A

Project II Report

on

SMART TRIAGE SYSTEM FOR PATIENT ASSIGNMENT AFTER DISASTERS

Submitted in Partial Fulfillment of
the Requirements for the Degree
of

Bachelor of Engineering

in

Computer Engineering

to

Kavayitri Bahinabai Chaudhary North Maharashtra University, Jalgaon

Submitted by

Anjali Dongre,
Neelam Netke,
Ritika Rajput,
Varsha Pardeshi

Under the Guidance of

Ms. Dhanashree Tayade

DEPARTMENT OF COMPUTER ENGINEERING
SSBT's COLLEGE OF ENGINEERING AND TECHNOLOGY,
BAMBHORI, JALGAON - 425 001 (MS)

2018 - 2019

SSBT's COLLEGE OF ENGINEERING AND TECHNOLOGY, BAMBHORI, JALGAON - 425 001 (MS)

DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the PROJECT II entitled *Smart Triage System For Patient Assignment After Disasters*, submitted by

Anjali Dongre,
Neelam Netke,
Ritika Rajput,
Varsha Pardeshi

in partial fulfillment of the degree of *Bachelor of Engineering* in *Computer Engineering* has been satisfactorily carried out under my guidance as per the requirement of North Maharashtra University, Jalgaon.

Date: June 10, 2019
Place: Jalgaon

Ms. Dhanashree Tayade
Guide

Prof. Dr. Girish K. Patnaik
Head

Prof. Dr. K. S. Wani
Principal

Acknowledgements

First of all we would like to extend our deep gratitude to almighty God, who has enlightened us with power of knowledge. First, we would like to express my sincere and deep gratitude to Prof. Dr. Kishor Wani, Principal, we wish to express our sincere and deep gratitude to Prof. Dr. Girish K. Patnaik, Head of Computer Department, for giving us such a great opportunity to develop this project. Inspiration and guidance are invaluable in all aspects of life especially on the fields of gratitude and sympathetic attitude which we received from respected project guide, Ms. Dhanashree S. Tayade, guidance and encouragement contributed greatly to completion of this project. We would also like to thank Mr.Akash D.Waghmare, project incharge for extending there precious support and guiding us throughout the project. We would like to thanks to all faculty members of Computer Engineering Department and all friends for his co-operation and supports in making this successful project. We acknowledge my sincere gratitude to all who have directly or indirectly helped us in completing this project successfully.

Anjali Dongre,
Neelam Netke,
Ritika Rajput,
Varsha Pardeshi

Contents

List of Figures

Abstract

The design and implementation of a system to automate patient handling and searching nearby hospitals in mass disasters involving a large number of injured victims over a wireless network. The project intends to apply the mentioned technology to the triage process based on real data of the Emergency Department . This research proposes a model to classify the care priority of patients in emergency using machine learning techniques based on medical criteria and predefined variables as well as quick assignment of patients to nearby hospitals. Patients are prioritized for medical care through a triage process. Manual systems allow for inconsistency and error. The model proposes a novel system to automate accident and emergency center triage and uses this triage score along with an artificial intelligence estimate of patient-doctor time to optimize the queue order. The optimal queue order is found using a novel procedure . The expected chaotic mass-disaster situations can be more suitably controlled and stabilized by using the techniques from this project, thus saving more lives.

Chapter 1

Introduction

The design and implementation of a triage system to automate patient handling and assignment to hospitals in mass disasters involving a large number of injured victims over a wireless network. System includes location-aware features at the disaster site, as well as quick classification and assignment of patients to nearby hospitals.

In section 1.1 background of the machine learning and triage is described. Motivation is described in section 1.2. In section 1.3 problem definition is described. Scope is described in section 1.4. Section 1.5 contains Objective. Identification Of Software Development Process Model is described in Section 1.6. Section 1.7 contains the organization of report. Summary is described in last section.

1.1 Background

In this section background of machine learning and triage is described.

1.1.1 Machine Learning

Machine Learning is a technique that allows computers to learn through programs that generalize behaviors from information or a set of patterns of data. Machine learning algorithms have existed for two decades, but recently, their application has become popular because of growth of power in computing and data storage. It is also important to indicate that there are several models for resolutions of problems in machine learning. Those models can be classified as Geometric, Probabilistic and Logical.Once gotten the final data to be analyzed, three classification algorithms i.e. Naive Bayes Modeling, Logistic Regression and Neural Networks were applied.

- Naive Bayes Modeling:
  For this model the initial probability was defined as 0 and the maximum number of

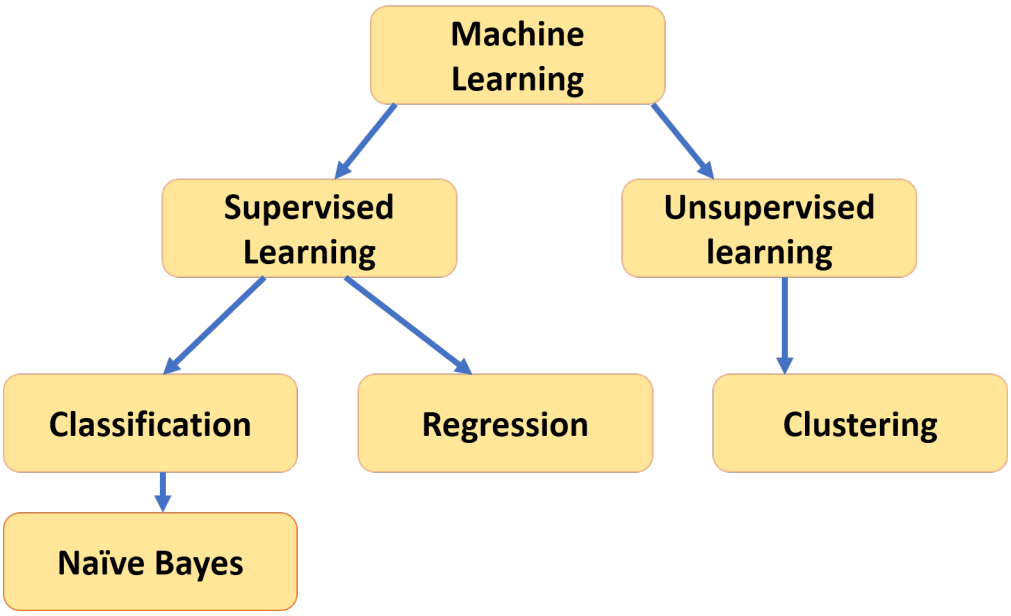

Figure 1.1: Types of Algorithms in Machine Learning

nominal values for each variable was set to 20. The learning node was connected to a Naive Bayes prediction node, where the medical criteria of both the test data and the model prediction are observed. The outputs of the prediction node was connected to enter to a scorer node to show the percentage of hits between the model and the medical criterion.

- Supervised learning:
  Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs.It infers a function from labeled training data consisting of a set of training examples In supervised learning, each example is a air consisting of an input object (typically a vector) and a desired output value (also called the supervisory signal). A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples. An optimal scenario will allow for the algorithm to correctly determine the class labels for unseen instances. This requires the learning algorithm to generalize from the training data to unseen situations in a "reasonable" way.

- Unsupervised learning
  The cluster analysis is a branch of machine learning that groups the data that has not been labelled, classified or categorized. Instead of responding to feedback, cluster analysis identifies commonalities in the data and reacts based on the presence or absence of such commonalities in each new piece of data. A central application of

unsupervised learning is in the field of density estimation in statistics,though unsupervised learning encompasses many other domains involving summarizing and explaining data features. It could be contrasted with supervised learning by saying that whereas supervised learning intends to infer a conditional probability distribution conditioned on the label of input data; unsupervised learning intends to infer an a priori probability distribution.

- Classification in machine learning
  On the other hand, classification algorithms attempt to estimate the mapping function (f) from the input variables (x) to discrete or categorical output variables (y). In this case, y is a category that the mapping function predicts. If provided with a single or several input variables, a classification model will attempt to predict the value of a single or several conclusions. For example, when provided with a dataset about houses, a classification algorithm can try to predict whether the prices for the houses sell more or less than the recommended retail price. Here, the houses will be classified whether their prices fall into two discrete categories: above or below the said price. Examples of the common classification algorithms include logistic regression, Nave Bayes, decision trees, and K Nearest Neighbors.

- Regression in machine learning
  In machine learning, regression algorithms attempt to estimate the mapping function (f) from the input variables (x) to numerical or continuous output variables (y). In this case, y is a real value, which can be an integer or a floating point value. Therefore, regression prediction problems are usually quantities or sizes. For example, when provided with a dataset about houses, and you are asked to predict their prices, that is a regression task because price will be a continuous output. Examples of the common regression algorithms include linear regression, Support Vector Regression (SVR), and regression trees. Some algorithms, such as logistic regression, have the name regression in their names but they are not regression algorithms.

- Clustering
  Clustering is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group than those in other groups. In simple words, the aim is to segregate groups with similar traits and assign them into clusters.

  Broadly speaking, clustering can be divided into two subgroups : Hard Clustering:In hard clustering, each data point either belongs to a cluster completely or not. For example,in the above example each customer is put into one group out of the 10

groups. Soft Clustering: In soft clustering, instead of putting each data point into a separate cluster, a probability or likelihood of that data point to be in those clusters is assigned. For example, from the above scenario each costumer is assigned a probability to be in either of 10 clusters of the retail store.

1.1.2 Triage

Triage is derived from the french term trier which means "to select or choose / to choose or to classify", and it refers to a system that quickly evaluates the severity of each patient and indicates the best treatment depending on his/her condition. The most important triage models i.e. (1) Australian Triage Scale, (2) Triage of the Emergency Department of Canada, (3) Manchester triage system, (4) Urgency severity index, and (5) Triage model Andorra. All of these models have five rating levels, starting from immediate care to treatment after several hours, depending on the patient's symptoms.

- Usage of Decision Trees in Public Health:
  In the last 30 years, different researches have used statistical techniques in the analysis and prediction of information. In this aspect, the field of medicine also have used such techniques to predict infections or diseases, to assign treatment priority, to create decision support systems, and so on. Under and over triage The accuracy of triage can be classified and evaluated using the terms under and over triage. An under triaged patient has life-threatening injuries but is assessed as non-critical, which may lead to delayed medical interventions and/or evacuation to hospital . This inaccurate triage decision may delay the patients immediate care, especially when resources are limited or strained .

- Anatomic and physiologic triage:
  Triage used in the pre-hospital setting is mainly based on anatomic or physiologic data, separately or in combination. Guidelines published in 2010 recommend that triage should be based on a combination of physiologic and anatomic parameters, along with the mechanism of injury, comorbidities, and demographics . By only using anatomic triage decisions based on the patients visible injuries, there is a risk of failing to identify severe injuries such as cavity hemorrhage . On the other hand, a trauma patient may display normal physiologic parameters but have visible signs (i.e., soot in the nostrils after exposure to fire) and be at potential risk of developing later complications .

- Pre-hospital triage:
  Triage systems have several structures in common. Most have a walking filter to identify and rapidly discriminate the most severely injured patients and evacuate them

from the immediate hazard zone . The use of color codes, generally red, yellow, green, and black, to identify severity levels are common in most triage systems. Tags, a practical device used for triage by ambulance personnel, are attached to each patient and follow this color code . The following paragraphs describe some of the most commonly used pre-hospital triage systems.

- Triage Sieve/Triage Sort:
  Triage Sieve and Triage Sort is a two-step triage model described and used in the Major Incident Medical Management and Support course (MIMMS) . The methodology has been widely advocated in the United Kingdom, parts of Australia, and in several regions in Sweden . The first step, Triage Sieve, is intended to be used at the incident site for primary sorting with a walking filter. This stage represent a very rapid form of triage, entirely conducted according to respiratory rate and capillary refill time or heart rate, in order to classify the patient into triage categories .

- Simple Triage and Rapid Transport:
  Simple Triage and Rapid Transport (START) was developed in the 1980s as one of the first civilian triage systems used in MCIs . The START model has been described as one of the most commonly used triage systems for handling MCIs in the United States, and is also used in Canada, Saudi-Arabia, parts of Australia and Israel. The system uses a qualitative, fixed-priority method . The triage categorization is based on whether the patient can walk, respiratory rate, perfusion, and mental status. It is designed so that the provider can complete an assessment within 60 seconds , based on strict medical criteria, and thereby identify the patients medical needs.

1.2 Motivation

Public hospital Accident and Emergency Centre queues are notoriously long. Recent years have seen these hospitals formalizing their medical triage systems, whereby patients are sorted before seeing the doctor to prioritize care to those most urgent. The Cape provinces are beginning to standardize their approaches . However, no such standardization exists in the majority of the country. Furthermore, does not make use of technology. The power of modern Computational Intelligence (CI) techniques has aided many industrial and service processes in becoming more automated and uniform . This project proposes a proof-ofconcept system that employs a wide variety of such techniques encompassing machine learning, expert systems and optimization to automate the process of medical triage and digitally aid the management of a hospitals.

1.3 Problem Defination

The design and implementation of a system to automate patient handling and assignment to hospitals in mass disasters involving a large number of injured victims over a wireless network. In addition, the triage system is modified and enhanced to include location-aware features at the disaster site, as well as quick classification and assignment of patients to nearby hospitals.It is expected that chaotic mass-disaster situations can be more suitably controlled and stabilized by using the techniques from this project, thus saving more lives.

1.4 Scope

Project will determine the process of determining the priority of patients' treatments based on the severity of their condition. This rations patient treatment efficiently when resources are insufficient for all to be treated immediately. The term comes from the French verb trier, meaning to separate, sift or select. Project may result in determining the order and priority of emergency treatment, the order and priority of emergency transport, or the transport destination for the patient. Project may also be used for patients arriving at the emergency department, or telephoning medical advice systems, among others. This project deals with the concept of triage as it occurs in medical emergencies, including the pre-hospital setting, disasters, and emergency department treatment.

1.5 Objectives

Project objectives are the specific objectives for which the project works to achieve them within a stipulated time. They should directly address the problem mentioned in the Problem Definition.

1. Examine Vital signs to calculate triage level.
2. Determine priority of patient by calculating triage level.
3. To find out the nearby hospitals.
4. Assign patient to the hospitals based on priority.

1.6 Identification Of Software Development Process Model

A software process model is a simplified representation of a software process. Each model represents a process from a specific perspective. Were going to take a quick glance about very general process models. These generic models are abstractions of the process that can

be used to explain different approaches to the software development. They can be adapted and extended to create more specific processes. A software process (also knows as software methodology) is a set of related activities that leads to the production of the software. These activities may involve the development of the software from the scratch, or, modifying an existing system. Any software process must include the following four activities:

- Software specification (or requirements engineering): Define the main functionalities of the software and the constrains around them.
- Software design and implementation: The software is to be designed and programmed.
- Software verification and validation: The software must conforms to its specification and meets the customer needs.
- Software evolution (software maintenance): The software is being modified to meet customer and market requirements changes.

1.7 Organization of Report

Organization the report is follows the development process steps:

In Chapter 1, introduction of the system is explain in detailed along with this it includes the background, motivation, problem definition and objective of project.

In Chapter 2, Literature survey along with proposed system, Feasibility study, Risk analysis, Project scheduling, Effort allocation.

In Chapter 3, system requirements specification which includes software, hardware, functional and non functional requirements.

In Chapter 4, system design with the help of various unified modeling language diagrams, architecture and Data ow diagrams.

In Chapter 5, Implementation of system is Specified in detail with the Steps and modules of the system.

In Chapter 6, Details for testing and various types of testing are provided along with the test cases used.

In Chapter 7, Results are declared that came out after development of the system.

In Chapter 8, Further future proposals regarding the system are discussed.

1.8 Summary

In this chapter,Introduction is presented. In the next chapter, Project planning and management is presented.

Chapter 2

Project Planning and Management

A project system information is the coherent organization of the information required for an organization to execute projects successfully. A system information is typically one or more software applications and a methodical process for collecting and using project information.

In the Section 2.1 feasibility study of proposed system is described. Risk analysis is described in Section 2.2. In Section 2.3 project scheduling is described. Effort allocation table is described in Section 2.4.In the section 2.5 cost estimation is described. In the last section summary is presented.

- Literature Survey

  Udaya B Kapu and Raghu B Korrapati in [**?**] has 2015 proposed system a neural network is used to foretell which of the patients seen in an emergency room need to be admitted, transferred to a specialty care or discharged. A multilayer feedforward network model maps input datasets to a corresponding output. The complexity of multilayer feedforward can be altered by changing the number of layers and the number of nodes in each layer. It has been shown that the multilayered neural network can estimate virtually any function to any desired accuracy with the given hidden nodes and enough data. Feedforward describes how the neural network processes the pattern and remembers the patterns. Backpropagation describes how this type of neural network is trained.

  H.A. Chong and K. B. Gan in [**?**] has proposed system in 2016 acquires vital signs, syndrome and chief complaint from patient. The acquired data will be analyzed using triage decision making algorithm and triage level of a patient will be reported instantly. All information and patient records will be stored in database for future reference.

Finally, an automated triage system has been designedand developed to assists assistant medical officer to perform triage assessment in Emergency Room. This system can provide specific triage output compared to the one done by assistant medical officer.

Byron Graham, Raymond Bond, Michael Quinn and Maurice Mulvenna In [**?**] has proposed system in 2018 in which drawing on logistic regression, we identify several factors related to hospital admissions including hospital site, age, arrival mode, triage category, care group, previous admission in the past month, and previous admission in the past year. This study highlights the potential utility of three common machine learning algorithms in predicting patient admissions. Practical implementation of the models developed in this study in decision support tools would provide a snapshot of predicted admissions from the emergency department at a given time, allowing for advance resource planning and the avoidance bottlenecks in patient flow, as well as comparison of predicted and actual admission rates.

Above all systems assigns patient on arrival at hospital which causes overcrowding in certain hospital resulting in risk to patient's life. Available medical equipment and resources of certain hospital may not be sufficient to tackle the patients condition.

- Proposed System

System provides a mechanism to enhance the triage process of the emergency room using machine learning technique based on several independent variables e.g. vital signs, pain scales and Glasgow coma scale, and dependent variables i.e. classification given to the patient. This process will deliver a level of priority based on the Canadian Triage and Acuity Scale (CTAS), which determines 5 levels of attention. The enhanced system identifies the nearest hospitals to a mass-disaster location. As far known, none of the existing patient-data communication schemes handle the patient assignment problem. Pre-assigning patients to most suitable hospitals can reduce the chaos and confusion in a triage room dealing with a mass disaster. A web portal is designed to let the authorized users obtain vital statistics about the overall disaster management scenario.

The client uses the GPS system for self-location. Based on its position, it searches for nearby hospitals using public data and connects to the server-side software of triage system. The position of the client is then transferred to the server to determine the

disaster location. The client software attempts to establish connection with the server and finds nearby hospitals within 50 km. The paramedics use color-coded paper triage tags. The six colour codes include white (non urgent), green (less urgent), yellow (urgent), red (emergent), blue (extremely urgent), and black (dead). Based on the available information, the server checks crates a queue and assigns a triage level to a patient. Next, the server notifies the client about the assignments for all the patients triage levels .A system is implemented at the client (disaster) side that finds the nearest hospitals and automates the process of patient data to the hospitals. In addition, algorithm at the server (hospital) side that assigns patients triage using naive bayes classification algorithm. The GPS location of both i.e. disaster site and hospital will be calculated .Assigned hospital will be informed to send ambulance at the disaster site.

2.1 Feasibility Study

Feasibility studies aim to objectively and rationally uncover the strengths and weaknesses of the existing business or proposed venture, opportunities and threats as presented by the environment, the resources required to carry through, and ultimately the prospects for success. In its simplest term, the two criteria to judge feasibility are cost required and value to be attained. The entire feasibility of the project is comprehended by economical feasibility, operational feasibility and technical feasibility. Feasibility has applied to code clone detection using hybrid approach pertains to the following areas:

1. Economical Feasibility
2. Operational Feasibility
3. Technical Feasibility

2.1.1 Economic Feasibility

Economical feasibility refers to whether the project can be developed at an affordable price. Talking from any organization's point of view, the economical feasibility refers that the organization should be able to finance the project. Moreover, the returns from the project has also to be considered. To decide whether a project is economically feasible, various factors are considered as:

1. Cost benefit analysis
2. Maintenance costs

The proposed system is computer based and it does not require any additional hardware components, hence there is no cost of hardware involved. Considering, the software part required for the project, the project is to be developed in python and python is open source therefore there is no need to required cost for software. It requires average computing capabilities, which are very basic requirements hence it doesn't include additional economic overheads, which renders the system economically feasible.

2.1.2 Operational Feasibility

Operational feasibility of the project describes the ease with which even the naive user can operate the developed system. The developed system should be as easy and user friendly to operate and should be self-comprehensive.To determine the operational feasibility of the system, the awareness level of the users should take into consideration. This system is operational feasible since the users are familiar with the technologies and hence there is no need to gear up the personnel to use system.Also the system is very friendly to use. The Proposed system provide classification using minimum time. Performance of the proposed system is enhanced by lift method. To determine the operational feasibility of the system, the awareness level of the users should take into consideration. This system is operational feasible since the users are familiar with the technologies and hence there is no need to gear up the personnel to use system. Also the system is very friendly to use.

2.1.3 Technical Feasibility

Technical feasibility of a project, is performing a check whether the development of project is possible with the available technological resources. The technical feasibility is a very important aspect to be considered before the official commencement of the project by the organization. The technical feasibility is checked by pondering over the functional requirements of the user. To determine whether the proposed system is technically feasible, the technical issues involved behind the system should taken into consideration. Proposed system uses python technology. Python is an open source technology, it is available for free of cost and conve- niently. As far as platform for the project is concerned, it is decided to perform the project on the window OS. Therefore, the project has to be done on any Windows OS and also on Ubuntu . Thus, it becomes quite sure the project is technically feasible.

2.2 Risk Analysis

Project risk analysis is the identification and quantification of the likelihood and impact of events that may damage the project. Risk analysis is an opportunity to help solve problems

and to enhance communications within the project for a more effective team effort. Risk analysis has applied to code clone detection using hybrid approach pertains to the following areas:

1. Technical Risks
2. Business Risks
3. Project Risks

2.2.1 Technical Risks

Technical risk is the risk that some feature of the correct system can not be implemented due to a technical reason. Technical risk is exposure to loss arising from activities such as design and engineering, manufacturing, technological processes and test procedures. To determine whether the proposed system has technical risk or not, the technical issues involved behind the system should taken into consideration. proposed system uses python technology. As python is an open source technology, it dose not have any technical risk. Thus, no technical risk associated with project.

2.2.2 Business Risks

The term business risk refers to the possibility of inadequate profit or even loss due to uncertainties e.g., changes in tastes, preferences of consumers, strikes, increased competition, change in government policy, obsolesce etc. Business risks implies uncertainty in profits or danger of loss and the events that could pose a risk due to some unforeseen events in future, which causes business to fail.Some of the risks can be

- Unfamiliar to the programming language, e.g. Advanced Java
- Unfamiliar to concept of project.
- Unfamiliar to software tools.
- Considerably resulting in schedule overruns.

There are numerous techniques proposed to classification in software system. The pro- posed system use a lift method for identify classification rules which is able to required less time for classification. Therefore there is no business risk associated with project.

2.2.3 Project Risks

Project risk is defined as an uncertain event or condition that, if it occurs, has a positive or negative effect on a projects objectives. Threaten the project plan, that is, if project risks become real, it is likely that project schedule will slip and that costs will increase. Project risks identify potential budgetary, schedule, personnel (staffing and organization), resource, customer, and requirement problems and their impact on a software project. As per as the project is concerned, the task in project is completed under estimated schedule and project is completed before deadline. Therefore there is no project risk associated with project.

2.3 Project Scheduling

This section specifies the project scheduling of the project. Software project scheduling is an activity that distributes estimated effort across the planed project duration by allocating the effort to specific software engineering task. In this phase we are identifying all major software engineering activities and the product function to which they are applying. As we have selected the linear sequential model for developing our project we divide the work according to the phases of this model. As we are four partners working on this project and having two months,we schedule the project. If the project has been developed according to the schedule, the project schedule defines the task and milestones that must be tracked and controlled as the project proceed. A Gantt chart helps in scheduling the activities of a project, but it does not help in identifying them. One can begin with the activities identified in the work breakdown structure, as we did for the compiler example. During the scheduling activity, and also during implementation of the project, new activities may be identified that were not envisioned during the initial planning. The manager must then go back and revise the breakdown structure and the schedules to deal with these activities .The fig shows the gantt chart of our project, we required 12 weeks to complete our project work, the timeline chart shows the work done in each week and the time required for the same.

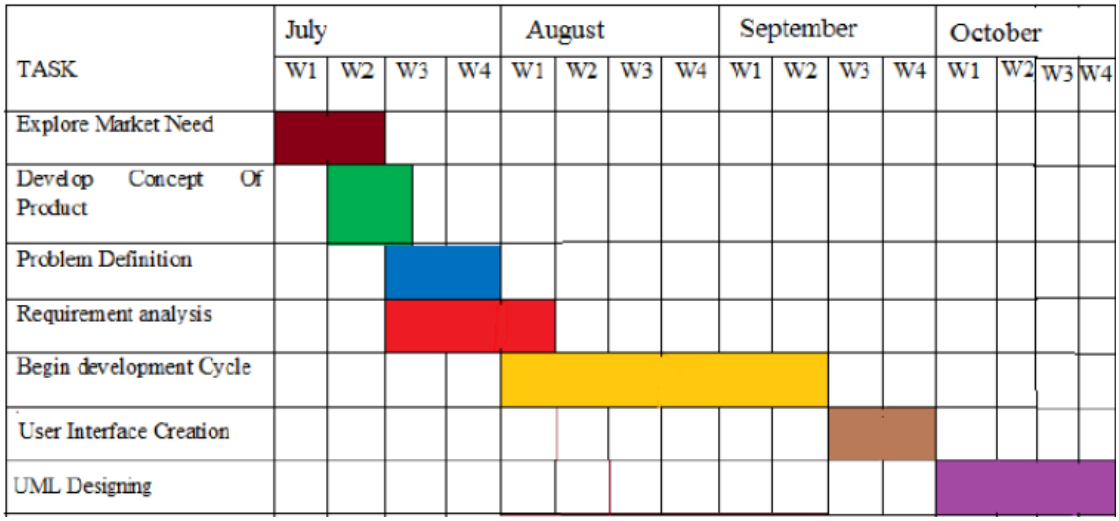

Figure 2.1: Gantt Chart

	Ritika Rajput	Anjali Dongre	Neelam Netke	Varsha Pardeshi
Project Planning	25%	25%	25%	25%
Requirement Gathering	20%	20%	30%	30%
Design	30%	30%	20%	20%
coding	30%	25%	20%	15%
Testing	15%	25%	30%	30%
Documentation	25%	25%	30%	25%

Table 2.1: Effort Allocation

2.4 Effort Allocation

Each of software project estimation techniques leads o estimate of work units requires completing software development. The characteristics of each project dictate the distribution of efforts. There are four members in this project and there are four phases such as Requirement Gathering, Project planning, Design and Coding, Testing. So below shows effort of each member in this project.

2.5 Cost Estimation

Cost of a project is computed on the basis of following key points:

- Number of developers involved.
- How much time each developer gave for development of project.
- Designation of Developer .i.e. senior or junior as the payscale varies for each designation.

- Cost of external resources (server, Support libraries,api etc.).

2.6 Summary

In this chapter, project planning and management is presented. In the next chapter, Analysis is presented.

Chapter 3

Analysis

A Software Requirement Specification (SRS) is a document is created when a detailed description of all aspects of the software to be built must be specified before the project is to commence. It is important to note that a formal SRS is not always written. In fact are many instances in effort expended on SRS might be better spent in other software is to be developed by a third party, software is to developed by a third party, a lack of specification would create severe issues, a system is extremely complex or critical, on SRS may be justified.

Section 3.1 describes Requirement Collection and Identification. Hardware and Software requirements describes in section 3.2. Section 3.3 describes Functional and Non-Functional requirements. Software requirement specification describes in section 3.4 . Finally Summary is presented in last section.

3.1 Requirement Collection And Identification

In the scope of systems and software engineering, requirement modeling is increasingly recognized as a separate activity. Its importance grows with the size and complexity of the intended system. To carry out requirement modeling, a number of different approaches have been developed, many of which are supported by dedicated CASE tools (to name but a few,Caliber RM, Rational Requisite Pro, Catalyze, etc.). The article outlines the requirement modeling approach based on experience accumulated by cybernatic intelligence in a number of medium to large scale software projects, as well as on the results of the internal research into the requirement modeling.

3.2 Hardware And Software Requirement

In this section the hardware and software requirement are described .The requirement of hardware require for computer system for execution.The software requirement is used for

development of the proposed system.

3.2.1 Hardware Requirements

Hardware requirements give the physical component required for the proposed system. The hardware requirement includes a system with following configurations:

1. Processor : Pentium IV or above
2. Display Type : VGA and higher.
3. RAM : 512MB or above
4. Storage Memory : 1GB or above

3.2.2 Software Requirements

The Software Requirements Specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional description, a representation of system behavior, an indication of performance requirements and design constraints, appropriate validation criteria, and other information pertinent to requirements. The various software requirements of the system are summarized here:

1. Operating system : Windows 7/8, Ubuntu.
2. System Type : 64-bit/32-bit operating system.
3. For Development:-
    - Tools:
        (a) PyCharm.
        (b) Postman.
    - Dependencies:
        (a) Python3.
        (b) flask.
        (c) pip Modules.
        (d) pip Version Manager.
        (e) scikit learn.
        (f) pandas.
        (g) numpy.
    - User End:- Web Browsers with HTML5 and Javascript Support.

3.3 Functional Requirement

In software engineering, a functional requirement defines a function of a software system or its component. A function is described as a set of inputs, the behavior, and outputs. Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. The functional requirements of the proposed system are:

1. The system should be able to examine Vital signs to calculate triage level.
2. The system should be able to determine priority of patient by calculating triage level.
3. The system should be able to find out the nearby hospitals.
4. The system should be able assign patient to the hospitals based on priority.

3.4 Non Functional Requirement

A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. This should be contrasted with functional requirements that define specific behavior or functions. The plan for implementing functional requirements is detailed in the system design. The plan for implementing non functional requirements is detailed in the system architecture. The non functional requirements are:

1. The system should be able to take a input dataset.
2. Warnings and error messages should be provided to user throughout the system.
3. Fast response time.
4. Easy enhancement.
5. Execution qualities, such as usability.
6. User interface - The system is designed in such a way that instructions are given clearly to navigate through the System.

3.5 Software Requirements Specification

A software requirements specification (SRS) is a detailed description of a software system to be developed with its functional and non-functional requirements. The SRS is developed

based the agreement between customer and contractors. It may include the use cases of how user is going to interact with software system. The software requirement specification document consistent of all necessary requirements required for project development. To develop the software system we should have clear understanding of Software system. To achieve this we need to continuous communication with customers to gather all requirement.

3.6 Summary

In this chapter, Analysis is presented. In the next chapter, Design is presented.

Chapter 4

Design

System Design chapter provides graphical structure of the project by using various UML diagrams. System design provides the understanding and procedural details necessary for implementing the system recommended in the system study. Design is a meaningful engineering representation of something that is to be built. System can be traced to a customers requirements and at the same time assessed for quality against a set of predefined criteria for good design. In the software engineering context, design focuses on four major areas of concern are data, architecture, interfaces and components.

Section 4.1 describes System Architecture. Database design describes in section 4.2. Section 4.3 dataflow diagram is describes. Interface Design describes in section 4.4. Section 4.5 UML Diagrams are describes . Finally Summary is presented in last section.

4.1 System Architecture

A system architecture or systems architecture is the conceptual model that defines the struc- ture, behavior, and more views of a system. The figure 4.1 shows the working of Smart Triage System in systematic manner.System architecture consist of authorised user disaster site will send patient data at server.Triage level will be calculated and patients will be assigned to nearby hospitals.Assigned data will be send back to user. The enhanced system identifies the nearest hospitals to a mass-disaster location . To the best of our knowledge, none of the existing patient-data communication schemes handle the patient assignment problem. Pre-assigning patients to most suitable hospitals can reduce the chaos and confusion in a triage room dealing with a mass disaster. A web portal is designed to let the authorized users obtain vital statistics about the overall disaster management scenario.

The client uses the GPS system for self-location. Based on its position, it searches for nearby hospitals using public data and connects to the server-side software of triage system.

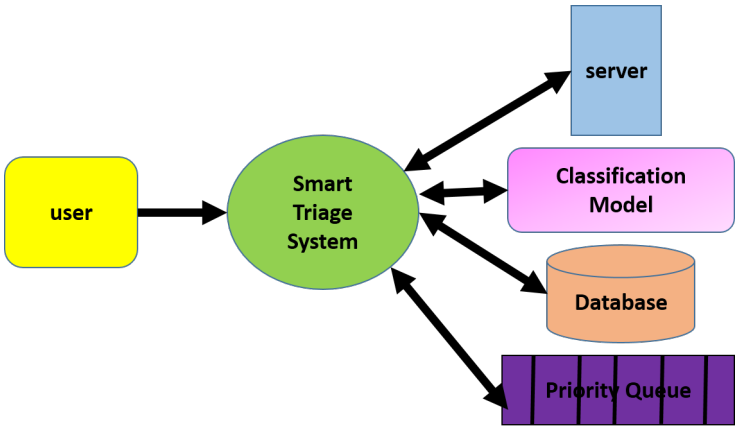

Figure 4.1: System Architecture

The position of the client is then transferred to the server to determine the disaster location. The client software attempts to establish connection with the server and finds nearby hospitals within 50 km. The paramedics use color-coded paper triage tags. The six color codes include white (non urgent), green (less urgent), yellow (urgent), red (emergent), blue (extremely urgent), and black (dead). Based on the available information, the server checks crates a queue and assigns a triage level to a patient.

Next, the server notifies the client about the assignments for all the patients triage levels.A system is implemented at the client (disaster) side that finds the nearest hospitals and automates the process of patient data flow to the hospitals. In addition, algorithm at the server (hospital) side that assigns patients triage using naive bayes classification algorithm. The GPS location of both i.e. disaster site and hospital will be calculated .Assigned hospital will be informed to send ambulance at the disaster site.

4.2 Data Flow Diagrams

As information moves through software, it is modified by a series of transformations.Data Flow Diagram(DFD) is a graphical representation that depicts information flow and the transforms that are applied as data move from input to output. The basic form of a data flow diagram, also known as a data flow graph or a bubble chart. The data flow diagram may be used to represent a system or software at any level of abstraction. In fact, DFDs may be partitioned into levels that represent increasing information flow and functional detail. Therefore, the DFD provides a mechanism for functional modeling as well as information flow modeling.

A level 0 DFD, also called a fundamental system model or a context model. The DFD level 0 shows the abstract of the whole system.

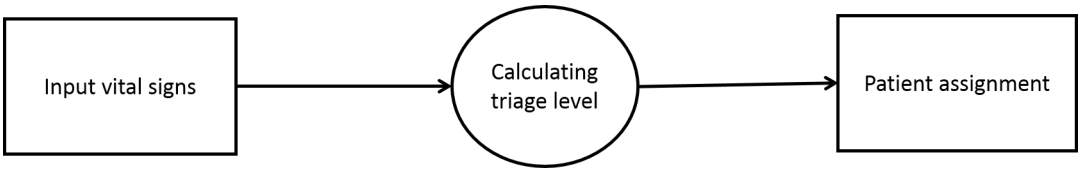

Figure 4.2: DFD Level 0

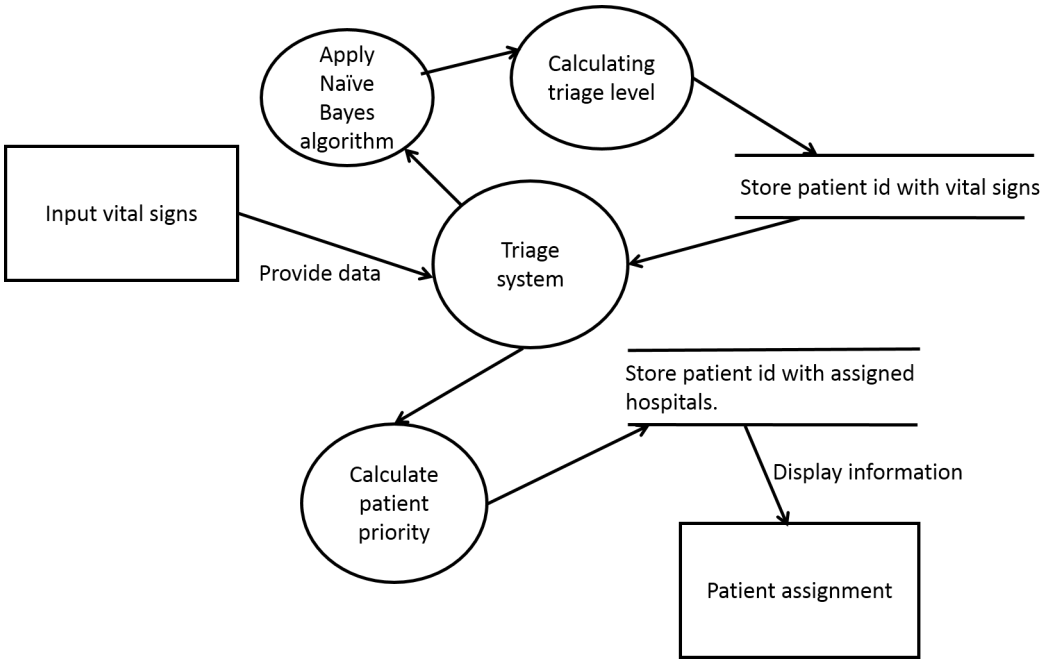

Figure 4.3: DFD Level 1

The Figure shows the level 1 DFD. The DFD level 1 shows some internal structure of the system also identifies data stores that are used by the major processes.

4.3 UML Diagrams

This section illustrates the various UML diagrams of the project. The Unified Modeling Language(UML) is a standard visual modeling language intended to be used for modeling business and similar processes, analysis, design, and implementation of software-based systems.UML is a common language for business analysts, software architects and developers used to describe, specify, design, and document existing or new business processes, structure and behavior of artifacts of software systems. UML is a standard modeling language, not a software development process

4.3.1 Use Case Diagram

A use case diagram displays the relationship among actors and use cases. It addresses the static use case view of a system. These diagrams are especially important in organizing and

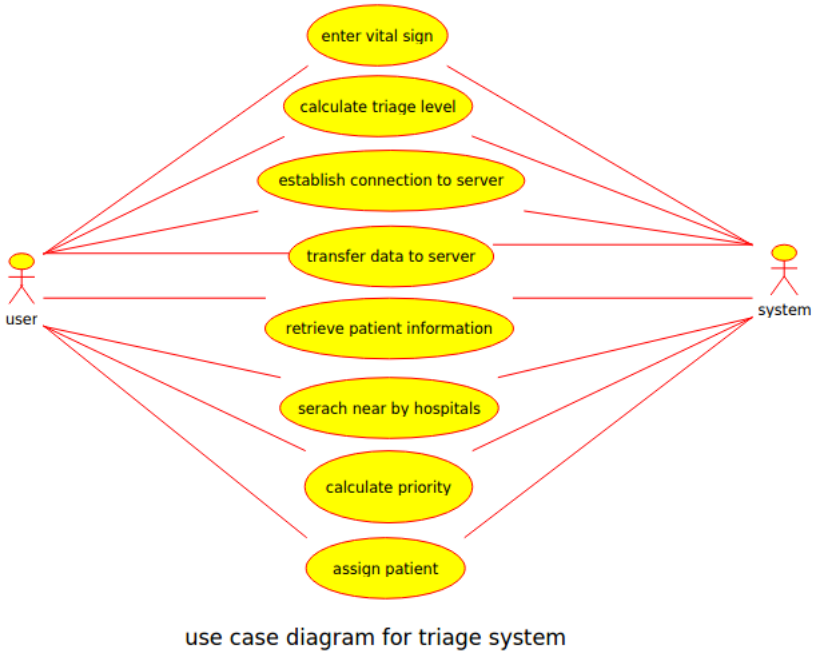

Figure 4.4: Usecase Diagram

modeling the behaviors of a system. An actor is represents a user or another system that will interact with the system you are modeling.

- Enter vital sign- check the symptoms of patient's how the patients are injured means the blood is circulate or not inshort user check the condition of patients
- Calculate triage level- set the priority of the patient's means how serious he is
- Connection to server - system connect to database
- Transfer data to server- system send the data to database or server after triage level.
- Retrieve patient's information- database get the info from server
- Search near by hospitals- as per the patients condition an availability of the seats or bed
- Calculate the priority of the patient's means how serious patient is

Actors are user and system.The use case diagram of proposed system shows the basic functionality such as enter vital sign,calculate triage level,establish connection to server,transfer data to server,retreive patient information,search nearby hospitals,calculate priority.

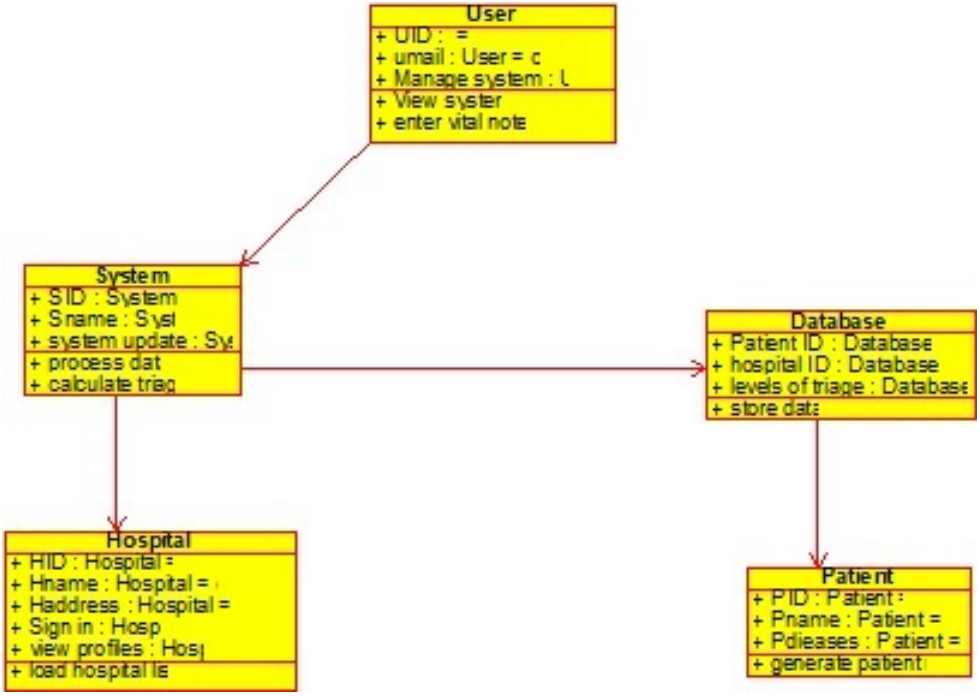

Figure 4.5: Class Diagram

4.3.2 Class Diagram

A Class diagram shows a set of classes, interfaces ,packages and collaborations and their relationships. These are most common diagram found in modelling object-oriented systems. Class diagram address the static design view of a system.

Figure above shows the Class Diagram of proposed system shows the attributes and the operations of the All the classes are described with their attributes and operations that are performed within respective classes.

4.3.3 Sequence Diagram

Sequence diagrams in UML show how objects interact with each other and the order those interactions occur. It addresses the dynamic view of a system. A sequence diagram is an interaction diagram that emphasizes the time ordering of messages.

In tne sequence diagrams there are two User's both are the actors that is system and user. System use to prompt the process for user and user assign the vital sign of patients to system from which triage is allocated an the data of bith actors are stored in database. as shown the first sequence diagram it's phenomena is to enter the vital sign notes and the another for calculate the levels of triage. The dataset of the actors are stored in database that data should be processed on system server before processing the data system find the condition of patients and calculate revised trauma score after that calculation system send the RTS to

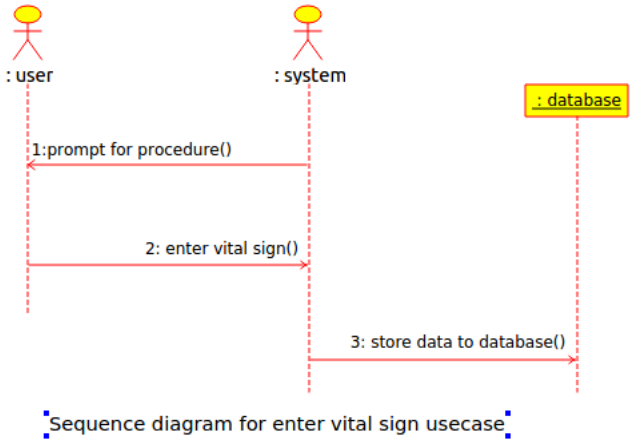

Figure 4.6: Sequence Diagram 1

the database and then triage level is calculated.

4.3.4 Activity Diagram

Activity diagram shows the basic activities between two immediate states of state chart diagram. Activity diagram shows the ow from activity to activity.The activity diagram is a collection of vertices and arcs. Activity diagram commonly contains activity states and action states, transitions and objects.

- Activity diagram which referred to check the condition between to different states. As per the states of activity the system has to entered the vital sign of the patient's.
- After knowing vital sign of patients the priority is allocated to the patient three priorities are assigned in the activity states.
- After assigning the priority as pr checking the vital note's of patients the data send to the server and hospital should assigned for best treatment of the patient's.

Figure shows the Activity diagram of system. If a state is satisfied it transits to next state. It also consists of forking and joining as shown in the figure.

4.3.5 Component diagram

A component diagram shows the organizations and dependencies among a set of components. Component diagrams address the static implementation view of a system. They are related to class diagrams in that a component typically maps to one or more classes, interfaces, or collaborations. Figure shows the component diagram of system.

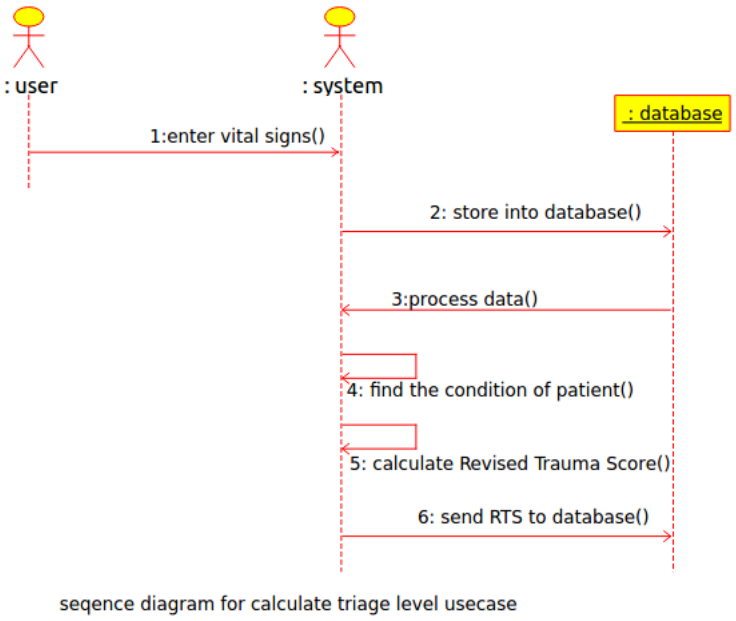

Figure 4.7: Sequence Diagram 2

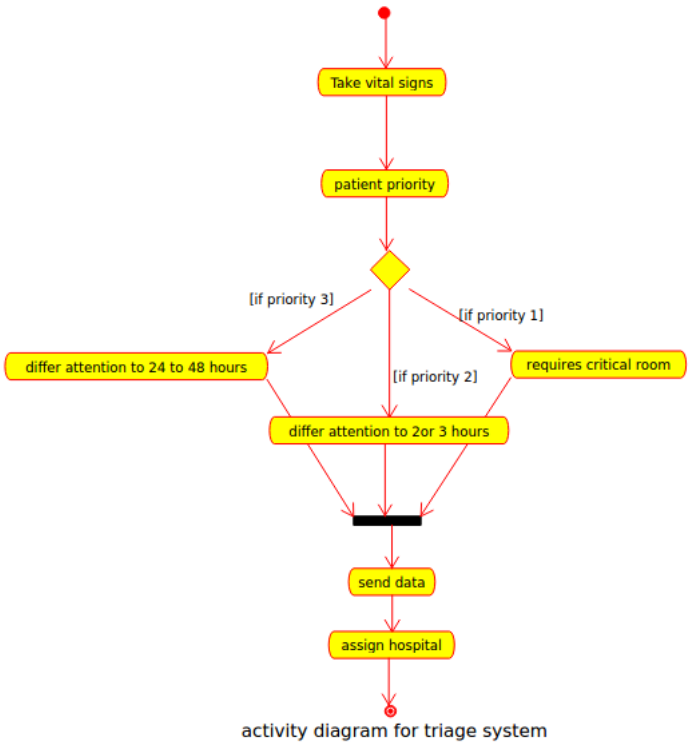

Figure 4.8: Activity Diagram

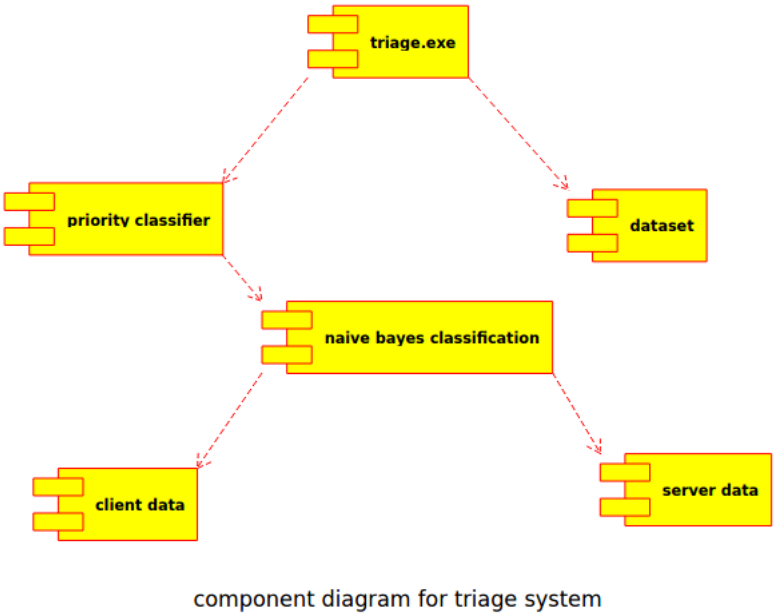

Figure 4.9: Component Diagram

- component diagram provides a physical view of the system .
- It's purpose to shows the dependencies that the software has on the other software components.
- A component is something required to execute a stereotype function .
- A component stereotype of executable , document, database tables, files, or another functions.
- Purpose of the component diagram is to show how components are combined to from longer components of systems.

Proposed system component diagram shows dependency of modules which are use in system such as suspicious detection depend

4.3.6 Deployment diagram

A deployment diagram shows the configuration of run-time processing nodes and the components that live on them. Deployment diagrams address the static deployment view of an architecture. They are related to component diagrams in that a node typically encloses one or more components. Figure shows the deployment diagram of proposed system ,Deployment diagram in- clude the physical component of the system such as internet and different servers. By using private connection we can deploy it to different servers.

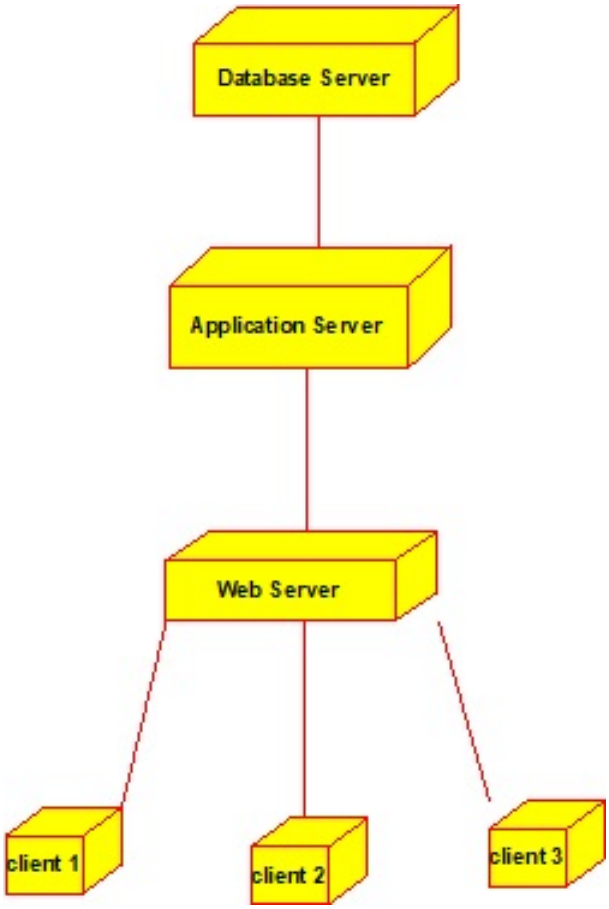

Figure 4.10: Deployment Diagram

- Deployment diagram shows how a physical deployed in the hardware environment.
- Deployment diagram demonstrate the component of the system are operated in the conjunction with each other.
- The models of physical deployment and structure of hardware components are assigned in to the deployment.
- Fhe purpose of the system is to show the different components of the system physically run an how communicate with each other.
- The models of diagram are physical runtime , a system is production of different components and make considerable use of deployment diagram.

4.4 Summary

In this chapter, design of the system are presented. In the next chapter, Implementation are described.

Chapter 5

Implementation

The implemented using Flask and scikit learn.Flask is a micro web framework written in Python. Flask is classified as a microframework because it does not require particular tools or libraries.Flask has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. Json format just consist of a key and value pair. Parameters in the Json data are loaded into a variable and then later on various elements are fetched out of that variable. Data from client side is received in form of Json.Scikit-learn features various classification, regression and clustering algorithm, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.System is using Naive Bayes Algorithm it is a statistical classification technique based on Bayes Theorem.Naive Bayes calculates the triage level of patient.

Section 5.1 describes algorithm. Required software and hardware for development describes in section 5.2 . Section 5.3 describes modules . Finally Summary is presented in last section.

- Simulation Environment

  Flask is a micro web framework written in Python. Flask is classified as a microframework because it does not require particular tools or libraries.Flask has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools. Extensions are updated far more regularly than the core Flask program.In system RESTful request dispatching for accessing classification model is used.The model is stored using pickles and flask supports to use model as an api.

Scikit-learn is a free software machine learning library for the Python programming language.Scikit-learn features various classification, regression and clustering algorithms including support vector machines,random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.System is using Naive Bayes Algorithm it is a statistical classification technique based on Bayes Theorem. Naive Bayes is one of the simplest supervised learning algorithms. Naive Bayes classifier is the fast, accurate and reliable algorithm. Naive Bayes classifiers have high accuracy and speed on large datasets.Naive Bayes classifier assumes that the effect of a particular feature in a class is independent of other features.

5.1 Algorithm

In this section the algorithm for calculating triage is shown and steps of working of different sections is shown.

- Proposed Naive Bayes Algorithm
  - Assuming the number of parameters, classes and probabilities for particular classes
  - Let, N is Number of parameters
  - Let, M[N] is Matrix of N
  - Let, P[N]is Probability of N
  - Let, c is classes
  - Let, Pi is Individual Probability
  - Let, Cn is Number of classes
  - Let, Pn is Number of probability
  - Initialize array for all number of parameters
  - Initialize an array M[N] for N no. of parameters
  - Where N is real number and 1 ¡ N ¡ 20
  - Assuming array of possible values
  - Let, P[N] be array of possible values in M[N]
  - P[N]= 1; 2; 3; .;
  - Calculate individual probability for all classes based on number of parameters
  - Hence, Pi = P (Cn);

- Where 1¡i¡Cn
- Calculate group probability for all combinations of parameters and for all classes
- Hence, Pn = P (n — n + c)
- where, n and c are no .of classes
- Calculate final probability from individual probability of classes and group probability of classes
- Hence, P (Ci — Pn) ¿ P(Cj — Pn)
- Calculate maximum probability for the class, from retrieved final probability
- Hence,
- P(Ci ¿ Pn) = P(N — Ci) P(Cj) — P(N)
- Pmax ¡ P(Ci — Pn)
- Pmax = P(Ci — Pn)

- Building Naive Bayes classifier Model:
  - Step 1: Load dataset from csv file and convert it into dataframe using pandas.
  - Step 2:convert these string labels into numbers .This is known as label encoding. Scikit-learn provides LabelEncoder library for encoding labels with a value between 0 and one less than the number of discrete classes.
  - Step 3: Clean and standardise the data.
  - Step 4: split data into training and testing data.
  - Step 5: Create naive bayes classifier and Fit the dataset on classifier to Perform prediction.
  - Step 6: store model for predicting triage using pickles.
- Server using flask and hospital searching:
  - Step 1: Input will be received from client side in JSON format.
  - Step 2: Load stored model using pickles at server side.
  - Step 3: Return calculated triage to client side in JSON format.
  - Step 4: Foursquare places api is used for finding nearby hospitals according to our provided parameters and endpoints.
  - Step 5: List of nearby hospitals and triage levels will be displayed at client side.

5.2 Required Software And Hardware For Development

In the section describes the required software and hardware for development.

5.2.1 Tools

- PyCharm

  PyCharm is an integrated development environment (IDE) used in computer programming, specifically for the Pythonlanguage. It is developed by the Czech company JetBrains.It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control system (VCSes), and supports web development with Django as well as Data science with Anaconda.PyCharm is cross-platform, with Windows ,macOS and Linux versions. The Community Edition is released under the Apache License,and there is also Professional Edition with extra features released under a proprietary license. The tool provides coding assistance and analysis, with code completion, syntax and error highlighting, linter integration, and quick fixes. The Project and code navigation which provides specialized project views, file structure views and quick jumping between files, classes, methods and usages.It supports python refactoring including rename, extract method, introduce variable, introduce constant, pull up, push down and others.Tool has integrated unit testing, with line-by-line code coverage and Google App Engine Python development.PyCharm has a Version control integration: unified user interface for Mercurial, Git, Subversion, Perforcee and CVS with change lists and merge

- Postman

  API endpoint testing is one of the most important things we do as web developers. If the routes to reach our data are incorrect, the required parameters are not included, the authorization is missing, or a host of other things arent correctly hooked up, your app doesnt work. If your app doesnt work, your customers cant use it. If your customers cant use it, its going to be a really bad day for you, your team, your boss, and so on and so forth. Postman is an API(application programming interface) development tool which helps to build, test and modify APIs. Almost any functionality that could be needed by any developer is encapsulated in this tool. Tool is used by over 5 million developers every month to make their API development easy and simple. It has the ability to make various types of HTTP requests(GET, POST, PUT,

PATCH), saving environments for later use, converting the API to code for various languages(like JavaScript, Python).Tool used postman to test our model.Model sending input in JSON format through pstman to test whether it is working properly and are getting output back correctly.

5.2.2 Technologies

- Flask

  Flask is a micro web framework written in Python. Tool is classified as a microframework because it does not require particular tools or libraries.It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools. Extensions are updated far more regularly than the core Flask program.we have used RESTful request dispatching for accessing classification model. Model stored using pickles and flask supports to use model as an api.

- Scikit-learn

  Scikit-learn is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines,random forests, gradient boosting, k-means and DBSCAN, and is designed to inter operate with the Python numerical and scientific libraries NumPy and SciPy.

  We are using Naive Bayes Algorithm it is a statistical classification technique based on Bayes Theorem. It is one of the simplest supervised learning algorithms. Naive Bayes classifier is the fast, accurate and reliable algorithm. Naive Bayes classifiers have high accuracy and speed on large datasets.Naive Bayes classifier assumes that the effect of a particular feature in a class is independent of other features.

5.2.3 Dependencies

- PIP

PIP is a package-management system used to install and manage software packages written in Python. Many packages can be found in the default source for packages and their dependencies Python Package Index (PyPI). Pip is a recursive acronym for "Pip Installs Packages".Most importantly pip has a feature to manage full lists of packages and corresponding version numbers, possible through a "requirements" file.This permits the efficient re-creation of an entire group of packages in a separate environment (e.g. another computer) or virtual environment.

5.3 Modules

Modules are divided based on users:

- User Module:
  - User Registration User will register himself for using the system.
  - Enter vital signs user will enter vital signs of patients as a input to the system.
  - View calculated trige with patient ids. user can take a view of currently calculated triage and patient id generated by system.
- Smart Triage System Module
  - Process Data System will be responsible for processing data before feeding it to the model.
  - Calculate Triage System will calculate triage according to given data using naive bayes.
- Sever Module
  - Manage Users And Data Admin will have rights to maintain all the users and their rights also the admin will be able to manage all the data related to them.
  - Handle Client Request server will be responsible for handling all the client requests.

5.4 Summary

In this chapter, Impletation are presented. In the next chapter, Testing are described.

Chapter 6

Testing

Software testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is Defect free. System involves execution of a software component or system component to evaluate one or more properties of interest.This chapter briefs about various testing approaches and test cases of the project.

Section 6.1 describes black box testing. White box describes in section 6.2 . Section 6.3 describes manual testing . Test cases identification and execution describes in 6.4. Finally Summary is presented in last section.

6.1 Black Box Testing

Black Box Testing, also known as Behavioral Testing, is a software testing method in which the internal structure/design/implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional. This method is named so because the software program, in the eyes of the tester, is like a black box; inside which one cannot see. This method attempts to find errors in the following categories:

- Incorrect or missing functions
- Interface errors
- Errors in data structures or external database access
- Behavior or performance errors
- Initialization and termination errors

Black Box Testing is not a type of testing; it instead is a testing strategy, which does not need any knowledge of internal design or code etc. As the name black box suggests,no knowledge of internal logic or code structure is required. The types of testing under this strategy

are totally based/focused on the testing for requirements and functionality of the work product/software application. The base of the Black box testing strategy lies in the selection of appropriate data as per functionality and testing it against the functional specifications in order to check for normal and abnormal behavior of the system.

- Functional Testing :- In this type of testing, the software is tested for the functional requirements. The tests are written in order to check if the application behaves as expected.
- Stress Testing :- The application is tested against heavy load such as complex numerical values, large number of inputs, large number of queries etc. which checks for the stress/load the applications can withstand.
- Load Testing :- The application is tested against heavy loads or inputs such as testing of web sites in order to find out at what point the website/application fails or at what point its performance degrades.
- Recovery Testing :- Recovery testing is basically done in order to check how fast and better the application can recover against any type of crash or hardware failure etc. Type or extent of recovery is specified in the requirement specifications.
- User Acceptance Testing :- In this type of testing, the software is handed over to the user in order to find out if the software meets the user expectations and works as it is expected to.

6.2 White Box Testing

White Box Testing (also known as Clear Box Testing, Open Box Testing, Glass Box Testing, Transparent Box Testing, Code-Based Testing or Structural Testing) is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester The tester chooses inputs to exercise paths through the code and determines the appropriate outputs. Programming know-how and the implementation knowledge is essential. White box testing is testing beyond the user interface and into the nitty-gritty of a system.

This method is named so because the software program, in the eyes of the tester, is like a white/transparent box; inside which one clearly sees. White Box Testing method is applicable to the following levels of software testing:

- Unit Testing: For testing paths within a unit.

- Integration Testing: For testing paths between units.
- System Testing: For testing paths between subsystems.

6.3 Manual Testing

Manual testing is the process of using the functions and features of an application as an end-user would in order to verify the software is working as required. With manual testing, a tester manually conducts tests on the software by following a set of pre-defined test cases. Lets take a close look at how this is done. In order to successfully conduct manual tests, you first need to understand the requirements of the software. By understanding the requirements, you will know what needs to be tested and what classifies a defect. This is a crucial part of manual testing as the main goal is to make sure the software is bug-free.

6.4 Test Cases Identification and Execution

Project Name: Data Classification Using SCIP-MA Algorithm.

Test case Template

Pre-condition: User should give the dataset(grocery dataset) as input.

Post-condition: Type of classification.

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly. The process of developing test cases can also help find problems in the requirements or design of an application. Different test cases for different modules are identified and their results are rendered in tabular form.The test cases for various modules and their results are presented in the Table

After running tests, its good to know the results of the tests at a high level. How many tests were run? How many tests failed? How many tests were skipped? TestLodge makes it easy to get a quick glance of these metrics

6.5 Summary

In this chapter, Testing are presented. In the next chapter, Results and Discussion are described.

	Test ID	Test case	Expected Result	Actual Result	Test Result
	1	Vital signs	Must accept all vital signs	Provide Expected Results	Pass
	2	Cleaning	Must clean data accurately	Provide Expected Results	Pass
	3	Label encoding	Must encode label classes correctly	Provide Expected Results	Pass
	4	Clustering	Classification must be according to rules	Provide Expected Results	Pass
	5	Triage Level	Triage levels must be accurate	Provide Expected Results	Pass

Table 6.1: Testing Table

Chapter 7

Results and Discussion

Results that illustrate how the system designed by you works in practice, and how it is intended to be used. Analysis summarizes the qualitative and quantitative analysis that explains why results are relevant. The chapter focuses on results generated by the system and the analysis of the results.

Section 7.1 describes the result and discussion . Finally Summary is presented in last section.

7.1 Results and Discussion

Before analyzing the results generated by the machine learning models, it is important to mention that the current computerized triage process based on an automated algorithm only has a precision of 17% when classifying patients priority levels (comparison based on automated algorithm result vs medical criteria registered by the health professionals after computerized triage process). As mentioned in previous section, two machine learning models were applied using the filtered data to improve the current patients priority level classification system.In the logistic regression model, we obtained a prediction accuracy of approximately 63%.For the Naive Bayes model, the data set had to be grouped into output variable (medical criterion) and each variable was totalized by the defined group. Then, the MAP or maximum were calculated after generating the mean and variance. The model determined a success prediction rate of 70.59%.

	Model	Accuracy	Time
	Naive Bayes	90%	0.8
	Logistic Regression	62.99%	1.2

Table 7.1: Result Table

7.2 Summary

In this chapter result and discussion is presented. In the next chapter, Conclusion is presented.

Chapter 8

Conclusion and Future work

The system describes automate patient handling and searching nearby hospitals in mass disasters involving a large number of injured victims over a wireless network. This paper intends to apply the mentioned technology to the triage process based on real data of the Emergency Department . This research has proposed a naive bayes model to classify the care priority of patients in emergency using machine learning techniques based on medical criteria and predefined variables as well as quick assignment of patients to nearby hospitals. Patients are prioritized for medical care through a triage process. Manual systems allow for inconsistency and error and they are slow and based on persons knowledge and experience. It proposes a novel system to automate accident and emergency center triage and optimal queue order is found using a novel procedure. It is expected that chaotic mass-disaster situations can be more suitably controlled and stabilized by using the techniques from this project, thus saving more lives.

Bibliography

[1] Ruigou yu,Jianrong Wang , Tianyi Xu "Design and Development of a Triage System in Predicting Patient Disposition using Artificial Neural Networks", School of Computer Science and Technology, Tianjin University, Tianjin 300350, China , Received August 31, 2017, accepted October 1, 2017, date of publication November 8, 2017.

[2] Amin Sahba, Arun Das, Paul Rad1 and Mo Jamshidi, "Development of Automated Triage System forEmergency Medical Service.", Department of Electrical and Computer Engineering The University of Texas at San Antonio amin.sahba@utsa.edu, arun.das@utsa.edu, paul.rad@utsa.edu, mojamshidi4@gmail.com

[3] Fares Jalled, Moscow, "Using Data Mining to Predict Hospital Admissions from the Emergency Department.", Moscow Institute of Physics and Technology, Department of Radio Engineering, [cs.CV] 23 Nov 2016.

[4] Alina Campan, Marius Truta, "Sentiment Classification using machine leaarning techinques", Computer science deptartment International journal of computer technology and Eelectronics Engineering, [cs.CV] 23 dec 2017.

Appendix

Figure .1 shows cleaning.

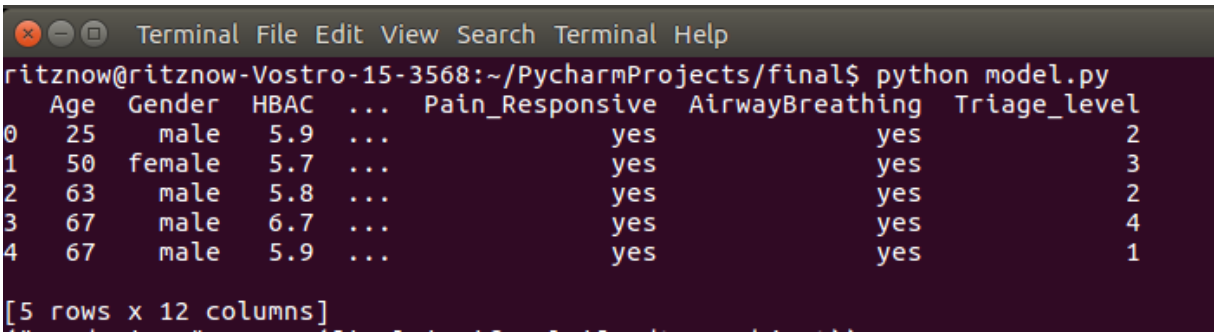

Figure .1: cleaning

Figure .2 shows lable encoding.

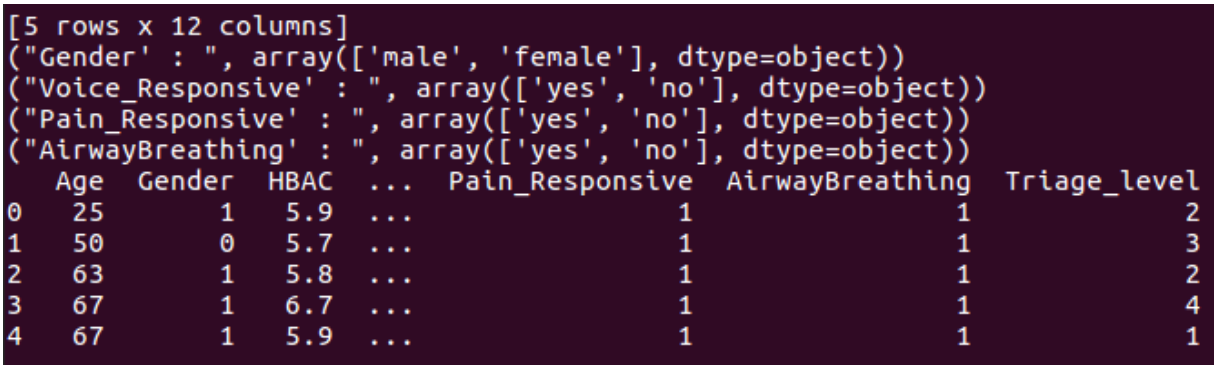

Figure .2: lable encoding

Figure .3 shows splitting,training and testing set.

```
[5 rows x 12 columns]
     Age  Gender  HBAC  ...  Voice_Responsive  Pain_Responsive  AirwayBreathing
617   54       1   6.8  ...                 0                1                1
572   57       1   6.5  ...                 1                1                1
99    66       0   4.0  ...                 1                1                1
78    62       0   4.1  ...                 1                0                0
132   51       0   7.1  ...                 0                1                1

[5 rows x 11 columns]
('training data', (700, 11))
('testing data', (301, 11))
```

Figure .3: splitting,training and testing set

Figure .4 shows calculated triage levels.

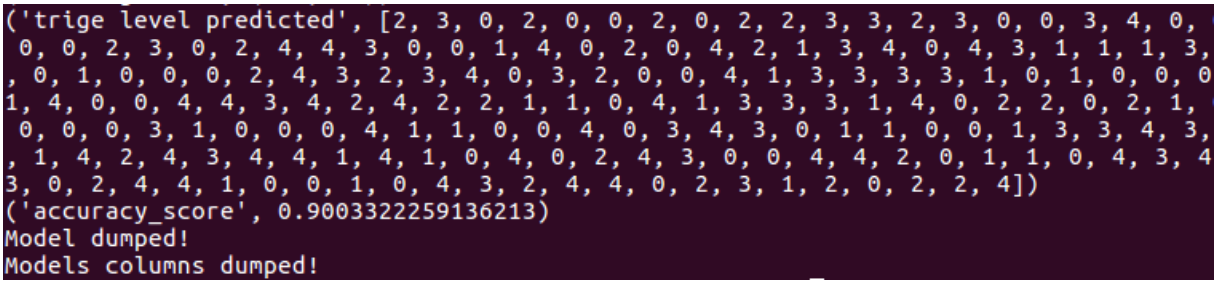

Figure .4: calculated triage levels

Figure .5 shows postman api request

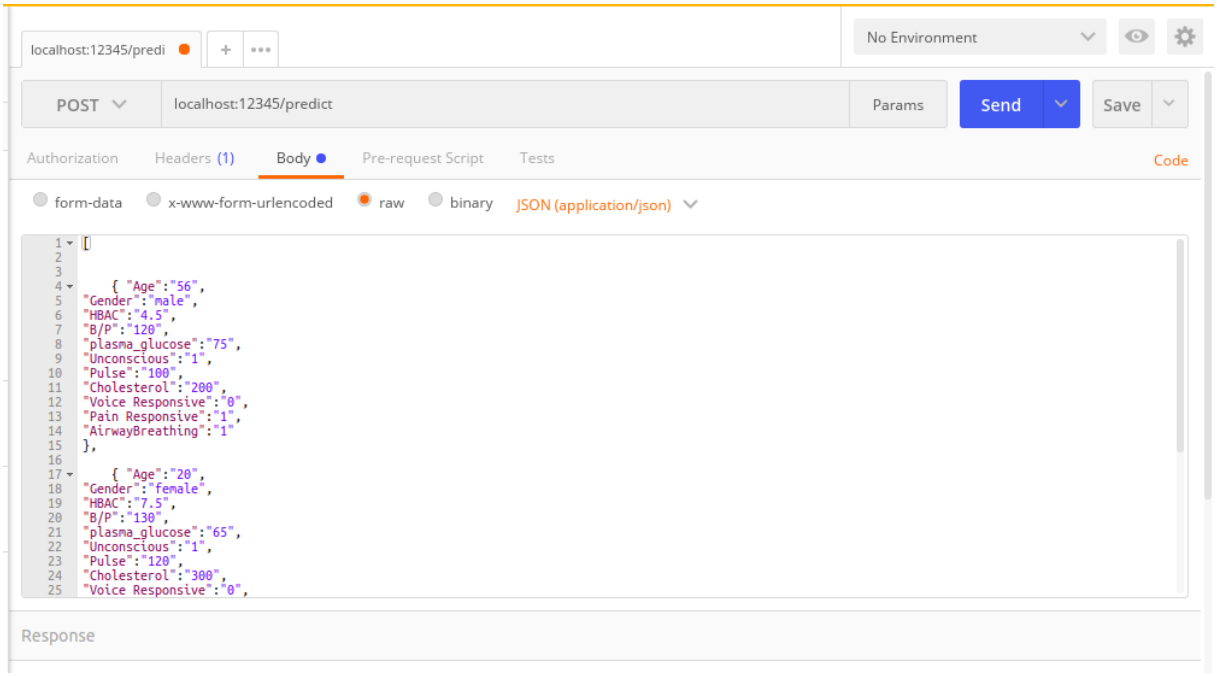

Figure .5: postman api request

Figure .7 shows postman api response

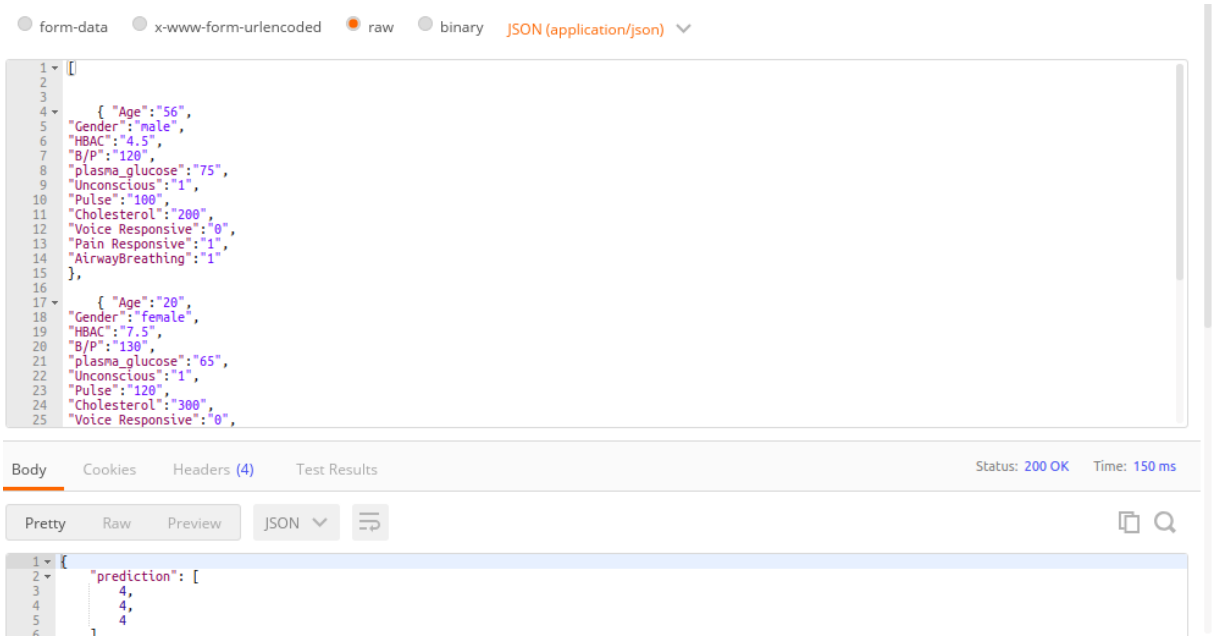

Figure .6: postman api response

Figure .7 shows postman api response

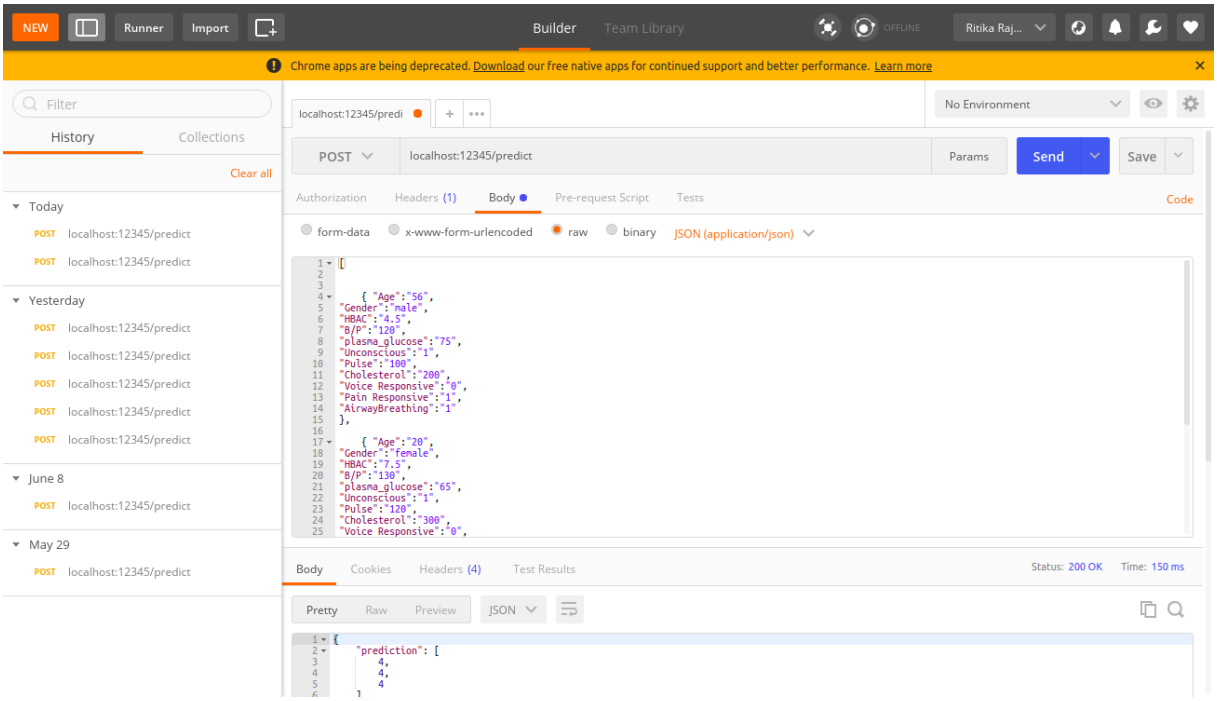

Figure .7: postman api response

Figure .8 shows pycharm ide configuration

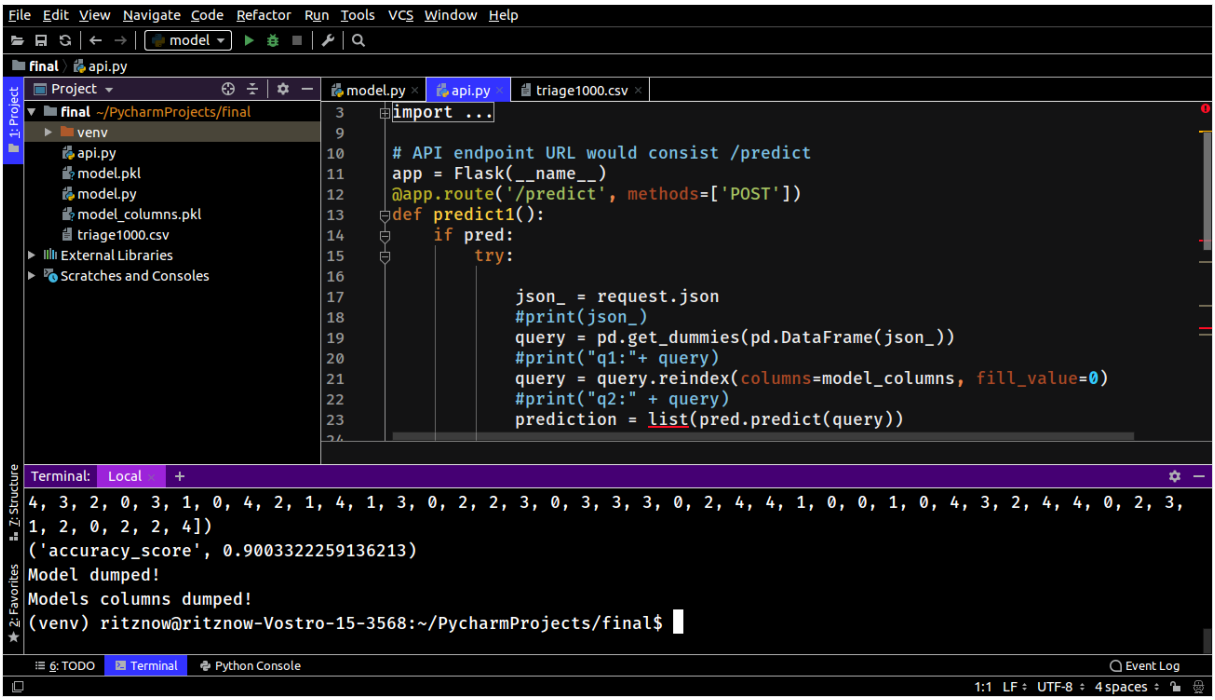

Figure .8: pycharm ide configuration

Index