HEAD HEMORRHAGE USING DEEP LEARNING

A SOCIALLY RELEVANT MINI PROJECT REPORT

Submitted by

D.L.V.S SATHWIK[211423104106]

S. JOSHUA PHINEHAS [211423104268]

L.B.M VIMAL [211423104737]

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING
in

COMPUTER SCIENCE AND ENGINEERING

PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

OCTOBER 2025

PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report**“HEAD HEMORRHAGE USING DEEP LEARNING”** is the Bonafide work of **D.L.V.S.SATHWIK[211423104106],S.JOSHUAPHINEHAS [211423104268], L.B.M.VIMAL[211423104737**] carried out the project work under my supervision.

Signature of the HOD with date

Dr L.JABASHEELA M.E., Ph.D.,

Professor and Head,
Department of Computer Science and
Engineering,
Panimalar Engineering College,
Chennai- 123

Signature of the Supervisor

Dr. V.SUBEDHA M-TECH, Ph.D.,

Professor,
Department of Computer Science and
Engineering, Panimalar Engineering
College,
Chennai- 123

Submitted for 23CS1512 - Socially Relevant Mini project viva-voce Examination

Held on ....

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION BY THE STUDENT

We of **D.L.V.S.SATHWIK [211423104106] ,S. JOSHUA PHINEHAS [211423104268], L.B.M VIMAL [211423104 737]** hereby declare that this project report titled **“HEAD HEMORRHAGE USING DEEP LEARNING”,** under the Guidence of **Dr.SUBEDHA.V M.E.,Ph.D.,** is the original work done by us and we have not plagiarized or submitted to any other degree in any university by us.

SIGNATURE OF THE STUDENTS

D.L.V.S.SATHWIK [211423104106]

S. JOSHUA PHINEHAS [211423104268]

L.B.M VIMAL [211423104737]

ACKNOWLEDGEMENT

We express our deep gratitude to our respected **Secretary and Correspondent Dr.P.CHINNADURAI, M.A., Ph.D.,** for his kind words and enthusiastic motivation, which inspired us a lot in completing this project.

We would like to extend our heartfelt and sincere thanks to our Directors **Tmt.C.VIJAYARAJESWARI, Dr.C.SAKTHIKUMAR, M.E., Ph.D.,** and **Dr.SARANYASREE SAKTHIKUMAR B.E.,M.B.A.,Ph.D.,** for providing us with the necessary facilities for completion of this project.

We also express our gratitude to our Principal **Dr.K.MANI, M.E., Ph.D.,** for his timelyconcern and encouragement provided to us throughout the course.

We thank the HOD of CSE Department, **Dr.L.JABASHEELA, M.E., Ph.D.,** for the support extended throughout the project.

We would like to thank our Project Coordinator **Dr.V. SUBEDHA M.TECH, Ph.D.,** and our Project Guide **Dr.V. SUBEDHA M.TECH, Ph.D.,** and all the faculty members of the Department of CSE for their advice and suggestions for the successful completion of the project.

D.L.V.S.SATHWIK [211423104106]

S.JOSHUA PHINEHAS[211423104268

L.B.M.VIMAL[211423104737]

ABSTRACT

The classification of cerebral hemorrhages in Computed Tomography (CT) scans using Convolutional Neural Networks (CNNs) is a rapidly evolving field in medical imaging. Brain hemorrhages are a critical condition that can result in serious health consequences and death. Recently, deep neural networks have been employed for image identification and classification, producing encouraging outcomes in medical image analysis. The objective of this study is to utilize deep learning methods and CNNs to identify brain hemorrhages in CT images. The inspiration for this research stems from the challenges faced by physicians in accurately recognizing brain hemorrhages, especially in the early stages when misdiagnosis is more likely. Through a series of CT experiments, two pre trained CNNs (VGG16 and VGG19) were developed and evaluated for image categorization as either hemorrhage or non-hemorrhage. The VGG16 pre-trained model showed exceptional accuracy compared to the VGG19 model. The VGG16 model also achieved the highest accuracy of 95.00%. The findings of this research underscore the potential of integrating deep learning into medical diagnostics to reduce human error, accelerate clinical decision-making, and enhance patient outcomes. Future enhancements may include extending the system to handle three-dimensional volumetric data, real-time detection, and multi-class classification for different types of intracranial hemorrhages. This approach can contribute significantly to the development of AI-assisted healthcare systems, improving accessibility and diagnostic accuracy in medical imaging.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

1	**CN – COMPOUND TOMOGRAPHY**
2	**CNN - CONVENTIONAL NEURAL NETWORK**
3	**VGG – VISUAL GEOGRAPHY GROUP**
4	**CAD – COMPUTER AIDED DIAGNOSIS**
5	**ANN – ARTIFICIAL NEURAL NETWORK**
6	**DCNN – DEEP CONVENTIONAL NEURAL NETWORK**

CHAPTER 1
INTRODUCTION

Intracranial hemorrhage refers to bleeding within the skull. Intracranial hemorrhage is an important cause of death and disability and is a cause of stroke. Intracranial hemorrhage can occur spontaneously or in the setting of trauma. Spontaneous intracranial hemorrhage can be related to variety of disease processes including, but not limited to arteriovenous malformations, ruptured aneurysms, anticoagulation, tumors, venous sinus thrombosis, hypertension, cerebral amyloid angiopathy, and hemorrhagic conversion of strokes. Traumatic intracranial hemorrhage can occur in anyone in the setting of trauma, but patients on anticoagulation are at significantly increased risk for intracranial hemorrhage. Intracranial hemorrhage is an emergency with rapid diagnosis of paramount importance for improving patient outcomes due to fast patient decline within the first few hours after onset of symptoms. There are four major types of intracranial hemorrhage: epidural, subdural, subarachnoid, and intra parenchymal, which refer to the location of bleeding. Unenhanced computed tomography (CT) scans of the brain are commonly used to evaluate for intracranial hemorrhage. Differences in x-ray attenuation and location of intracranial hemorrhage on unenhanced CT scans of the brain make them detectable and allow the different types of intracranial hemorrhage to be differentiated. Hemorrhages can be subtle for several reasons: small size; blood pooling along the edge of normal intracranial structures with similar attenuation, in particular, the falx cerebri or hemorrhage age of days or weeks, in which case the attenuation on CT becomes more similar to that of the brain tissue. Although intracranial hemorrhages are typically hyper attenuating compared to gray matter, they are sometimes hypo attenuating. Other features in the brain, such as calcifications of the pineal glands and choroid plexus, can also have attenuation similar to intracranial hemorrhages.

.

hemorrhages, a computer-aided diagnosis (CAD) system could prioritize cases in the radiologist's queue. Flagging scans with possible significant hemorrhages could help triage,which imaging studies the radiologist should review first, and this could significantly decrease the time to detection of a potential life- threating intracranial hemorrhage. Then, once detected, intracranial hemorrhages are commonly followed with serial CT scans to evaluate the stability of the size of the hemorrhage to help make the clinical decision of whether surgery is necessary or not and automated detection and size measurement of known hemorrhages would allow for rapid comparison across multiple scans. Development of CAD systems in general has focused on other pathologies, such as cancer. No commercial CT systems offer CAD for intracranial hemorrhage. Academic research for intracranial hemorrhage has focused on standard multi-step image processing approaches. Our work is novel in that it applies deep learning to this application to perform the detection in a single classification step, optimized based on data, as well as developing postprocessing to significantly improve specificity from CNN outputs.

PROBLEM DEFINITION:

Head hemorrhage, characterized by bleeding within or around the brain, poses a critical threat to human life due to its rapid progression and the need for immediate medical intervention. Despite advances in neuroimaging and diagnostic technologies, the early detection and classification of head hemorrhages remain major challenges — particularly in emergency or low-resource settings where expert radiologists and advanced imaging tools may not be readily available. Manual interpretation of CT or MRI scans is often time-consuming, subjective, and prone to human error, leading to delayed or inaccurate diagnoses. Therefore, there is a crucial need for an automated, accurate, and efficient system capable of detecting and classifying head hemorrhages from medical images, assisting clinicians in making faster and more reliable diagnostic decisions.

CHAPTER 2

LITERATURE SURVEY

Title: *RADnet:* Radiologist-Level ICH Detection with Deep Learning

Authors: Grewal, D. et al.

This study introduced RADnet, one of the early deep learning-based systems for detecting intracranial hemorrhage from CT scans. The architecture used an ensemble of convolutional neural networks (CNNs) to classify head CT slices. The system achieved radiologist-level performance in terms of accuracy and sensitivity for hemorrhage detection. The work highlighted the importance of model ensembling and large-scale data training in medical imaging tasks.

Title: CNN-LSTM Architectures for Intracranial Hemorrhage Detection

Authors: Nguyen, D. et al.

This research proposed a hybrid deep learning approach combining Convolutional Neural Networks (CNNs) for feature extraction and Long Short-Term Memory (LSTM) networks for temporal sequence modeling of CT slices. By treating CT scans as sequences, the method captured contextual information across multiple slices, improving detection of subtle hemorrhages. The CNN-LSTM model demonstrated higher sensitivity and specificity compared to standalone CNN classifiers.

Title: *3D* Anisotropic Segmentation for Intracranial Hemorrhage*:* INSTANCE Challenge

Authors: Multiple researchers (INSTANCE Challenge, 2023)

The INSTANCE challenge benchmarked various 3D segmentation models for non-contrast CT scans in detecting and segmenting intracranial hemorrhage. Unlike 2D slice- wise classifiers, these models processed the CT volume holistically, improving performance in localization and volumetric quantification of hemorrhage. Dice coefficient and lesion-level sensitivity were primary metrics, with top-performing methods demonstrating state-of-the-art volumetric segmentation accuracy.

Title: Dual-Task Vision Transformer for ICH Classification and Localization

Authors: Zhang, Y. et al. (2024)

This work explored the use of Vision Transformers (ViTs) for dual-task learning: both classifying hemorrhage types and localizing lesions. Global self-attention in ViTs allowed better modeling of relationships across CT slices. The dual-task architecture achieved faster inference and improved accuracy over CNN-only models, while providing explainable heatmaps to assist clinicians.

Title: Explainable AI in Head CT Analysis with Grad-CAM

Authors: Kumar, A. et al. (2023)

This paper investigated the integration of explainability techniques (Grad-CAM) with CNN-based haemorrhage detection. The model not only classified hemorrhage types but also generated visual heatmaps highlighting suspicious regions in CT scans. The explainability approach enhanced clinician trust and usability by making AI decisions transparent, while maintaining strong on RSA and institutional datasets.

2.1 ORGANIZATION PROFILE

RAJESWARI TECHNO SOLUTIONS located at NLR, has a rich background in developing academic student projects, especially in solving latest IEEE Papers, Software Development and continues its entire attention on achieving transcending excellence in the Development and Maintenance of Software Projects and Products in Many Areas. In Today's Modern Technological Competitive Environment, Students in Computer Science Stream Want to Ensure That They Are Getting Guidance In An Organization That Can Meet Their Professional Needs. With Our Well-Equipped Team of Solid Information Systems Professionals, Who Study, Design, Develop, Enhance, Customize, Implement, Maintain and Support Various Aspects of Information Technology, Students Can Be Sure.

We Understand the Students' Needs, And Develop Their Quality Of Professional Life By Simply Making The Technology Readily Usable For Them. We Practice Exclusively in Software Development, Network Simulation, Search Engine Optimization, Customization And System Integration. Our Project Methodology Includes Techniques For Initiating A Project, Developing The Requirements, Making Clear Assignments To The Project Team, Developing A Dynamic Schedule, Reporting Status To Executives And Problem Solving.

The indispensable factors, which give the competitive advantages over others in the market, may be slated as:

- Performance
- Pioneering efforts
- Client satisfaction
- Innovative concepts
- Constant Evaluations

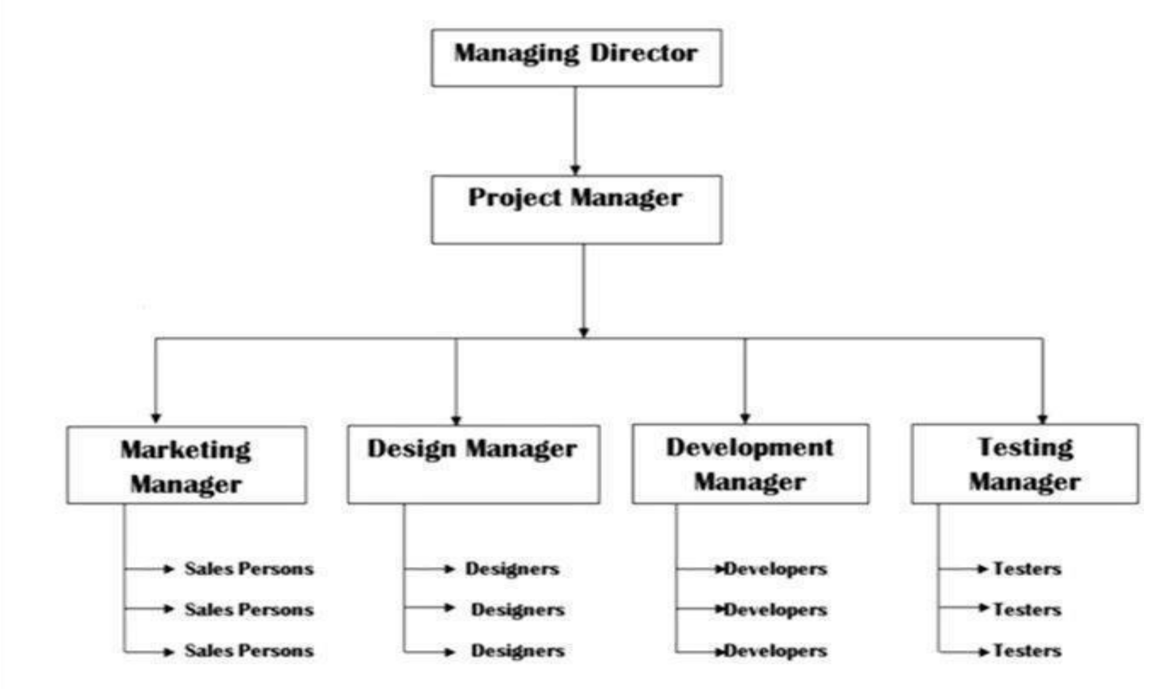
OUR VISION:

“WE CAN DEVELOP YOUR OWN IDEAS” this is our vision; we work according to our vision.

CUSTOMER FOCUS :

We view our customer relationships as partnerships and are committed to being accountable to ensure that the development, integration and implementation of solutions are performed in a professional and timely manner. We unite this accountability with our dedication to applying the most appropriate methodologies & technologies, and we dispatch our 'best-of- breed' technology professionals to make it all happen. The result is satisfied customers who consistently give us high marks for our expansive offerings of precise engineering.

2.2 ORGRANZATION CHART:



CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

Most existing methods in the literature uses the extraction of explicit features, including statistical based, geometrical based, wavelet-based, block based, key point based, transformations based, texture based and so on. Most of the methods require hand-crafted or feature engineering. Most of the features have good results but not invariant to different types of geometrical operations and less robust to various types of image forgery.

DISADVANTAGES:

1) Need more computational power to detect disease images.

2) Model not supported for checking more and more images.

3.2 PROPOSED SYSTEM

Performance of the deep features extracted from a pre-trained Alex Net based model is quite satisfactory, even in the presence of rotational and geometrical transformation and also compared the results of the given approach with the existing state-of-the-art approaches.

ADVANTAGES

3) Less computational power required to detect head image.

4) Model trained based on CNN and Deep learning so there is no ambiguity to detect disease images.

IMPLEMENTATION:

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus, it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

RULE-BASED FEATURES

Human experts with years of experience created many rules to detect whether a user submitted YouTube comment is spam or not. An example of such rules is “blacklist”, i.e. whether the user has been detected or complained as spam before. Each rule can be regarded as a binary feature that indicates the fraud likeliness.

SELECTIVE LABELING:

If the spam score is above a certain threshold, the case will enter a queue for further investigation by human experts. Once it is reviewed, the final result will be labeled as Boolean, i.e. spam or clean. Cases with higher scores have higher priorities in the queue to be reviewed. The cases whose spam score are below the threshold are determined as clean by the system without any human judgment.

1) SPAM CHURN:

Once one case is labeled as spam by human experts, it is very likely that the reviewer is not trustable and may be also commenting other spam; hence all the spam submitted by the same reviewer are labeled as spam too. The fraudulent spammer along with his/her cases will be removed from the website immediately once detected.

a. FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

i .ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available

ii . TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resource

3.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

FUNCTIONAL REQUIREMENTS

- **ADMIN**

It is used by the admin for activating and deactivating user request.

- **USER**

It is used to add document into database or used to download document from database.

- **MODULAR MANAGER**

It is used to accept the document that is to be upload

- **NON-FUNCTIONAL REQUIREMENTS:**

Non-Functional Requirements (quality attributes) ensure the delivery of an operable and manageable system which provides the required functionality reliable, uninterrupted or with minimal time of interruption even under unusual situation

- **SECURITY**

  Login requirements - access levels, CRUD levels

  Password requirements - length, special characters, expiry, recycling policies Inactivity timeouts – durations, actions

- **AUDIT**

  Audited elements – what business elements will be audited? fields – which data fields will be audited?

  Audit file characteristics - before image, after image, user and time stamp, etc

- **PERFORMANCE**

  Response times - application loading, screen open and refresh times, etc Processing times – functions, calculations, imports, exports Queryand Reporting times – initial loads and subsequent loads Capacity.

- **AVAILABILITY**

  Hours of operation – when is it available? Consider weekends, holidays, maintenance times, etc Locations of operation – where should it be available from, what are the connection requirements?

- **RELIABILITY**

  Mean Time Between Failures – What is the acceptable threshold for down-time? e .g. One a year, 4,000 hours

  Mean Time To Recovery – if broken, how much time is available to get the system back up again?

- **INTEGRITY**

  Fault trapping (I/O) – how to handle electronic interface failures, etc

  Bad data trapping - data imports, flag-and-continue or stop the import policies, etc

Data integrity –referential integrity in database tables and interfaces.Image compression and decompression standards.

- **RECOVERY**

  Recovery process – how do recoveries work, what is the process? Recovery time scales – how quickly should a recovery take to perform?
  Backup frequencies – how often is the transaction data, set-up data, and system (code) backed-up?

- **COMPATIBILITY**

  Compatibility on different operating systems – What does it have to be able to run on? Compatibility on different platforms – What are the hardware platforms it needs to work on?

- ***MAINTAINABILITY***

  Conformance to architecture standards – What are the standards it needs to conform to or have exclusions from?
  Conformance to design standards – What design standards must be adhered to or exclusions created?
  Conformance to coding standards – What coding standards must be adhered to or exclusions created?

- ***USABILITY***

  Look and feel standards - screen element density, layout and flow, colours, UI metaphors, keyboard shortcuts
  Internationalization / localization requirements – languages, spellings, keyboards, paper sizes, etc

CHAPTER 4
SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

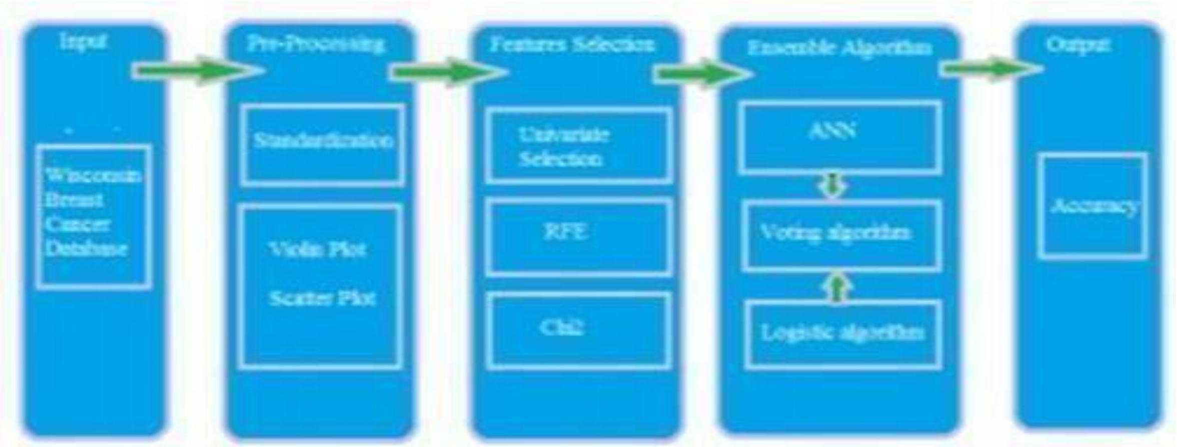


Fig : SYSTEM ARCHITECTURE

MODULES:

DATA COLLECTION:

Data collection is a very basic module and the initial step towards the project. It generally deals with the collection of the right dataset. The dataset that is to be used in the market prediction hasto be used to be filtered based on various aspects. Data collection also complements to enhance the dataset by adding more data that are external. Our data mainly consists of the previous year stock prices. Initially, we will be analyzing the Kaggle dataset and according to the accuracy, wewill be using the model with the data to analyze the predictions.

PRE-PROCESSING DATA

pre-processing is a part of data mining, which involves transforming raw data into a more coherent format. Raw data is usually, inconsistent or incomplete and usually contains many errors. The data pre-processing involves checking out for missing values, looking scaling to limit the range of variables so that they can be compared on common environs.

TRAINING THE MACHINE

The machine is similar to feeding the data to the algorithm to touch up the test data. The training sets are used to tune and fit the models. The test sets are untouched, as a model should not be judged based on unseen data. The training of the model includes cross- validation where we get awell-grounded approximate performance of the model using the training data. Tuning models aremeant to specifically tune the hyperparameters like the number of trees in a random forest. We perform the entire cross-validation loop on each set of hyperparameter values. Finally, we will calculate a cross-validated score, for individual sets of hyperparameters. Then, we select the besthyperparameters. The idea behind the training of the model is that we some initial values with the dataset and then optimize the parameters which we want to in the model. This is kept on repetition until we get the optimal values. Thus, we take the predictions from the trained model on the inputs from the test dataset. Hence, it is divided in the ratio of 80:20 where 80% is for thetraining set and the rest 20% for a testing set of the data.

DATA SCORING

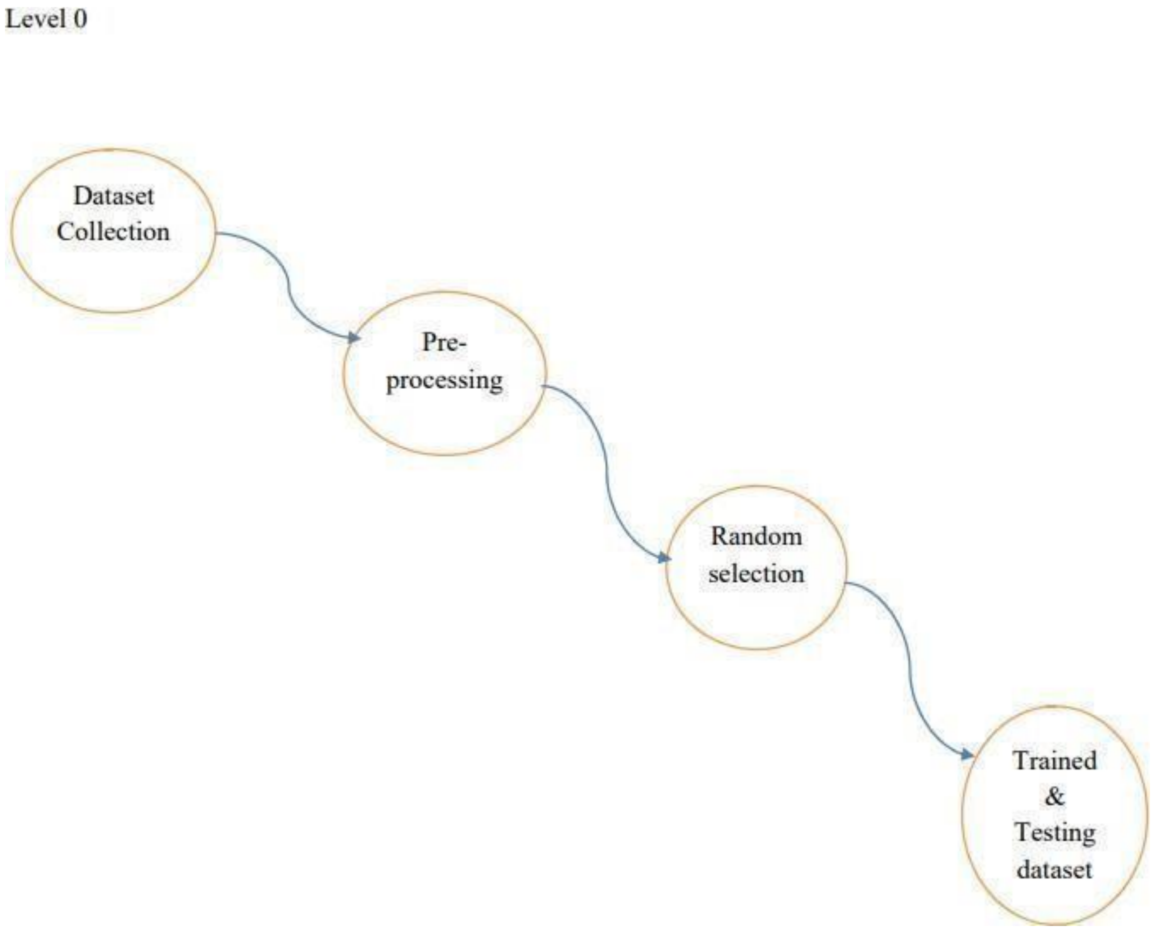
The process of applying a predictive model to a set of data is referred to as scoring the data. The technique used to process the dataset is the Random Forest Algorithm.

Random forest involves an ensemble method, which is usually used, for classification and as well as regression. Based onthe learning models, we achieve interesting results. The last module thus describes how the resultof the model can help to predict the probability of a stock to rise and sink based

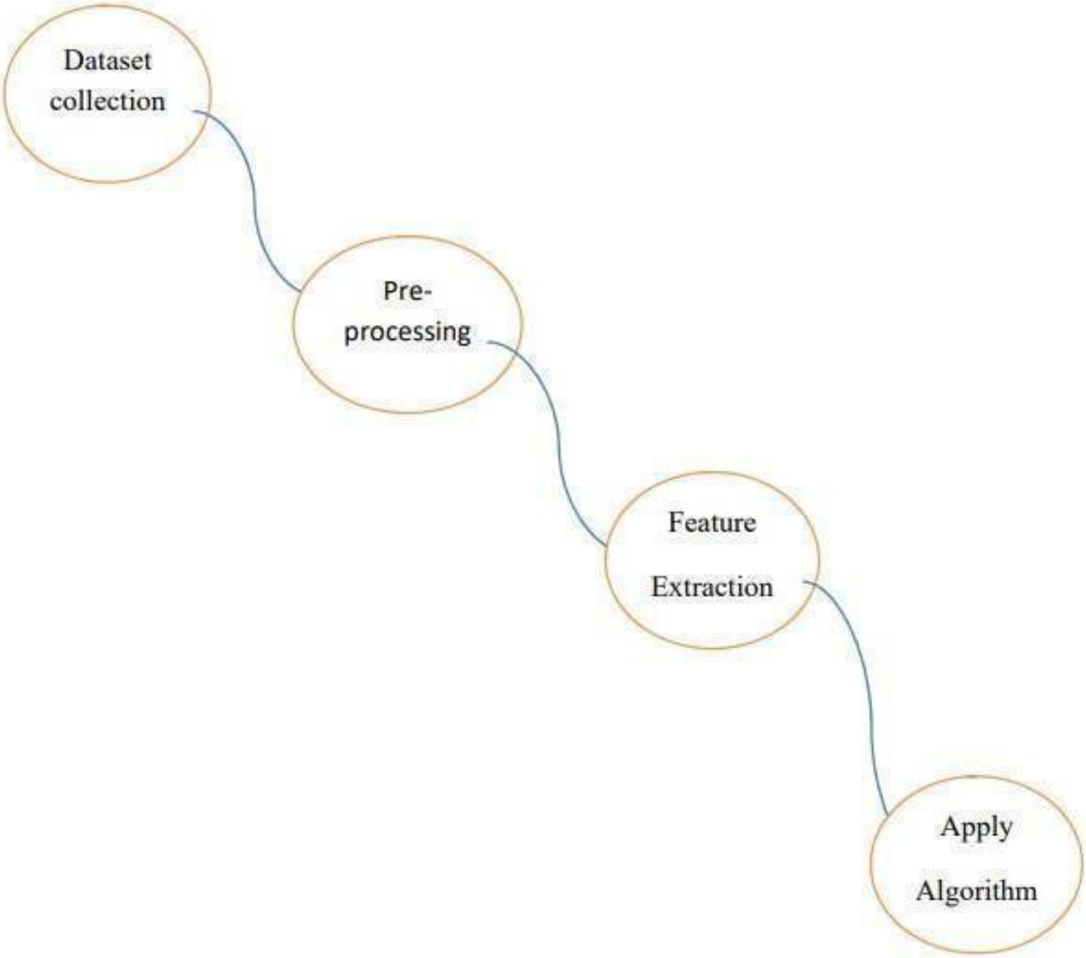
4.2DATA FLOW DIAGRAM:

The data flow diagram (DFD) is one of the most important tools used by system analysis. Data flow diagrams are made up of number of symbols, which represents system components. Most data flow modeling methods use four kinds of symbols: Processes, Data stores, Data flows and external entities. These symbols are used to represent four kinds of system components. Circles in DFD represent processes.

Data Flow represented by a thin line in the DFD and each data storehas a unique name and square or rectangle represents external entities name and square or rectangle represents external entities.



LEVEL 1:


Dataset collection
Pre-processing
Feature Extraction
Apply Algorithm

UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object- oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non- software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

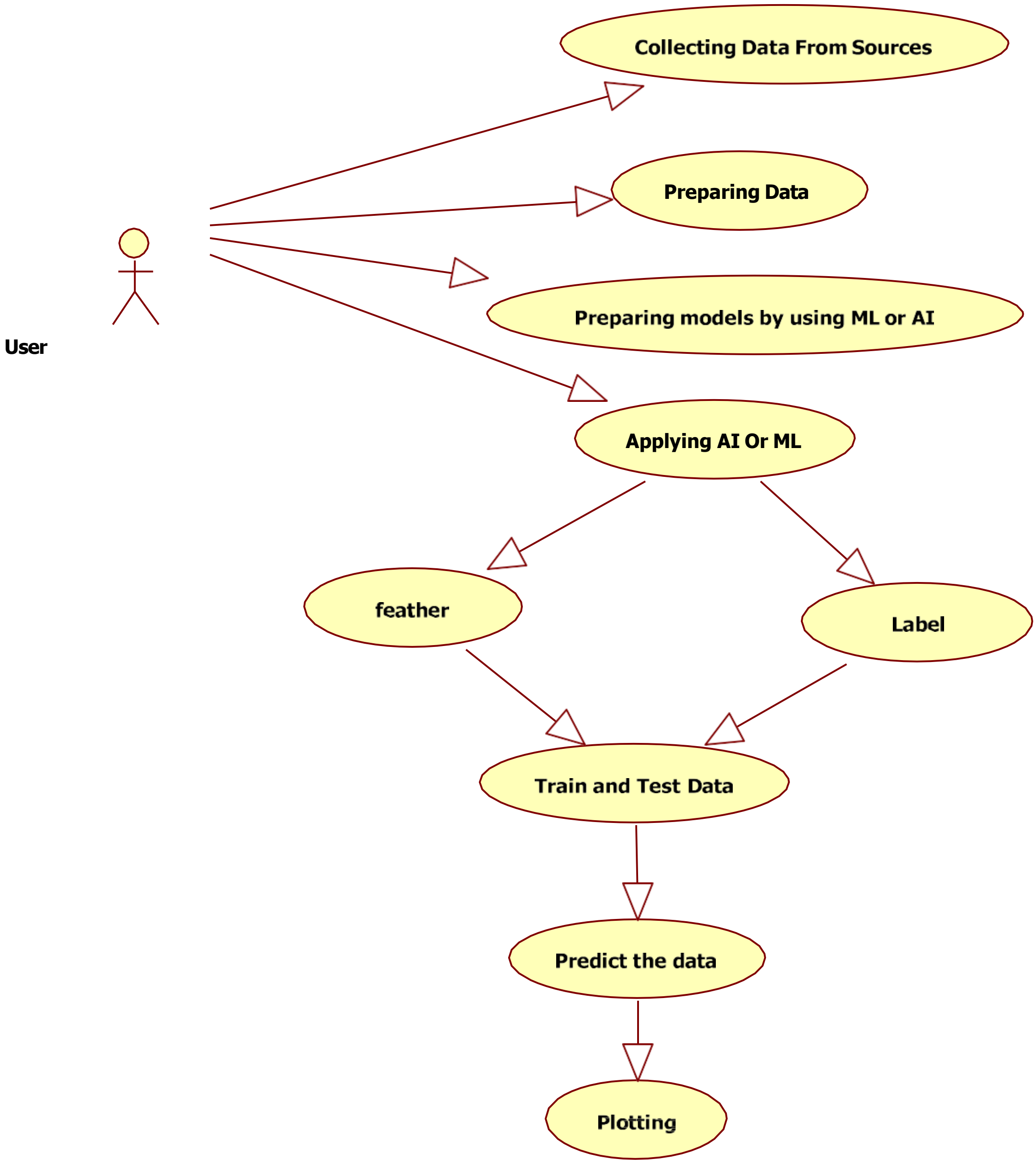
GOALS:

The Primary goals in the design of the UML are as follows:

- Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
- Provide extendibility and specialization mechanisms to extend the core concepts.
- Be independent of particular programming languages and development process.
- Provide a formal basis for understanding the modeling language.

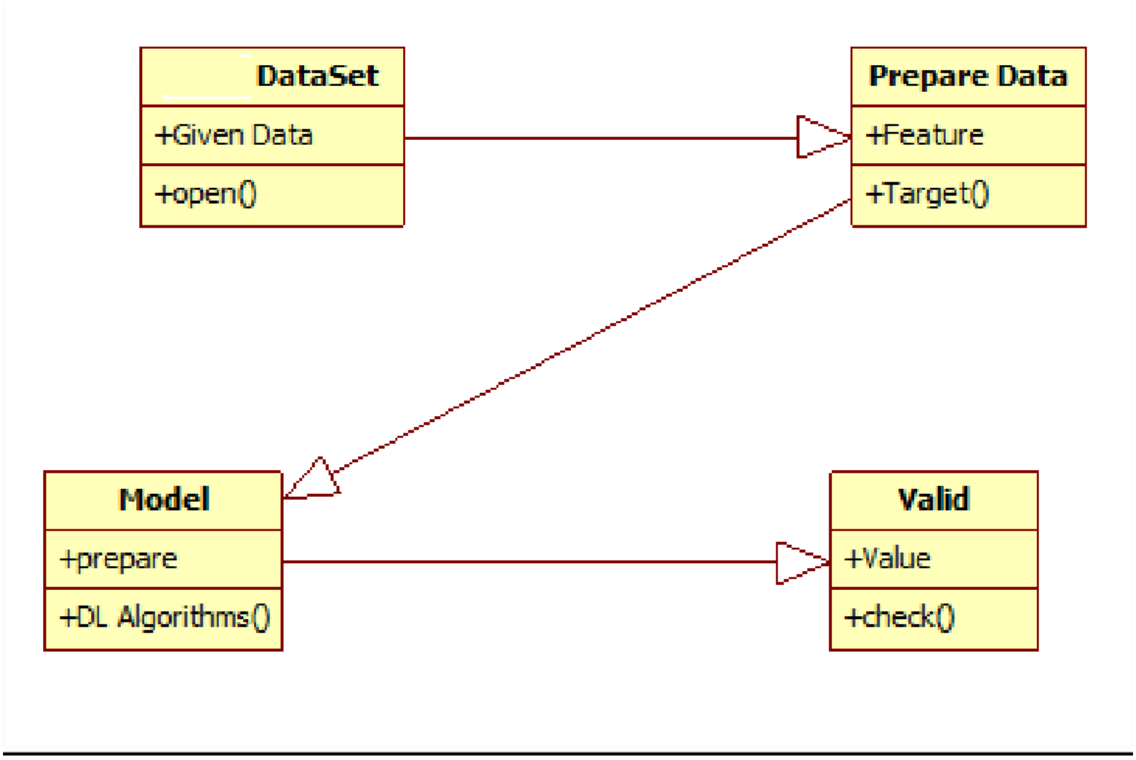
4.3 USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor.



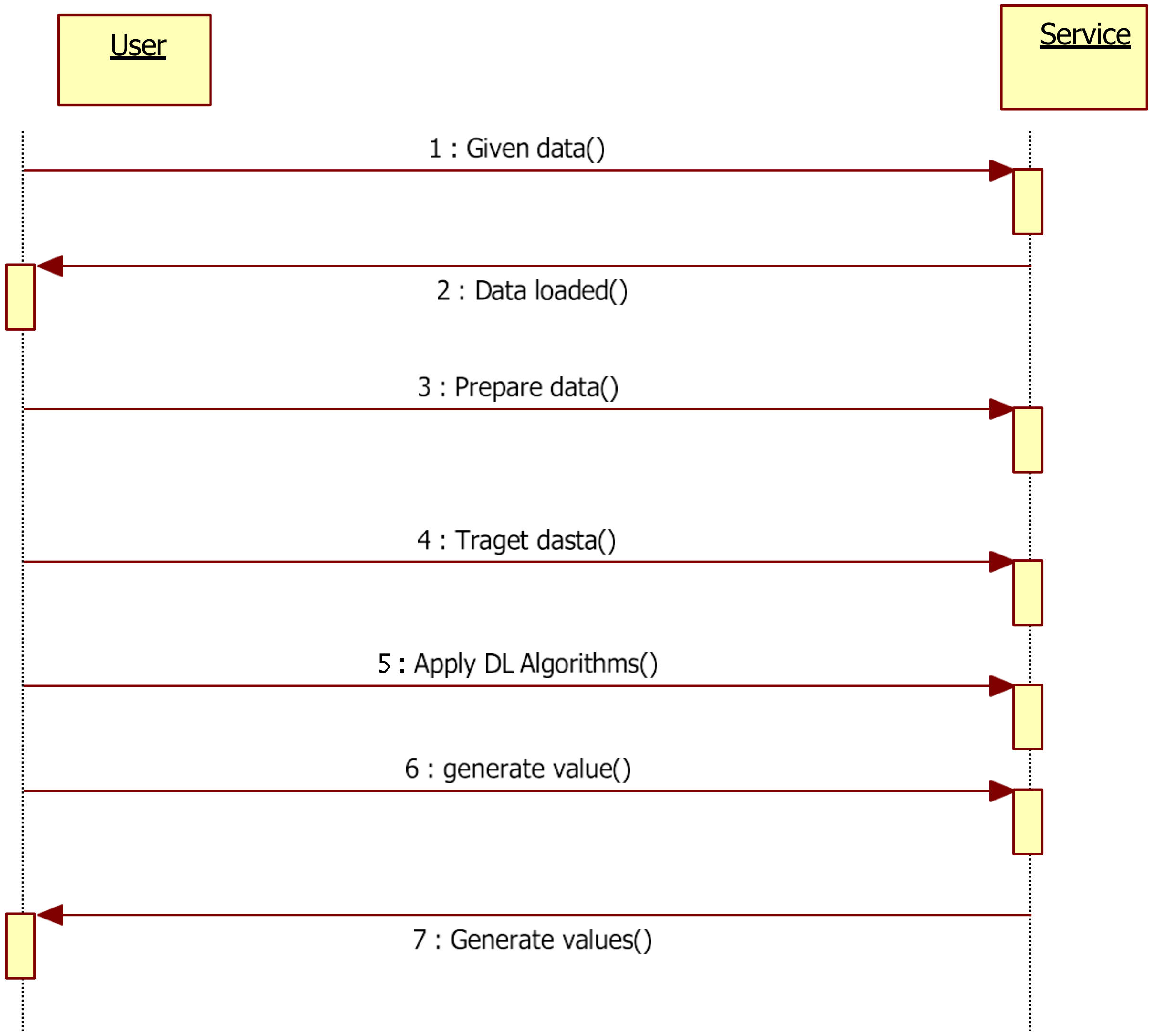
4.4 CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



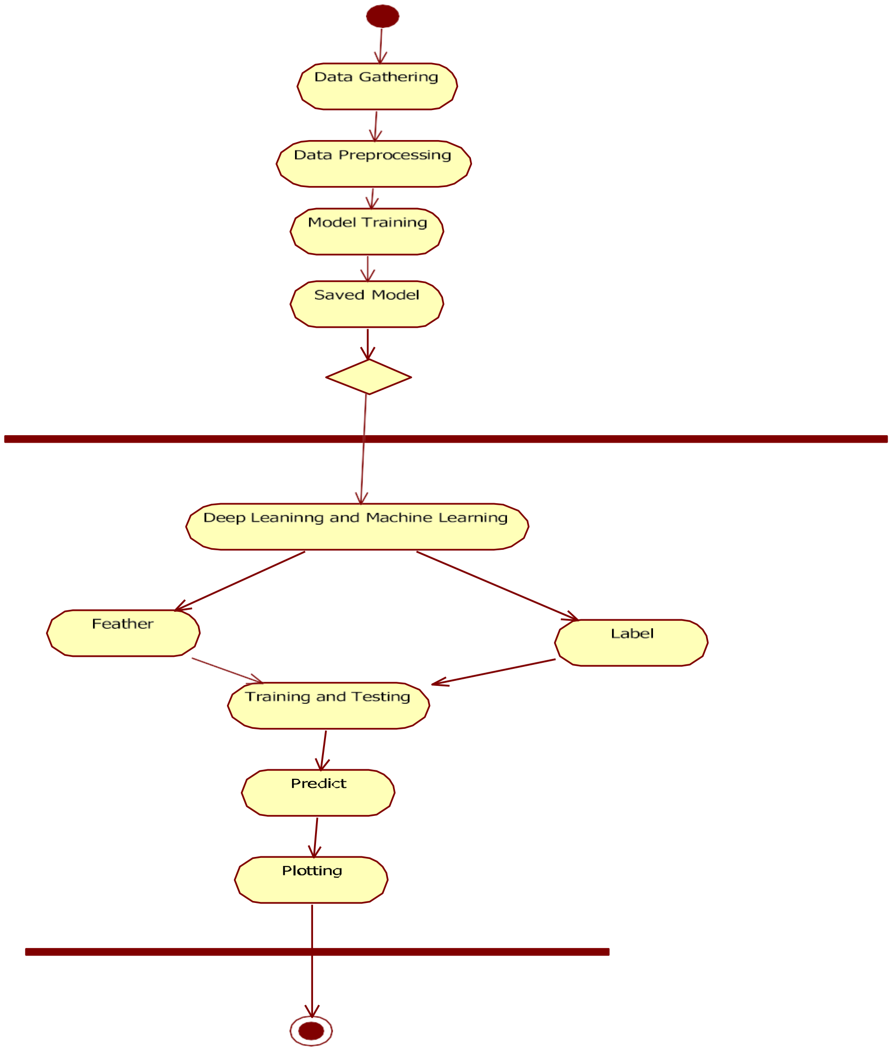
4.5 SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

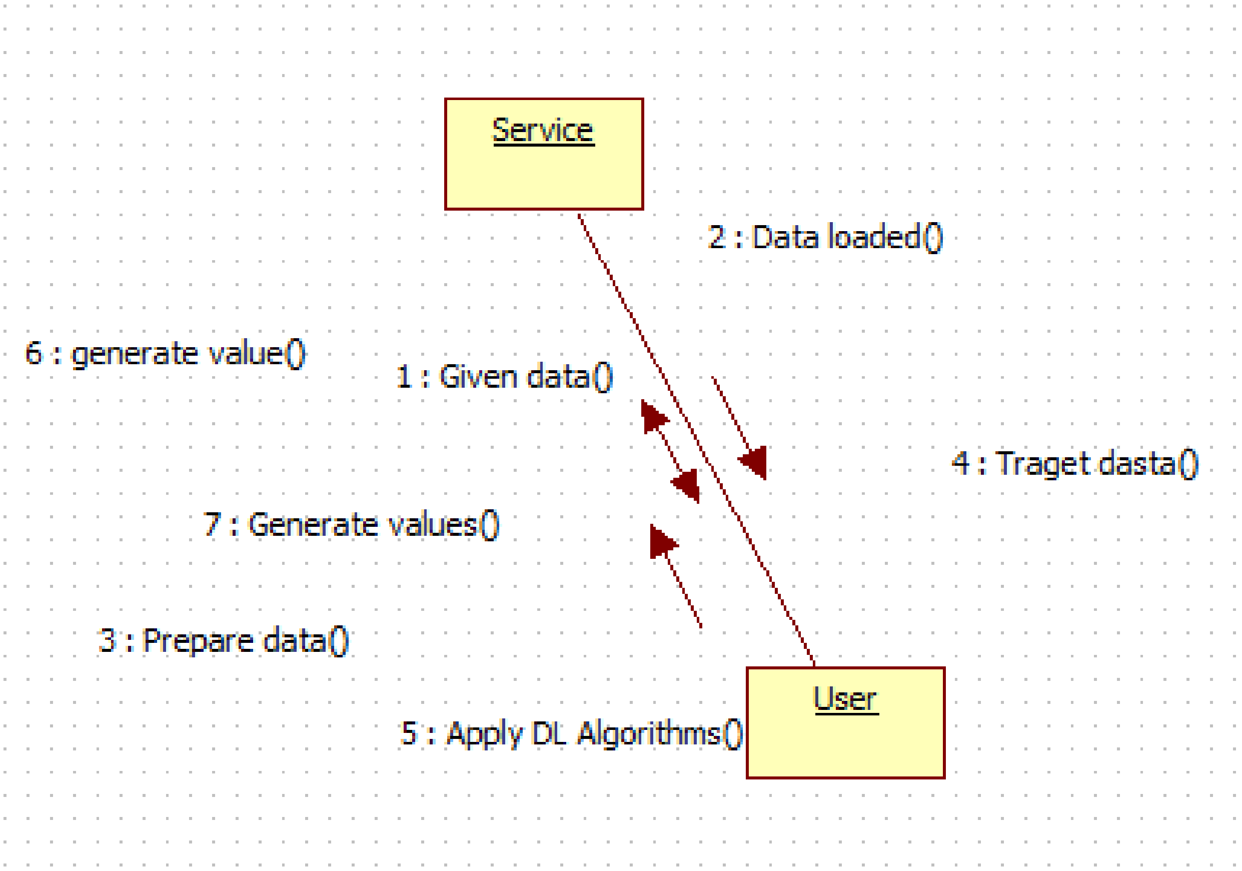


4.6 ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by- step workflows of components in a system. An activity diagram shows the overall flow of control.



4.7 COLLABORATION DIAGRAM:



4.1.1 INPUT AND OUTPUT DESIGN:

INPUT DESIGN :

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

What data should be given as input?

How the data should be arranged or coded?

The dialog to guide the operating personnel in providing input

Methods for preparing input validations and steps to follow when error occur.

OBJECTIVES:

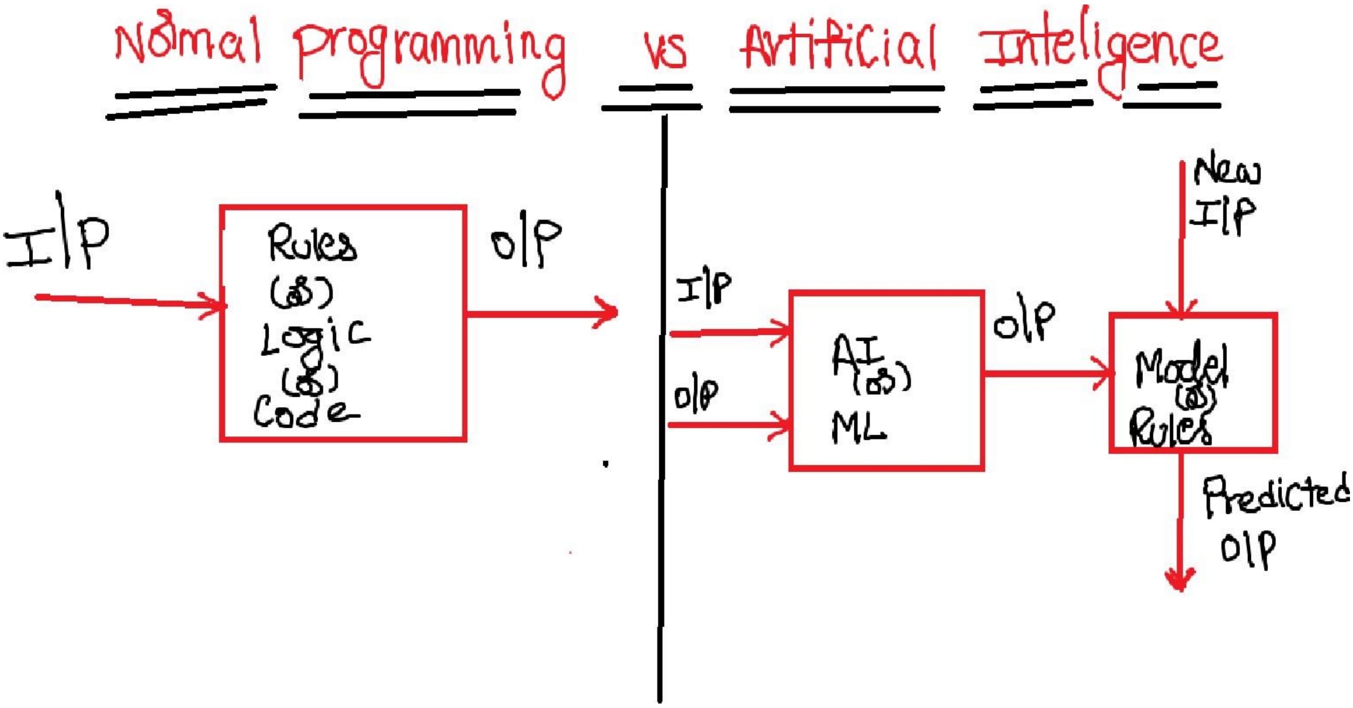
- Input Design is the process of converting a user-oriented description of the input into a computer- based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.
- It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.
- When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus, the objective of input design is to create an input layout that is easy to follow.

OUTPUT DESIGN:

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

- Convey information about past activities, current status or projections of the Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.

- **COMPARISON OF TRADITIONAL PROGRAMMING AND AI/ML PARADIGMS**


Nomal Programming vs Artificial Inteligence
I/P
Rules (or) Logic (or) Code
O/P
I/P
O/P
AI (or) ML
O/P
New I/P
Model (or) Rules
Predicted O/P

CHAPTER 5
SYSTEM IMPLEMENTATION

5.1 IMPLEMENTATION:

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus, it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

5.1.1 MODULES:

1. UPLOAD PRODUCTS

Uploading the products is done by admin. Authorized person is uploading the new arrivals to system that are listed to users. Product can be uploaded with its attributes such as brand, color, and all other details of warranty. The uploaded products are able to block or unblock by users.

2. PRODUCT REVIEW BASED ORDER

The suggestion to user's view of products is listed based on the review by user and rating to particular item. Naïve bayes algorithm is used in this project to develop the whether the sentiment of given review is positive or negative. Based on the output of algorithm suggestion to users is given. The algorithm is applied and lists the products in user side based on the positive and negative.

3. RATINGS AND REVIEWS

Ratings and reviews are main concept of the project in order to find effective product marketing. The main aim of the project is to get the user reviews based on how they purchased or whether they purchased or not. The major find out of the project is when they give the ratings and how effective it is. And this will helpful for the users who are willing to buy the same kind of product.

4. DATA ANALYSIS

The main part of the project is to analysis the ratings and reviews that are given by the user. The products can be analysis based on the numbers which are given by user. The user data analysis of the data can be done by charts format. The graphs may vary like pie chart, bar chart or etc

FUTURE WORK:

In our current work, the review content is not considered. In the future, we will explore effective ways in incorporating review content into our early reviewer prediction model. Also, we have not studied the communication channel and social network structure in diffusion of innovations partly due to the difficulty in obtaining the relevant information from our review data. We will look into other sources of data such as Flixster in which social networks can be extracted and carry out more insightful analysis. Currently, we focus on the analysis and prediction of early reviewers, while there remains an important issue to address, i.e., how to improve product marketing with the identified early reviewers. We will investigate this task with real e-commerce cases in collaboration with e-commerce companies in the future.

CHAPTER 6

SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

TYPES OF TESTS:

6.1 UNIT TESTING:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

TEST STRATEGY AND APPROACH:

Field testing will be performed manually and functional tests will be written in detail.

TEST OBJECTIVES:

All field entries must work properly.

Pages must be activated from the identified link.

The entry screen, messages and responses must not be delayed.

FEATURES TO BE TESTED

Verify that the entries are of the correct format No duplicate entries should be allowed

All links should take the user to the correct page.

6.2 INTEGRATION TESTING

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects

6.3 ACCEPTANCE TESTING

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements. **Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

SYSTEM TESTING:

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

WHITE BOX TESTING:

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

BLACK BOX TESTING:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

TEST STRATEGY AND APPROACH:

Field testing will be performed manually and functional tests will be written in detail.

TEST OBJECTIVES:

All field entries must work properly.

Pages must be activated from the identified link.

The entry screen, messages and responses must not be delayed.

FEATURES TO BE TESTED:

Verify that the entries are of the correct format No duplicate entries should be allowed All links should take the user to the correct page. Integration Testing Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.The task of the integration test is to check that components or software applications, e.g. Components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

ACCEPTANCE TESTING:

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

SYSTEM TESTING:

TESTING METHODOLOGIES:

- Unit Testing.
- Integration Testing.
- User Acceptance Testing.
- Output Testing.

6.1.1 UNIT TESTING

Unit testing focuses verification effort on the smallest unit of Software design that is the module. Unit testing exercises specific paths in a module's control structure to ensure complete coverage and maximum error detection. This test focuses on each module individually, ensuring that it functions properly as a unit. Hence, the naming is Unit Testing.

During this testing, each module is tested individually and the module interfaces are verified for the consistency with design specification. All-important processing path are tested for the expected results. All error handling paths are also tested.

6.2.1 INTEGRATION TESTING:

Integration testing addresses the issues associated with the dual problems of verification and program construction. After the software has been integrated a set of high order tests are conducted. The main objective in this testing process is to take unit tested modules and builds a program structure that has been dictated by design.

6.2.2 THE FOLLOWING ARE THE TYPES OF INTEGRATION TESTING:

I. TOP-DOWN INTEGRATION:

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module. The module subordinates to the main program module are incorporated into the structure in either a depth first or breadth first manner.

In this method, the software is tested from main module and individual stubs are replaced when the test proceeds downwards.

II. BOTTOM-UP INTEGRATION:

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom-up integration strategy may be implemented with the following steps:

- The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.
- A driver (i.e.) The control program for testing is written to coordinate test case input and output. The cluster is tested.
- Drivers are removed and clusters are combined moving upward in the program structure.

The bottom-up approaches test each module individually and then each module is module

OTHER TESTING METHODOLOGIES:

6.3.1 USER ACCEPTANCE TESTING:

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

OUTPUT TESTING:

performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways – one is on screen and another in printed format.

VALIDATION CHECKING:

Validation checks are performed on the following fields.

TEXT FIELD:

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables. Incorrect entry always flashes and error message.

NUMERIC FIELD:

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error message. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested.

A successful test is one that gives out the defects for the inappropriate data and produces and output revealing the errors in the system.

Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

USING LIVE TEST DATA:

Live test data are those that are actually extracted from organization files. After a system is partially constructed, programmers or analysts often ask users to key in a set of data from their normal activities. Then, the systems person uses this data as a way to partially test the system. In other instances, programmers or analysts extract a set of live data from the files and have them entered themselves.

It is difficult to obtain live data in sufficient amounts to conduct extensive testing. And, although it is realistic data that will show how the system will perform for the typical processing requirement, assuming that the live data entered are in fact typical, such data generally will not test all combinations or formats that can enter the system. This bias toward typical values then does not provide a true system test and in fact ignores the cases most likely to cause system failure.

USING ARTIFICIAL TEST DATA:

Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information systems department, make possible the testing of all login and control paths through the program. The most effective test programs use artificial test data generated by persons other than those who wrote the programs. Often, an independent team of testers formulates a testing plan, using the systems specifications.

The package “Virtual Private Network” has satisfied all the requirements specified as per software requirement specification and was accepted.

USER TRAINING:

Whenever a new system is developed, user training is required to educate them about the working of the system so that it can be put to efficient use by those for whom the system has been primarily designed. For this purpose, the normal working of the project was demonstrated to the prospective users. Its working is easily understandable and since the expected users are people who have good knowledge of computers, the use of this system is very easy.

MAINTAINENCE

This covers a wide range of activities including correcting code and design errors. To reduce the need for maintenance in the long run, we have more accurately defined the user's requirements during the process of system development. Depending on the requirements, this system has been developed to satisfy the needs to the largest possible extent. With development in technology, it may be possible to add many more features based on the requirements in future. The coding and designing are simple and easy to understand which will make maintenance easier.

TESTING STRATEGY:

A strategy for system testing integrates system test cases and design techniques into a well- planned series of steps that results in the successful construction of software. The testing strategy must co- operate test planning, test case design, test execution, and the resultant data collection and evaluation. A strategy for software testing must accommodate low- level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements.

CHAPTER 7
APPENDICES

A1– SDG GOALS

PRIMARY GOAL NO: SDG 3 - GOOD HEALTH AND WELL-BEING

- Target 1 : Reduce the global maternal mortality ratio.
- Target 2: Reduce premature mortality from non-communicable diseases.
- Target 3: Reduce premature mortality from non-communicable diseases.

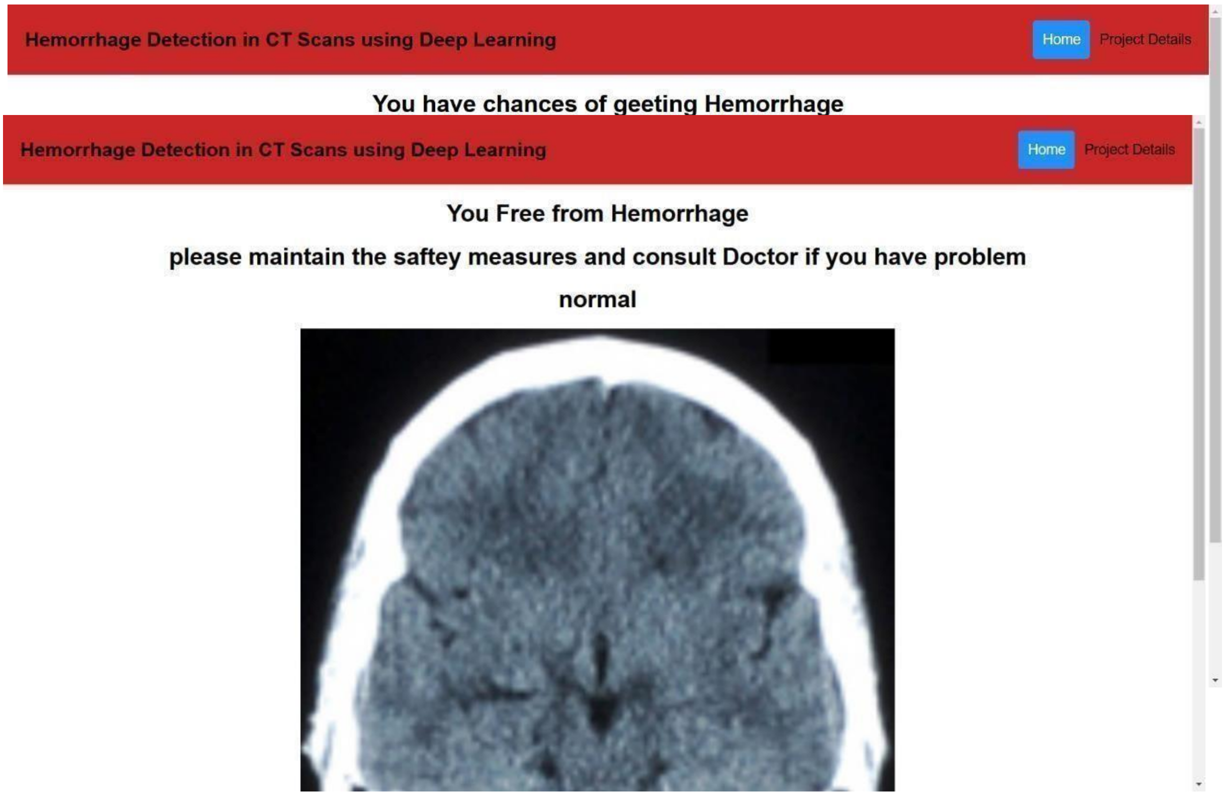
SECONDARY GOAL NO: SDG 11 - SUSTAINABLE CITIES AND COMMUNITIES

- Target 1:Provide access to safe, affordable, accessible and sustainable transport systems.
- Target 2:Reduce the number of deaths and people affected by disasters.
- Target 3: Urban design with injury prevention.

TERTIARY GOAL NO:SDG 4-QUALITY EDUCATION

- Target 1:Free, equitable primary and secondary education.
- Target 2:Education for sustainable development and global citizenship.
- Target 3:Build safe and inclusive learning environments

1.2 SCREENSHOT



Hemorrhage Detection in CT Scans using Deep Learning

You have chances of geeting Hemorrhage

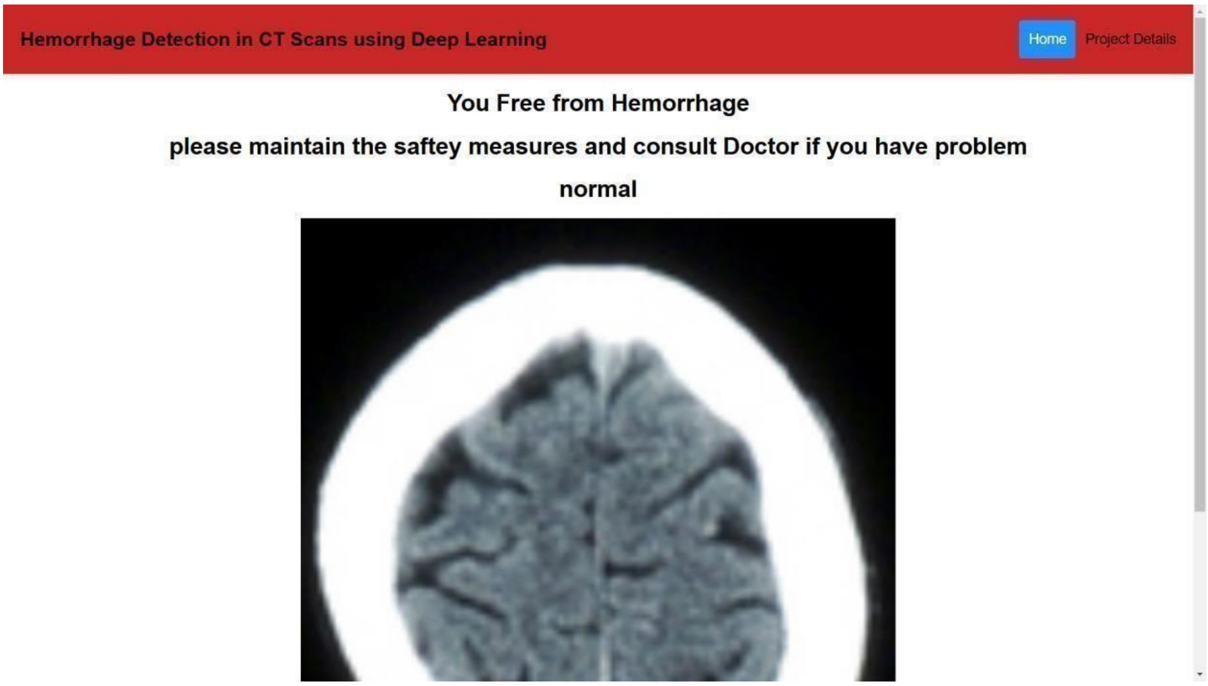
Hemorrhage Detection in CT Scans using Deep Learning

You Free from Hemorrhage

please maintain the saftey measures and consult Doctor if you have problem

normal

A2.1 Image upload page

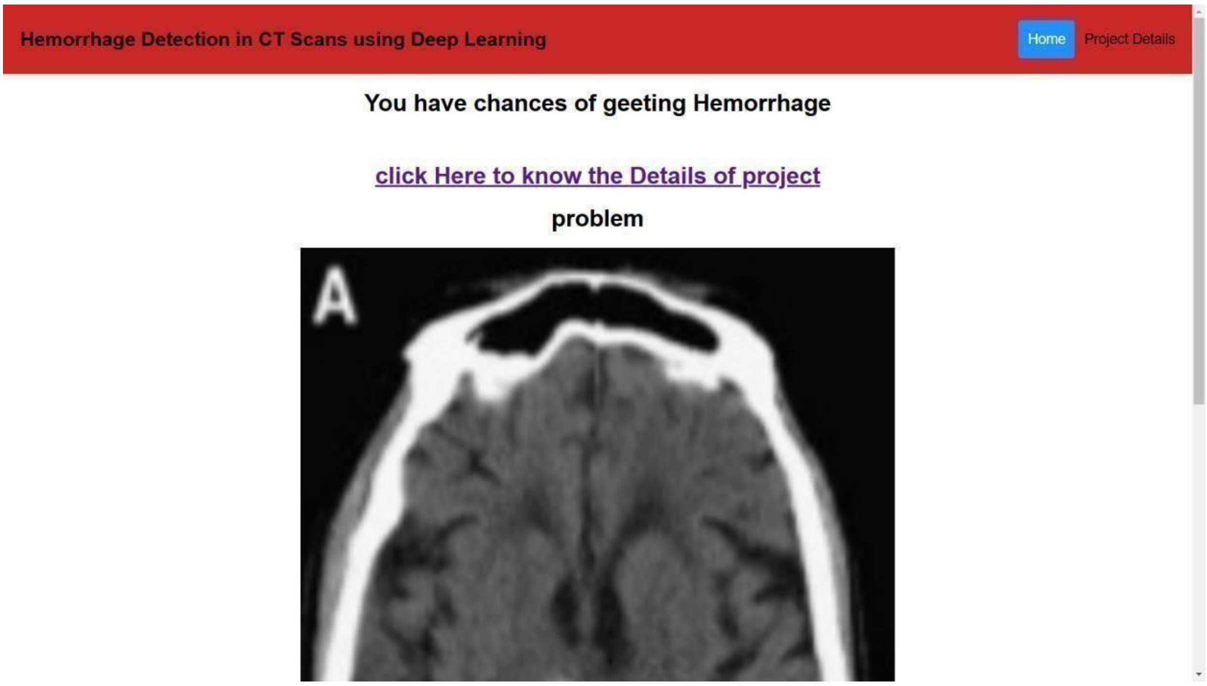


You Free from Hemorrhage

please maintain the saftey measures and consult Doctor if you have problem

normal

A2.2 Image detection page

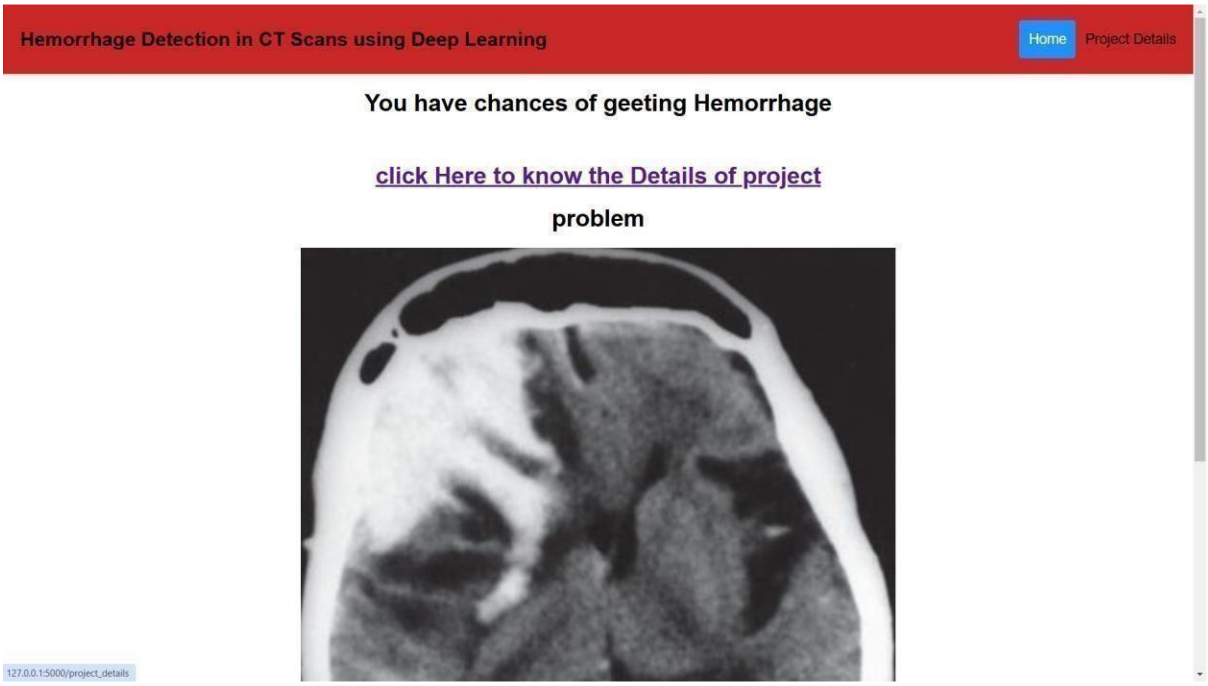


You have chances of geeting Hemorrhage

click Here to know the Details of project

problem

A2.3 Performance Analysis Graph


Hemorrhage Detection in CT Scans using Deep Learning
Home
Project Details
127.0.0.1:5000/project_details

You have chances of geeting Hemorrhage

click Here to know the Details of project

problem

A2.4 Accuracy and Loss Visualization

What is SDLC?

SDLC stands for Software Development Life Cycle. A Software Development Life Cycle is essentially a series of steps, or phases, that provide a model for the development and lifecycle management of an application or piece of software.

SDLC is the process consisting of a series of planned activities to develop or alter the software products.

BENEFITS OF THE SDLC PROCESS

The intent of a SDLC process it to help produce a product that is cost-efficient, effective, and of high quality. Once an application is created, the SDLC maps the proper deployment and decommissioning of the software once it becomes a legacy. The SDLC methodology usually contains the following stages: Analysis (requirements and design), construction, testing, release, and maintenance (response). Veracode makes it possible to integrate automated security testing into the SDLC process through use of its cloud based platform.

1. **REQUIREMENTS GATHERING:**

In this phase we gather all the requirements from the client, i.e. what are the client expected input, output……

2. **ANALYSIS**:

In this phase based upon the client requirements we prepare one documentation is called “High Level Design Document”. It contains Abstract, Functional Requirements, Non Functional Requirements, Existing System, Proposed System, SRS,………

3. **DESIGN:**

It is difficult to understand the High Level Design Document for all the members, so to understand easily we use “Low Level Design Document”. To design this document we use UML (Unified Modeling Language). In this we have Use case, Sequence,
Collaboration……..

4. **CODING**:

In this phase we develop the coding module by module. After developing all the modules we integrate them.

5. **TESTING**:

After developing we have to check weather client requirements are satisfied or not. If not we are again going to develop.

6. **IMPLEMENTATION**:

In testing phase if client requirements are satisfied, we go for implementation. i.e. we need to deploy the application in some server.

7. **MAINTENANCE**:

After deployment, if at all any problems come from the client side; we are providing maintenance for that application.

CHAPTER 8
CONCLUSION

Although the number of hemorrhages in the test set, especially when broken down by type, is small, the results provide important insights. The intraparenchymal hemorrhages were detected with the highest probability.

These were typically hyperattenuating and surrounded by normal tissue. Similarly, the epidural hemorrhage was straightforward to detect. The subdural hemorrhage that was missed was primarily hypoattenuating with respect to normal tissue.

Hypoattenuating examples were not well represented in the training set. The four intraventricular hemorrhages that were detected were larger and hyperattenuating, whereas the two that were missed were small, at the posterior of the occipital horn. The subarachnoid hemorrhages were relatively difficult to detect. These are typically narrow, with blood filling the sulci (grooves or fissures in the cortex) and sometimes isoattenuation.

Future work will improve accuracy through expanding the size of the database, and will also consider including 1.25 mm slices (in addition to the 5 mm) and classifying in 3D or in additional views (sagittal and coronal planes in addition to axial). The additional annotations will be driven by active learning based on CNN performance and the particular data examples that are most needed, particularly for hypoattenuating and subarachnoid hemorrhages.

It was observed that the CNN detections sometimes were improved over the radiologist annotations (partly based on limitations of the annotation tool as well as the radiologist’s time), so these improved detections will be incorporated back into the training set as part of the active learning approach.

CHAPTER 9
FUTURE ENHANCEMENT

The project titled "Head Hemorrhage Detection Using Deep Learning" aims to develop an automated, intelligent system capable of identifying intracranial hemorrhages from CT scan images using deep learning techniques. The primary motivation is to provide a supportive diagnostic tool for radiologists, especially in high-pressure environments like trauma centers, where quick and accurate detection of brain hemorrhages can save lives.

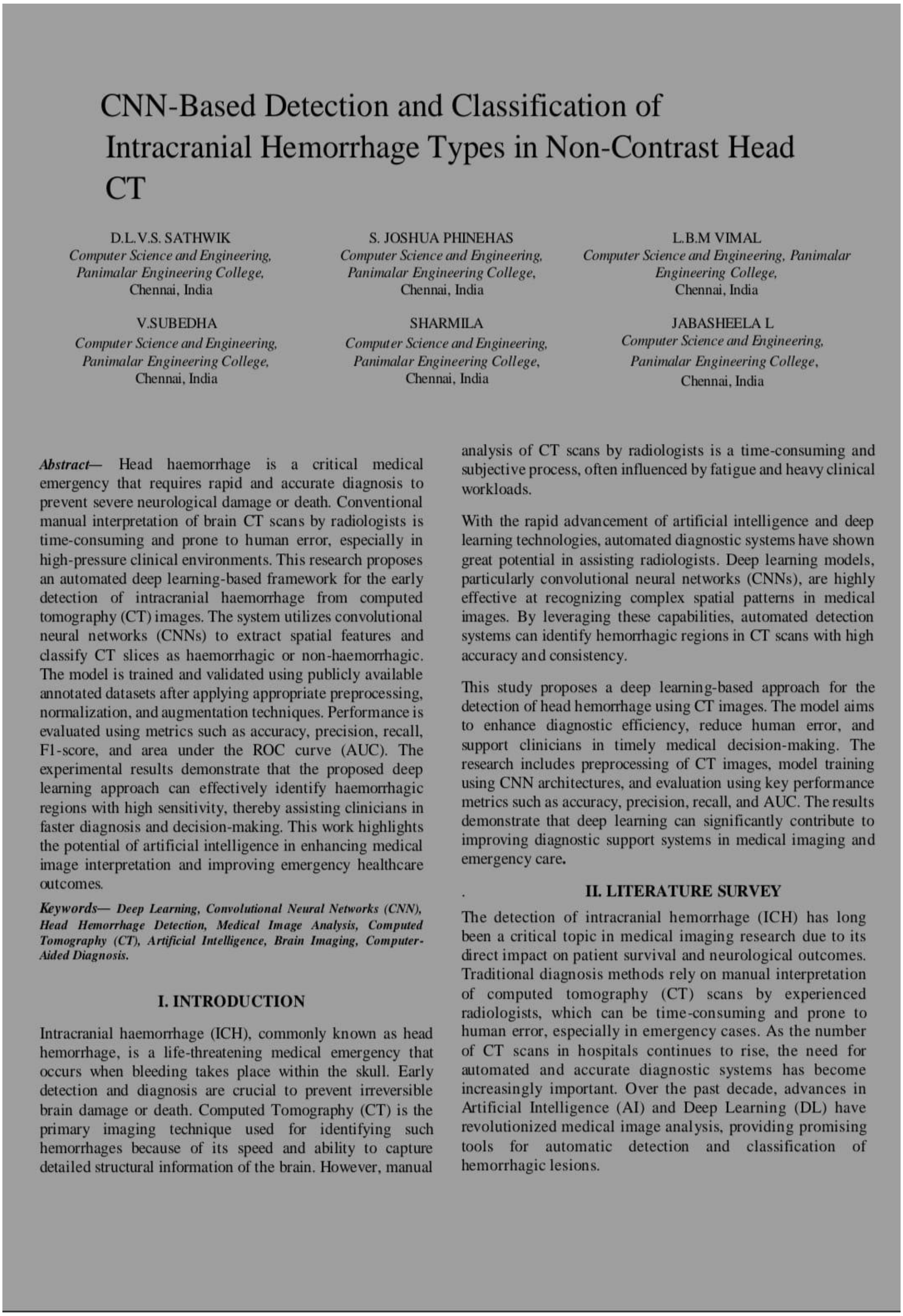
The system is built using a Convolutional Neural Network (CNN) model that analyzes CT images to classify them as either “normal” or “problematic,” indicating the presence of a hemorrhage. The model is trained on annotated medical image datasets and performs image preprocessing such as resizing, normalization, and reshaping to ensure compatibility with the model's input structure. A simple and intuitive Flask-based web interface allows users to upload CT images and receive real-time predictions. The system architecture also incorporates user roles such as admin for managing image submissions and users for uploading and viewing results.

To ensure robustness and usability, the project includes both functional (image upload, classification, role-based access) and non- functional (performance, security, reliability, and usability) features.

The system is designed to require minimal computational resources while maintaining high accuracy, making it scalable for hospitals or remote diagnostic setups. It supports basic image transformations and shows strong performance even when hemorrhages are subtle or hypoattenuating, which are typically harder to detect. Security is implemented through login systems and audit trails, ensuring secure access and traceability. The system adheres to software engineering best practices, including testing (unit, integration, and acceptance), maintainability, and compatibility across different platforms.

It is built primarily using Python, with tools such as Jupyter Notebook and PyCharm, and leverages packages like Tensor Flow/Keras and OpenCV for model execution and image processing. Future enhancements include expanding the dataset, enabling 3D image classification, incorporating sagittal and coronal views, and using active learning to further improve model accuracy. Ultimately, this system provides a foundation for AI- assisted diagnosis in healthcare and opens doors to faster, more reliable decision-making in critical medical situations.

A3- Paper Publication



CNN-Based Detection and Classification of Intracranial Hemorrhage Types in Non-Contrast Head CT

D.L.V.S. SATHWIK
Computer Science and Engineering,
Panimalar Engineering College,
Chennai, India

S. JOSHUA PHINEHAS
Computer Science and Engineering,
Panimalar Engineering College,
Chennai, India

L.B.M VIMAL
Computer Science and Engineering, Panimalar
Engineering College,
Chennai, India

V.SUBEDHA
Computer Science and Engineering,
Panimalar Engineering College,
Chennai, India

SHARMILA
Computer Science and Engineering,
Panimalar Engineering College,
Chennai, India

JABASHEELA L
Computer Science and Engineering,
Panimalar Engineering College,
Chennai, India

Abstract— Head haemorrhage is a critical medical emergency that requires rapid and accurate diagnosis to prevent severe neurological damage or death. Conventional manual interpretation of brain CT scans by radiologists is time-consuming and prone to human error, especially in high-pressure clinical environments. This research proposes an automated deep learning-based framework for the early detection of intracranial haemorrhage from computed tomography (CT) images. The system utilizes convolutional neural networks (CNNs) to extract spatial features and classify CT slices as haemorrhagic or non-haemorrhagic. The model is trained and validated using publicly available annotated datasets after applying appropriate preprocessing, normalization, and augmentation techniques. Performance is evaluated using metrics such as accuracy, precision, recall, F1-score, and area under the ROC curve (AUC). The experimental results demonstrate that the proposed deep learning approach can effectively identify haemorrhagic regions with high sensitivity, thereby assisting clinicians in faster diagnosis and decision-making. This work highlights the potential of artificial intelligence in enhancing medical image interpretation and improving emergency healthcare outcomes.

Keywords— Deep Learning, Convolutional Neural Networks (CNN), Head Hemorrhage Detection, Medical Image Analysis, Computed Tomography (CT), Artificial Intelligence, Brain Imaging, Computer-Aided Diagnosis.

I. INTRODUCTION

Intracranial haemorrhage (ICH), commonly known as head hemorrhage, is a life-threatening medical emergency that occurs when bleeding takes place within the skull. Early detection and diagnosis are crucial to prevent irreversible brain damage or death. Computed Tomography (CT) is the primary imaging technique used for identifying such hemorrhages because of its speed and ability to capture detailed structural information of the brain. However, manual analysis of CT scans by radiologists is a time-consuming and subjective process, often influenced by fatigue and heavy clinical workloads.

With the rapid advancement of artificial intelligence and deep learning technologies, automated diagnostic systems have shown great potential in assisting radiologists. Deep learning models, particularly convolutional neural networks (CNNs), are highly effective at recognizing complex spatial patterns in medical images. By leveraging these capabilities, automated detection systems can identify hemorrhagic regions in CT scans with high accuracy and consistency.

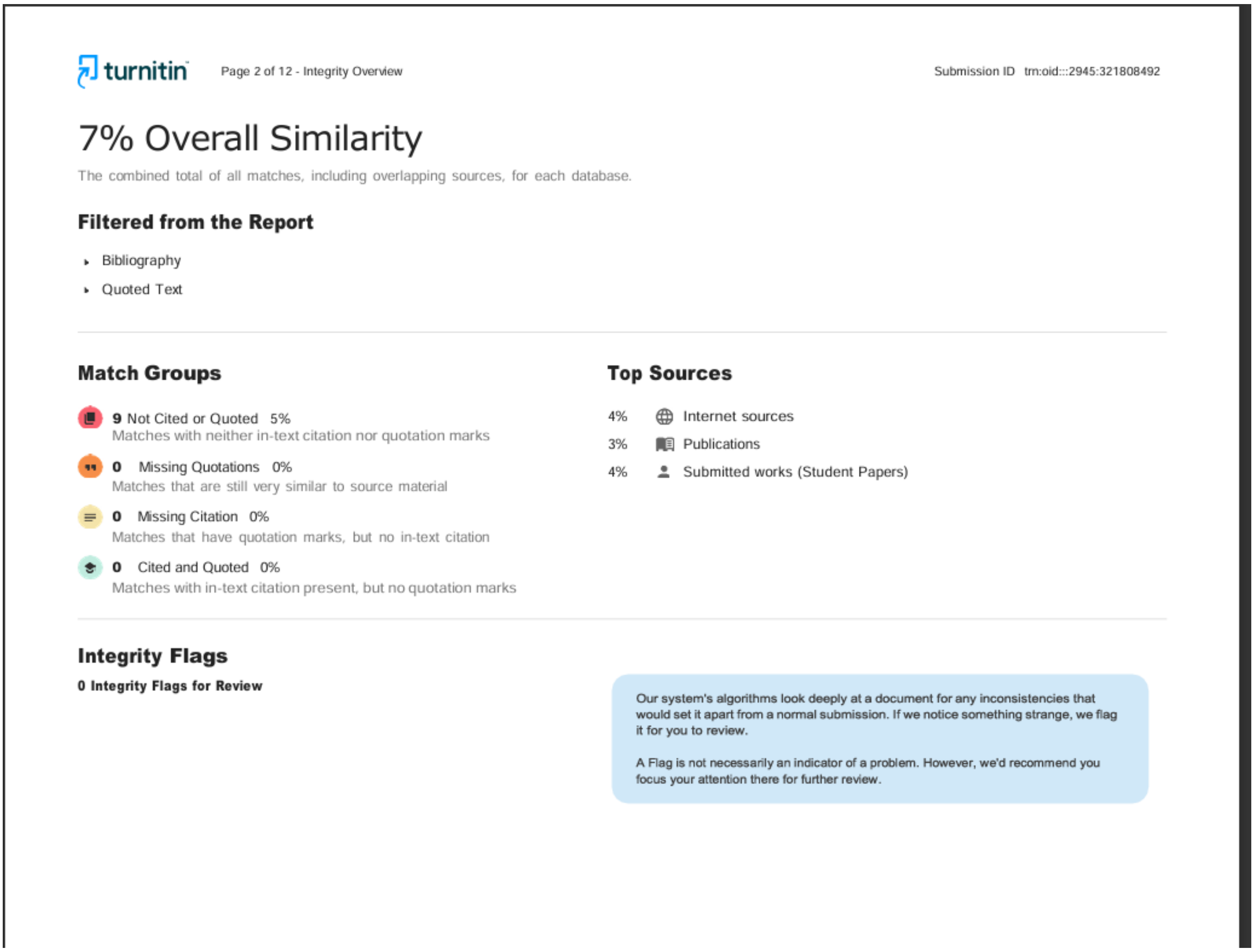
This study proposes a deep learning-based approach for the detection of head hemorrhage using CT images. The model aims to enhance diagnostic efficiency, reduce human error, and support clinicians in timely medical decision-making. The research includes preprocessing of CT images, model training using CNN architectures, and evaluation using key performance metrics such as accuracy, precision, recall, and AUC. The results demonstrate that deep learning can significantly contribute to improving diagnostic support systems in medical imaging and emergency care**.**

.

II. LITERATURE SURVEY

The detection of intracranial hemorrhage (ICH) has long been a critical topic in medical imaging research due to its direct impact on patient survival and neurological outcomes. Traditional diagnosis methods rely on manual interpretation of computed tomography (CT) scans by experienced radiologists, which can be time-consuming and prone to human error, especially in emergency cases. As the number of CT scans in hospitals continues to rise, the need for automated and accurate diagnostic systems has become increasingly important. Over the past decade, advances in Artificial Intelligence (AI) and Deep Learning (DL) have revolutionized medical image analysis, providing promising tools for automatic detection and classification of hemorrhagic lesions.

A4 - Plagiarism report



turnitin Page 2 of 12 - Integrity Overview Submission ID trn:oid:::2945:321808492

7% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text

Match Groups

9 Not Cited or Quoted 5%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

4%	Internet sources
3%	Publications
4%	Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

REFERENCES

[1] Raturi, G., Rani, P., Madan, S., & Dosanjh, S. (2019, November). ADoCW: An automated technique for identifying hidden weapons. In 2019 Fifth International Conference on Image Information Processing (ICIIP) (pp. 181-186). IEEE.

[2] Bhagyalakshmi, P., Indhumathi, P., &Bhavadharini, L. R. (2019). Live video monitoring for automated identification of fire arms. International Journal of Trend in Scientific Research and Development (IJTSRD), 3(3).

[3] Lim, J., Al Jobayer, M. I., Baskaran, V. M., Lim, J. M., Wong, K., & See, J. (2019, November). Utilizing deep neural networks to detect firearms in surveillance videos. In 2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC) (pp. 1998-2002). IEEE.

[4] Yuan, J., & Guo, C. (2018, June). An advanced neural network approach for identifying hazardous machinery. In 2018 Eighth International Conference on Information Science and Technology (ICIST) (pp. 159-164). IEEE.

[5] Ilgin, F. Y. (2020). Utilizing copulas for cognitive radios, we implement energy-based spectrum sensing.. Bulletin of the Polish Academy of Sciences. Technical Sciences, 68(4), 829-834.

[6] Navalgund, U. V., & Priyadharshini, K. (2018, December). Deep learning-based approach for detecting criminal intentions. In 2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET) (pp. 1-6). IEEE.

[7] Chandan, G., Jain, A., & Jain, H. (2018, July). Utilizing Deep Learning and OpenCV, this system enables real-time detection and tracking of objects. In 2018 International Conference on inventive research in computing applications (ICIRCA) (pp. 1305-1308). IEEE.

[8] Deng, L., & Yu, D. (2014). Exploration of deep learning techniques and their practical uses. Foundations and trends® in signal processing, 7(3–4), 197-387.

[9] Bengio, Y. (2009).Studying complex structures for artificial intelligence. Foundations and trends® in Machine Learning, 2(1), 1-127.

[10] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436- 444.

[11] Song, H. A., & Lee, S. Y. (2013). NMF is used to create a hierarchical representation.. In Neural Information Processing: 20th International Conference, ICONIP 2013, Daegu, Korea, November 3-7, 2013. Proceedings, Part I 20 (pp. 466-473).

[12] Masood, S., Ahsan, U., Munawwar, F., Rizvi, D. R., & Ahmed, M. (2020). Image scene recognition using a convolutional neural network. Procedia Computer Science, 167, 1005-1012.

[13] Masood, S., Ahsan, U., Munawwar, F., Rizvi, D. R., & Ahmed, M. (2020). Video scene recognition using a convolutional neural network. Procedia Computer Science, 167, 1005-1012.

[14] Sai, B. K., & Sasikala, T. (2019, November). The task involves utilizing the Tensorflow object identification API to find and quantify things in a picture.. In 2019 International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp. 542- 546). IEEE.

[15] Verma, G. K., & Dhillon, A. (2017, November). A portable firearm detection system utilizing the faster region-based convolutional neural network (Faster R-CNN) deep learning algorithm.. In Proceedings of the 7th international conference on computer and communication technology (pp. 84-88).

[16] Castillo, A., Tabik, S., Pérez, F., Olmos, R., & Herrera, F. (2019). Preprocessing based on brightness enables the automatic detection of cold steel weapons in surveillance films using deep learning. Neurocomputing, 330, 151-161.

[17] Ineneji, C., &Kusaf, M. (2019). A hybrid weapon identification technique is proposed, which combines material testing with a fuzzy logic system.Computers & Electrical Engineering, 78, 437 448.

[18] Dwivedi, N., Singh, D. K., & Kushwaha, D. S. (2019, December). Deep convolutional neural networks for weapon categorization. In 2019 IEEE Conference on Information and Communication Technology (pp. 1-5). IEEE.

[19] Egiazarov, A., Mavroeidis, V., Zennaro, F. M., &Vishi, K. (2019, November). Detection and segmentation of firearms using a combination of semantic neural networks.In 2019 European Intelligence and Security Informatics Conference (EISIC) (pp. 70-77). IEEE.

[20] Abdallah, H. B., Abdellatif, T., &Chekir, F. (2018). AMSEP is an automated system that manages security at multiple levels for multimedia event processing. Procedia computer science, 134, 452-457.

[21]Grega, M., Matiolański, A., Guzik, P., &Leszczuk, M. (2016). Automated identification of images including firearms and knives. Sensors, 16(1), 47. in a CCTV

[22] Tiwari, R. K., & Verma, G. K. (2015). A framework utilizing computer vision and the Harris interest point detector for the purpose of visually detecting guns. Procedia Computer Science, 54, 703- 712.