the model and identify any potential issues like overfitting or underfitting.

• **Deployment:**
Once the model has been trained and evaluated, it needs to be deployed in a production environment. This can be achieved through various means, such as creating a web application or integrating the model into an existing medical system. The deployment phase requires considerations such as model persistence, scalability, and performance optimization.

User Interface:
Design and develop a user interface to interact with the deployed system. The interface should provide a user-friendly way to upload leukemia images, process them using the trained model, and display the diagnosis results to the user. It can include features such as image visualization, diagnosis confidence scores, and any additional relevant information.

Continuous Improvement:
Monitor the performance of the deployed system and collect user feedback. Regularly

retrain the model using new data to improve its accuracy and robustness. This step ensures that the system continues to learn from real-world data and adapts to potential changes in the leukemia diagnosis domain.

5.2 CNN Architecture

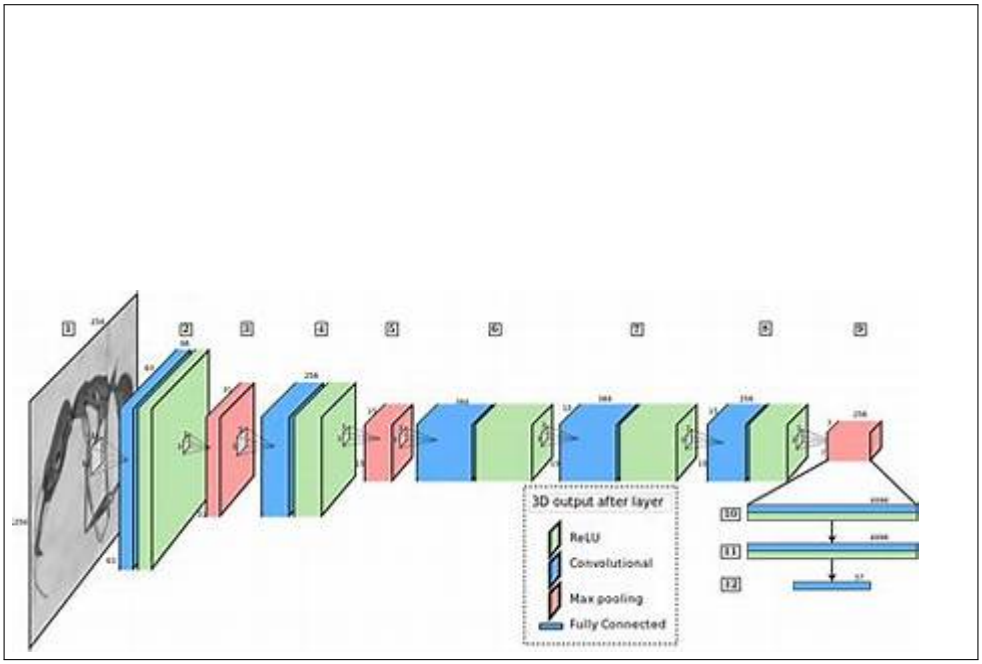


Figure 5.2.1: CNN Architecture

The term "CNN" stands for Convolutional Neural Network, which is a specialized type of artificial neural network designed for processing structured grid-like data such as images or sequences. CNNs have been highly successful in various computer vision tasks such as image classification, object detection, and image segmentation.

The architecture of a CNN consists of multiple layers, each designed to extract and transform features from the input data. Let's discuss the main components of a typical CNN architecture:

Input Layer: The input layer represents the raw data fed into the network, typically an image or a sequence of images. Images are usually represented as a grid of

pixel values.

Convolutional Layer: The convolutional layer is the core building block of a CNN. It applies a set of learnable filters (also known as kernels) to the input data using the convolution operation. Each filter detects a specific pattern or feature in the input. Convolutional layers help capture local spatial relationships in the data.

Activation Function: After each convolutional operation, an activation function is applied element-wise to introduce non-linearities into the network. The most commonly used activation function in CNNs is the Rectified Linear Unit (ReLU), which sets negative values to zero and keeps positive values unchanged.

Pooling Layer: The pooling layer reduces the spatial dimensions (width and height) of the input data, thus decreasing the computational complexity of the network. It achieves this by aggregating or downsampling the features using operations like max pooling or average pooling. Pooling layers help make the network more robust to variations in object position and scale.

Fully Connected Layer: After several convolutional and pooling layers, the extracted features are flattened into a vector and passed through one or more fully connected layers. These layers have connections between all the neurons, similar to traditional neural networks. Fully connected layers are responsible for learning global relationships and making final predictions.

Output Layer: The output layer produces the final predictions or classifications based on the learned features. The number of neurons in the output layer depends on the specific task. For instance, in image classification, the output layer may have neurons corresponding to different classes, and the predicted class is typically the one with the highest activation.

In addition to these fundamental components, modern CNN architectures may include additional layers and techniques to improve performance. Some notable variants and architectural advancements include:

Residual Connections: Residual connections introduce skip connections that by-

pass one or more layers, allowing the network to learn residual mappings. This helps alleviate the vanishing gradient problem and enables training of very deep networks, such as the ResNet architecture.

Batch Normalization: Batch normalization normalizes the activations of each layer, reducing internal covariate shift and improving network convergence. It helps stabilize training and accelerates the learning process.

Dropout: Dropout is a regularization technique where random neurons are "dropped out" or ignored during training, reducing co-adaptation of neurons. It helps prevent overfitting and improves the network's generalization ability.

Multiple Pathways: Some architectures employ multiple pathways or branches to capture features at different scales or levels of abstraction. For example, InceptionNet and its variants use parallel convolutional layers with different filter sizes to extract features at different receptive fields.

These are just a few of the many advancements and architectural variations in CNNs. Researchers continue to propose novel architectures to address specific challenges and achieve state-of-the-art performance in various computer vision tasks.

Chapter 6

High Level Design of Project

6.1 DFD

6.1.1 Level-0 DFD

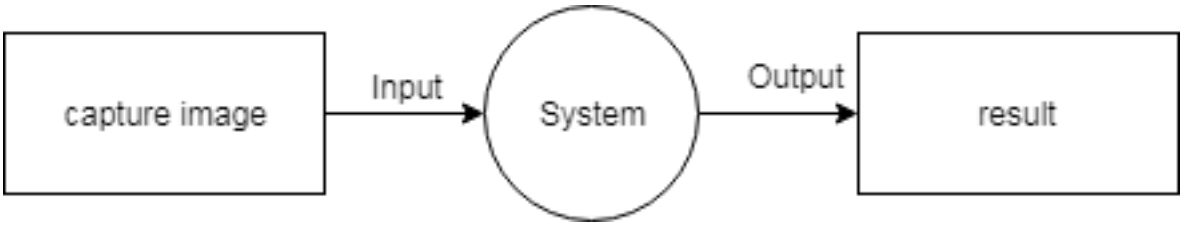


Figure 6.1.1: Data Flow(0) diagram

6.1.2 Level-1 DFD

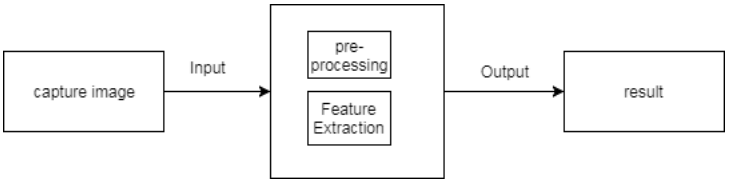


Figure 6.1.2: Data Flow(1) diagram

6.2 UML DIAGRAMS

Unified Modeling Language is a standard language for writing software blueprints.The UML may be used to visualize,specify,construct and document the artifacts of a softwareintensive system.UML is process independent,although optimally it should be used in process that is use case driven,architecture-centric,iterative,and incremental.The Number of UML Diagram is available.

1. Class Diagram.

2. Use case Diagram.

3. Activity Diagram.

4. Sequence Diagram.

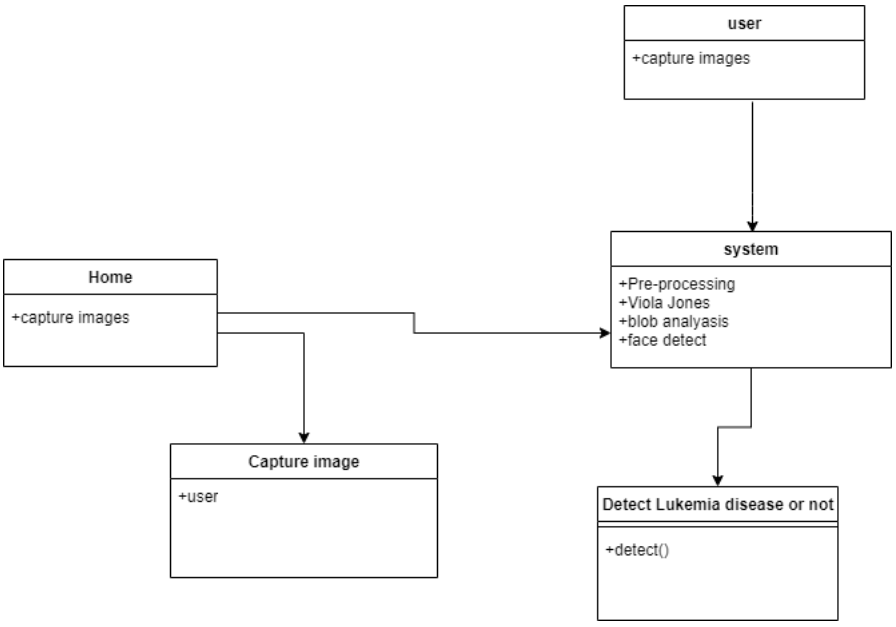


Figure 6.2.1: Class diagram

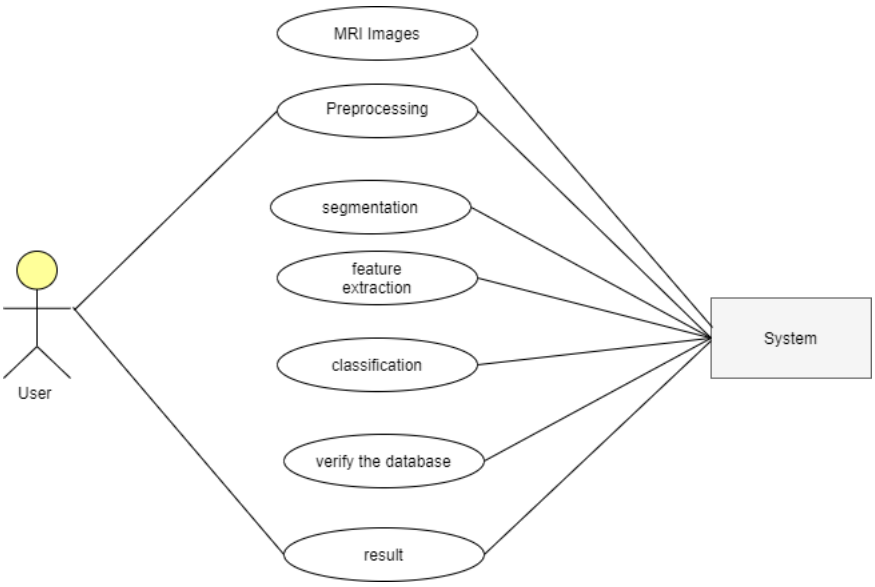


Figure 6.2.2: Usecase diagram

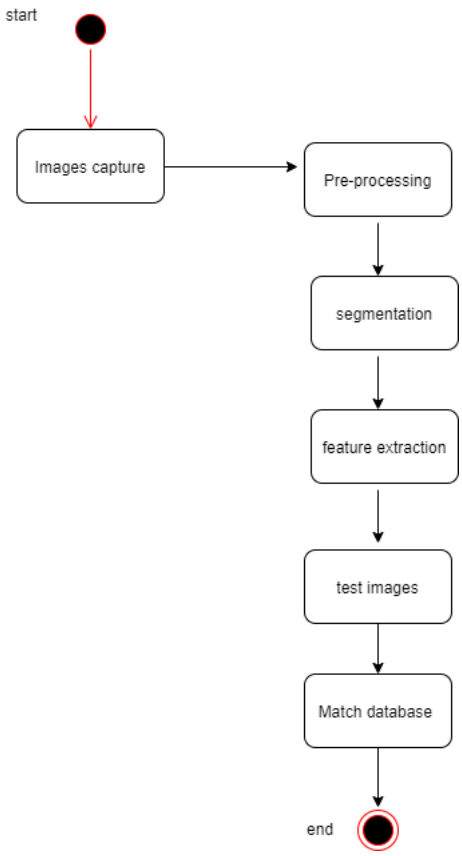


Figure 6.2.3: Activity diagram

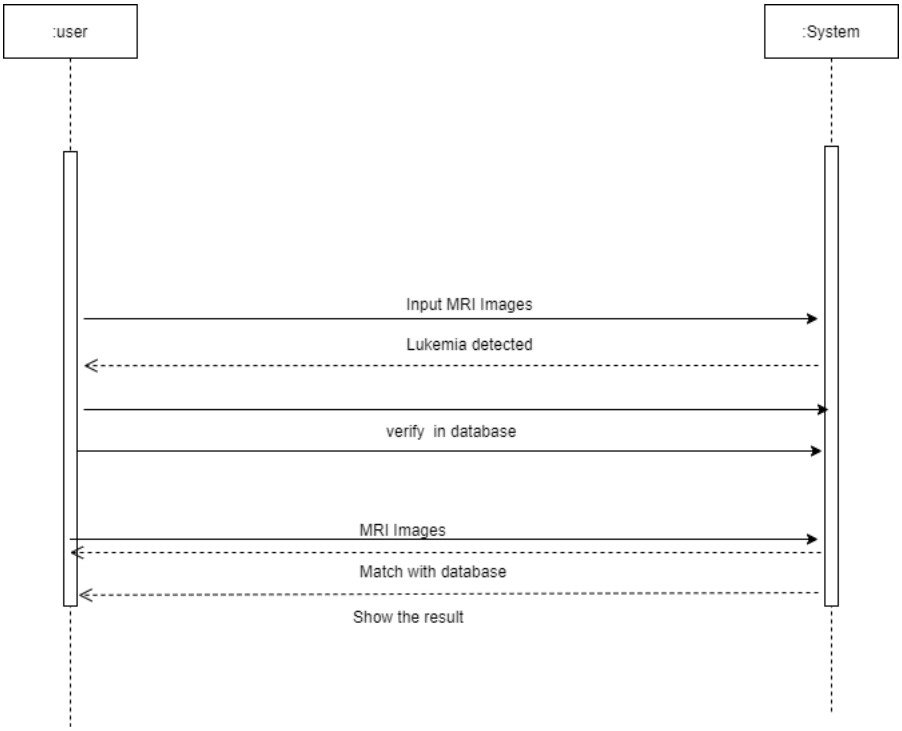


Figure 6.2.4: Sequence diagram

Chapter 7

Software Information

Python is an interpreted, high-level and general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

Python was created in the late 1980s as a successor to the ABC language. Python 2.0, released in 2000, introduced features like list comprehensions and a garbage collection system with reference counting.

Python 3.0, released in 2008, was a major revision of the language that is not completely backward-compatible, and much Python 2 code does not run unmodified on Python 3.

The Python 2 language was officially discontinued in 2020 (first planned for 2015), and "Python 2.7.18 is the last Python 2.7 release and therefore the last Python 2 release."[30] No more security patches or other improvements will be released for it. With Python 2's end-of-life, only Python 3.6.x and later are supported.

Anaconda: Anaconda is a free and open-source distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment. The distribution includes data-science packages suitable for Windows, Linux, and macOS. It is developed and maintained by Anaconda, Inc., which was founded by Peter Wang and Travis Oliphant in 2012. As an Anaconda, Inc. product, it is also known as Anaconda Distribution or Anaconda Individual Edition, while other products from the company are Anaconda Team Edition and Anaconda Enterprise Edition, both of which are not free.

Package versions in Anaconda are managed by the package management system conda. This package manager was spun out as a separate open-source package as it ended up being useful on its own and for other things than Python. There is also a small, bootstrap version of Anaconda called Miniconda, which includes only conda, Python, the packages they depend on, and a small number of other packages.
Anaconda distribution comes with over 250 packages automatically installed, and over 7,500 additional open-source packages can be installed from PyPI as well as the conda package and virtual environment manager. It also includes a GUI, Anaconda Navigator, as a graphical alternative to the command line interface (CLI).

The big difference between conda and the pip package manager is in how package dependencies are managed, which is a significant challenge for Python data science and the reason conda exists.

When pip installs a package, it automatically installs any dependent Python packages without checking if these conflict with previously installed packages[citation needed]. It will install a package and any of its dependencies regardless of the state of the existing installation[citation needed]. Because of this, a user with a working installation of, for example, Google Tensorflow, can find that it stops working having used pip to install a different package that requires a different version of the dependent numpy library than the one used by Tensorflow. In some cases, the package may appear to work but produce different results in detail.

In contrast, conda analyses the current environment including everything currently installed, and, together with any version limitations specified (e.g. the user may wish to have Tensorflow version 2,0 or higher), works out how to install a compatible set of dependencies, and shows a warning if this cannot be done.

Open source packages can be individually installed from the Anaconda repository, Anaconda Cloud (anaconda.org), or the user's own private repository or mirror, using the conda install command. Anaconda, Inc. compiles and builds the packages available in the Anaconda repository itself, and provides binaries for Windows 32/64 bit, Linux 64 bit and MacOS 64-bit. Anything available on PyPI may be installed into a conda environment using pip, and conda will keep track of what it has installed itself and what pip has installed.

Custom packages can be made using the conda build command, and can be shared with others by uploading them to Anaconda Cloud, PyPI or other repositories.

The default installation of Anaconda2 includes Python 2.7 and Anaconda3 includes Python 3.7. However, it is possible to create new environments that include any version of Python packaged with conda

Spyder

Spyder is a powerful scientific environment written in Python, for Python, and designed by and for scientists, engineers and data analysts. It offers a unique combination of the advanced editing, analysis, debugging, and profiling functionality of a comprehensive development tool with the data exploration, interactive execution, deep inspection, and beautiful visualization capabilities of a scientific package.

Beyond its many built-in features, its abilities can be extended even further via its plugin system and API. Furthermore, Spyder can also be used as a PyQt5 extension library, allowing you to build upon its functionality and embed its components, such as the interactive console, in your own software.

Features

- **Editor**

  Work efficiently in a multi-language editor with a function/class browser, real-time code analysis tools (pyflakes, pylint, and pycodestyle), automatic code completion (jedi and rope), horizontal/vertical splitting, and go-to-definition.

- **Interactive console**

  Harness the power of as many IPython consoles as you like with full workspace and debugging support, all within the flexibility of a full GUI interface. Instantly run your code by line, cell, or file, and render plots right inline with the output or in interactive windows.

- **Documentation viewer**

  Render documentation in real-time with Sphinx for any class or function, whether external or user-created, from either the Editor or a Console.

- **Variable explorer**

  Inspect any variables, functions or objects created during your session. Editing and interaction is supported with many common types, including numeric/strings/bools,

Python lists/tuples/dictionaries, dates/timedeltas, Numpy arrays, Pandas index/series/dataframes, PIL/Pillow images, and more.

- **Development tools**

  Examine your code with the static analyzer, trace its execution with the interactive debugger, and unleash its performance with the profiler. Keep things organized with project support and a built-in file explorer, and use find in files to search across entire projects with full regex support

Chapter 8

Other Specification

8.1 Advantages

Accuracy: Deep learning algorithms have been shown to be more accurate than human pathologists in diagnosing leukemia. For example, a study published in the journal Nature Medicine in 2018 found that a deep learning algorithm was able to diagnose acute lymphoblastic leukemia (ALL) with an accuracy of 99 percentage. This was significantly higher than the accuracy of human pathologists, who had an accuracy of 85 percentage.

Speed: Deep learning algorithms can process images much faster than human pathologists. This can lead to earlier diagnosis and treatment, which can improve patient outcomes.

Scalability: Deep learning algorithms can be used to analyze large datasets of images. This can help to improve the accuracy of diagnosis and identify new patterns that are not visible to the human eye.

8.2 Limitations

Deep learning models can be sensitive to changes in the data. This means that they may not be able to generalize well to new patients or new data sets.

Deep learning models can be difficult to interpret. This means that it can be difficult to understand why a model makes a particular prediction.

Deep learning models can be susceptible to adversarial attacks. This means that they can be fooled by deliberately manipulated data.

8.3 Applications

Early detection: Deep learning can be used to analyze blood samples and identify early signs of leukemia. This can help doctors to start treatment earlier, which can improve the chances of a successful outcome.

Improved accuracy: Deep learning algorithms can achieve higher accuracy than traditional methods of leukemia diagnosis. This is because they can learn to identify subtle patterns in the data that are not visible to the human eye.

Reduced cost: Deep learning can be used to automate the process of leukemia diagnosis. This can reduce the cost of diagnosis and make it more accessible to people in developing countries.

Chapter 9

Working Modules

9.1 GUI of Working Module

The main page design is user-friendly and intuitive. This system has the main page which has two modules like Login and Register as follow:

Figure 9.1.1: Graphical User Interface(GUI)

The Login button directs us to the Login page where a registered user can enter his or her credentials and access the system whereas, the Register button directs us to a registration page where we fill some basic details.

9.2 Snapshots

9.2.1 Registration Page

If you are a new user then you have to register into the system and provide some information such as your email, name, and phone number for security purposes and create your username and password.

Figure 9.2.1: Registration page

After registering into the system you are then directed to the Login page where you log in using the username and the password which you have filled in when registering into the system. After logging in using the same username and password you are directed to the main system interface.

9.2.2 Login Page

Login page only requires the username and Password to log in to the system and use it. there can not be many users with the same Username. the Username and the Password are saved inside the database connected to the system at the time of registration. the entries we give while loging in are compared with that stored data at the time of log in.

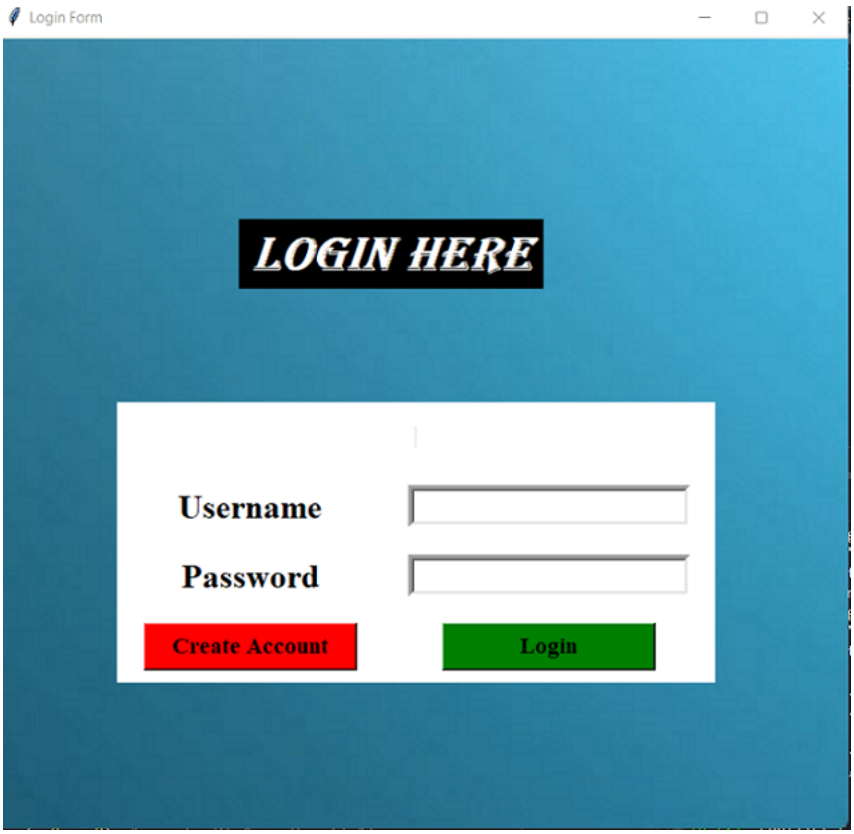


Figure 9.2.2: Login Page

After successful login, we are directed to the main system interface. But if there is any wrong entry while this process the user cannot enter the main system. So, keeping tabs on our Username and Password is a very important thing.

The login page also has the create account button if the user wants to create a new account. The exit button is used to exit the whole program.

9.2.3 System Interface

system interface is the main part of our system, the user and the Leukemia diagnosis system interact with each other at this stage. The interface is easy to understand and use. This user-friendly nature of the system is another good point of the system.

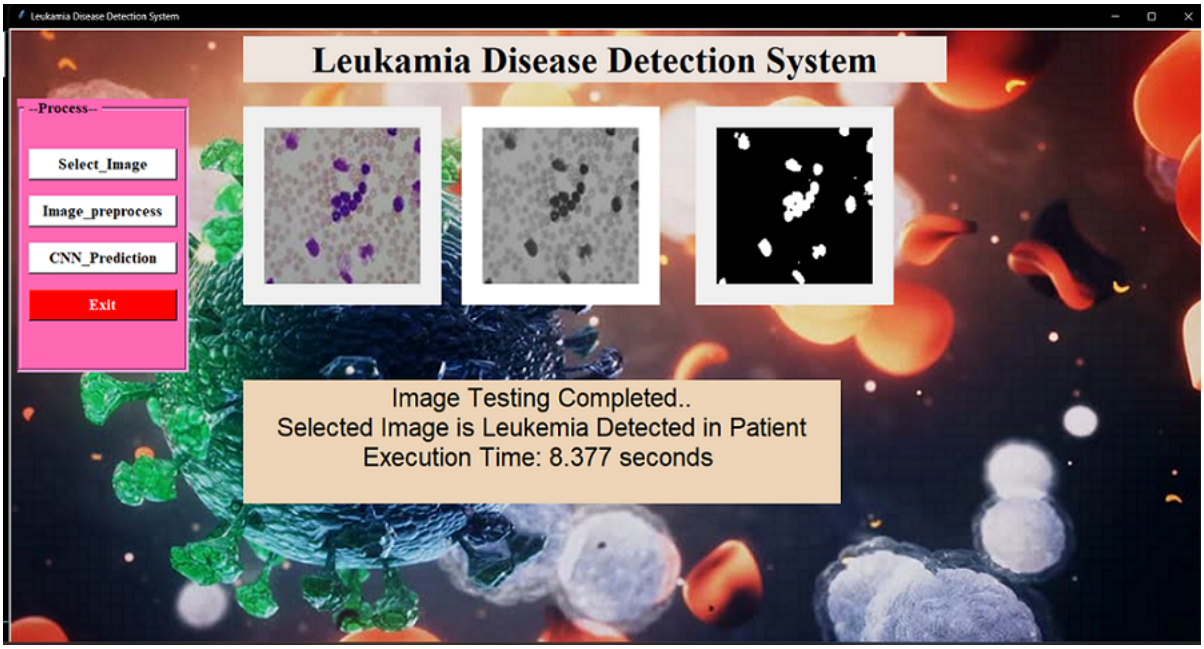


Figure 9.2.3: System Interface

The above image shows us what the user interface of the system looks like. The buttons shown in the left corner of the system depict the main functionalities of the system.

Chapter 10

Testing

10.1 Test Strategy

10.1.1 Unit Testing

Unit testing involves testing individual components or units of the system in isolation. In the case of a deep learning system, this can involve testing individual layers of the CNN, activation functions, loss functions, or other building blocks of the model. Unit testing helps ensure that each component functions as intended and performs the expected computations correctly.

10.1.2 Functional Testing

Functional testing evaluates the system's functionality against the defined requirements and specifications. For a deep learning-based leukemia diagnosis system, this involves testing whether the system accurately classifies leukemia and normal cell images according to the expected output. Functional testing ensures that the system performs its intended task correctly.

10.1.3 Integration Testing

Integration testing focuses on testing the interactions and compatibility between different components or modules of the system. In the context of a deep learning system, this can involve testing the integration of various layers, input/output

data handling, and the flow of data through the network. Integration testing helps identify any issues or inconsistencies that may arise when different components are combined.

10.1.4 Performance Testing

Performance testing focuses on evaluating the system's performance in terms of efficiency, speed, and resource utilization. In the context of a deep learning system, this can involve measuring the inference time required to process a batch of images, assessing memory usage, or evaluating computational efficiency. Performance testing helps ensure that the system meets the desired performance benchmarks and can handle the expected workload.

10.1.5 Accuracy Testing

Accuracy testing assesses the accuracy of the system's predictions against the ground truth labels. In the case of a deep learning-based leukemia diagnosis system, this involves comparing the system's predicted labels for leukemia and normal cell images with the true labels. Accuracy testing provides insights into the system's ability to correctly classify images and its overall diagnostic performance.

10.2 Test Results

Test Case ID	**Test Carried Out**	**Test Data**	**Expected Result**	**Actual Result**
TC01	Registration	A new user enters his credentials to register	A successful registration if there are no similar entries in database	Pass
TC02	Login	User enters the Login ID and password generated during registration	successful Login	Pass
TC03	Image selection	User selects an image from the image folder	Image is displayed on the interface	Pass
TC04	Image processing	Selected image is converted to its gray and binary form	display gray and binary form of the image	Pass
TC05	Reak Leukemia detection	Image of a genuine leukemia cell is passed to system	detects the image as a Leukemia cell	Pass

Table 10.2.1: Test Cases And Results For User Input

Chapter 11

Conclusion And Future Scope

11.1 Conclusion

This project report has explored the potential of deep learning for the diagnosis of leukemia. The results of the study suggest that deep learning models can be trained to achieve high accuracy in the classification of leukemia patients. This is a promising development, as it could lead to the development of new diagnostic tools that could help to improve the early detection and treatment of leukemia.

There are a number of limitations to the study that should be considered. First, the study was conducted on a relatively small dataset of patients. This means that the results of the study may not be generalizable to a larger population of patients. Second, the study only considered a single type of leukemia. It is possible that deep learning models could be trained to achieve even higher accuracy if they were trained on a dataset that included multiple types of leukemia.

Despite these limitations, the results of the study suggest that deep learning has the potential to be a valuable tool for the diagnosis of leukemia. Further research is needed to confirm the findings of this study and to develop deep learning models that can be used in clinical practice..

Bibliography

[1] R. Sigit, M. M. Bachtiar and M. I. Fikri, "Identification Of Leukemia Diseases Based On Microscopic Human Blood Cells Using Image Processing," 2018 International Conference on Applied Engineering (ICAE), 2018, pp.15, doi: 10.1109/INCAE.2018.8579387.

[2] P. Kumar and S. M. Udwadia, "Automatic detection of acute myeloid leukemia from microscopic blood smear image," 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2017, pp. 1803-1807, doi: 10.1109/ICACCI.2017.8126106.

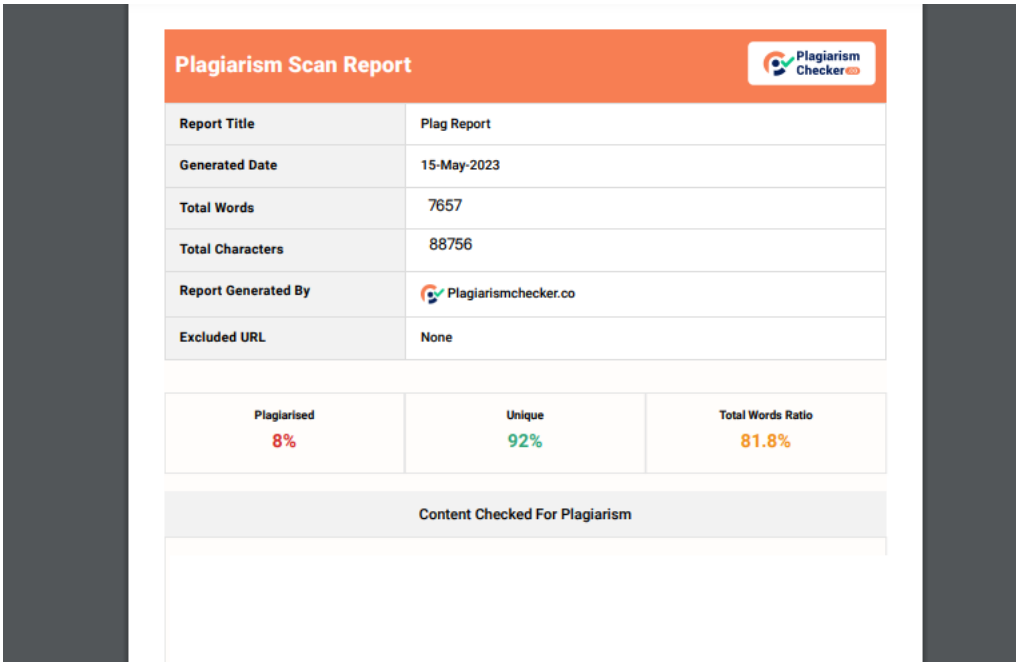
[3] T. Dharani and S. Hariprasath, "Diagnosis of Leukemia and its types Using Digital Image Processing Techniques," 2018 3rd International Conference on Communication and Electronics Systems (ICCES), 2018, pp. 275-279, doi: 10.1109/CESYS.2018.8724075.

[4] Daqqa KA, Maghari AY, Al Sarraj WF, "Prediction and diagnosis of leukemia using classification algorithms", 8th International Conference on Information Technology (ICIT), 2017 May 17 (pp. 638-643).

[5] Jha, K. K., Dutta, H. S. (2019). Nucleus and cytoplasm-based segmentation and actor-critic neural network for acute lymphocytic leukaemia detection in single cell blood smear images. Medical Biological Engineering Computing, 58(1), 171-186.

Appendices

11.2 Plagiarism Report of Published Paper(s)



Plagiarism Scan Report

Report Title	Plag Report
Generated Date	15-May-2023
Total Words	7657
Total Characters	88756
Report Generated By	Plagiarismchecker.co
Excluded URL	None

Plagiarised	Unique	Total Words Ratio
8%	92%	81.8%

Content Checked For Plagiarism

Figure 11.2.1: Plagiarism Report

11.3 Base Paper(s)

Sr. No.	**Title**	**Author**	**Date**
1.	Automatic Detection of Acute Myeloid Leukemia from Microscopic Blood Smear Image	. Kumar and S. M. Udwadia	2017
2.	Ientification Of Leukemia Dis- eases Based On Microscopic Human Blood Cells Using Image Process- ing	R. Sigit, M. M. Bachtiar and M. I. Fikri,	2018

11.4 Papers Published And Certificates

List of Publications

Sr. No.	Name of Conference or Journals	National/ International	Date	ISBN/ISSN No.
1	**International Journal for Research in Applied Science and Engineering Technology (IJRASET),**	**International Journal**	**06thNov, 2022**	**Online: 2321-9653**
2	**Advances In Mordern Technologies of Multidisciplinary Research in Engineering Field (AIMTMREF)**	**International Conference**	**25thApril, 2023**	**Online**

Table 11.4.1: List of Publication

Certificates