A PRELIMENERY REPORT ON

GLOTTAL PATHOLOGY DETECTION SYSTEM USING MACHINE LEARNING

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)

SUBMITTED BY

ANUJ CHAVAN	**71600708J**
FEBIN AHMED	**71600701M**
MANMOHAN CHAKOLE	**71526475D**
PRANAV JADHAV	**71600611B**

DEPARTMENT OF COMPUTER ENGINEERING

MAEER's
MAHARASHTRA INSTITUTE OF TECHNOLOGY
PAUD RD-KOTHRUD, PUNE - 411038

SAVITRIBAI PHULE PUNE UNIVERSITY

2019 -2020

CERTIFICATE

This is to certify that project entitled

GLOTTAL PATHOLOGY DETECTION SYSTEM USING MACHINE LEARNING

Submitted by

ANUJ CHAVAN	**71600708J**
FEBIN AHMED	**71600701M**
MANMOHAN CHAKOLE	**71526475D**
PRANAV JADHAV	**71600611B**

is a bonafide student of this institute and the work has been carried out by him under the supervision of **Prof. P K Kosamkar** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the degree of **Bachelor of Engineering (Computer Engineering)**.

Prof P K Kosamkar
Guide,
Department of Computer Engineering

Dr V Y Kulkarni
Head,
Department of Computer Engineering

Dr L K Kshirsagar
Principal,
MAEER's Maharashtra Institute of Technology, Pune - 38

Place: Pune
Date: / / 2020

ACKNOWLEDGEMENT

I have great pleasure in presenting report on **GLOTTAL PATHOLOGY DETECTION SYSTEM USING MACHINE LEARNING**. Completing a task is never a one man effort. It is often a result of invaluable contribution of a number of individuals in direct or indirect manner. I would like to express deepest appreciation towards **Dr L K Kshirsagar**, Principal MAEER's Maharashtra Institute of Technology, Pune – 38 and **Dr V Y Kulkarni**, Head of Department of Computer Engineering, whose invaluable guidance supported me in completing this report.

I am profoundly grateful to **Prof P K Kosamkar** for her expert guidance and continuous encouragement throughout to see that this project reaches its target since its commencement to its completion.

At last I want to express my sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped me directly or indirectly during this course of work.

ANUJ CHAVAN

FEBIN AHMED

MANMOHAN CHAKOLE

PRANAV JADHAV

Date: / / 2020

ABSTRACT

The identification and classification of pathological voice is still a challenging area of research in speech processing. Acoustic features of speech are used mainly to discriminate normal voices from pathological voices.

Here an attempt is made to analyze and to differentiate pathological voice from normal voice using data mining technique like Support Vector Machine (SVM). We conducted cross-validation experiments on The Saarbruecken Voice Database using support vector machines (SVM) for classification of normal and pathological voices. The speech signal is analyzed to extract the acoustic parameters such as 12 Mel-Frequency Filter Bank Cepstral Coefficients (MFCC) and zero-crossing rate (ZCR). The system gives promising accuracy in the detection of Glottal Pathology.

Keywords-*Glottal Pathology Detection, Saarbruecken Voice Database, Support Vector Machines (SVM), 12 Mel-Frequency Filter Bank Cepstral Coefficients (MFCC), Zero-Crossing Rate (ZCR).*

LIST OF ABBREVATIONS

- MFCC - Mel-Frequency Filter Bank Cepstral Coefficients
- ZCR- Zero Crossing Rate
- SVM - Support Vector Machine

List of Figures

3.1 Iterative SDLC Model 14

4.1 System Architecture 16
4.2 DFD Level-0 20
4.3 DFD Level-1 21
4.4 DFD Level-2 22
4.5 ER Diagram 23
4.6 Use Case Diagram 24
4.7 Sequence Diagram 25
4.8 Class Diagram 26
4.9 Component Diagram 27
4.10 Deployment Diagram 28
4.11 State Diagram 29
4.12 Activity Diagram 30

5.1 Task Network Diagram 37
5.2 Timeline Chart 39

6.1 Distinguishing Hyper Plane to Minimize the Error 42
6.2 Separating Hyper Plane by Equation 43

8.1 Confusion Matrix 48
8.2 Precision Table 48
8.3 Plot for Precision 49
8.4 Table for Recall 49
8.5 Plot for Recall 49
8.6 Training 50

8.7 Testing 51
8.8 SVM Train 51

B.1 Introduction Plagiarism 55
B.2 Literature Survey Plagiarism 56
B.3 Proposed System Plagiarism 57
B.4 Conclusion Plagiarism 58

Contents

List of Figures

1 Introduction **1**

1.1 OVERVIEW 1

1.2 MOTIVATION 2

1.3 PROBLEM DEFINITION AND OBJECTIVES 2

1.4 PROJECT SCOPE & LIMITATIONS 3

1.5 METHODOLOGIES OF PROBLEM SOLVING 3

2 LITERATURE SURVEY **4**

3 SOFTWARE REQUIREMENTS SPECIFICATION **7**

3.1 ASSUMPTIONS AND DEPENDENCIES 7

3.2 SYSTEM FEATURE 1 (FUNCTIONAL REQUIREMENTS) 7

3.3 EXTERNAL INTERFACE REQUIREMENTS 8

3.3.1 User Interfaces 8

3.3.2 Hardware Interfaces 8

3.3.3 Software Interfaces 8

3.3.4 Communication Interfaces 8

3.4 NON-FUNCTIONAL REQUIREMENTS 9

3.4.1 Performance Requirements 9

3.4.2 Safety Requirements 9

3.4.3 Security Requirements 10

3.4.4 Software Quality Attributes 10

3.5 SYSTEM REQUIREMENTS 10

3.5.1 Database Requirements 10

3.5.2 Hardware Requirements 10
3.5.3 Software Requirements 11
3.6 NON-FUNCTIONAL REQUIREMENTS 13
3.6.1 Performance Requirements 13
3.7 ANALYSIS MODELS: SDLC MODEL TO BE APPLIED 14

4 SYSTEM DESIGN 16
4.1 SYSTEM ARCHITECTURE 16
4.2 MATHEMATICAL MODEL 18
4.3 DATA FLOW DIAGRAMS 20
4.3.1 DFD Level-0 20
4.3.2 DFD Level-1 21
4.3.3 DFD Level-2 22
4.4 ENTITY RELATIONSHIP DIAGRAMS 23
4.5 UML DIAGRAMS 24
4.5.1 Use Case Diagram 24
4.5.2 Sequence Diagram 25
4.5.3 Class Diagram 26
4.5.4 Component Diagram 27
4.5.5 Deployment Diagram 28
4.5.6 State Diagram 28
4.5.7 Activity Diagram 29

5 PROJECT PLAN 31
5.1 PROJECT ESTIMATE 31
5.1.1 Reconciled Estimates 31
5.1.2 Project Resources 31
5.2 RISK MANAGEMENT 32
5.2.1 Risk Identification 32
5.2.2 Risk Analysis 33
5.2.3 Overview of Risk Mitigation, Monitoring, Management 33
5.3 PROJECT SCHEDULE 35
5.3.1 Project Task Set 35

5.3.2 Task Network 37
5.3.3 Timeline Chart 38
5.4 TEAM ORGANIZATION 38
5.4.1 Team Structure 38
5.4.2 Management Reporting and Communication 40

6 PROJECT IMPLEMENTATION 41
6.1 OVERVIEW OF PROJECT MODULES 41
6.2 TOOLS AND TECHNOLOGIES USED 41
6.3 ALGORITHM DETAILS 42
6.3.1 Support Vector Machine 42

7 SOFTWARE TESTING 44
7.1 TYPE OF TESTING 44
7.2 TEST CASES & TEST RESULTS 45

8 RESULTS 47
8.1 OUTCOMES 47
8.2 SCREEN SHOTS 50

9 CONCLUSION 52
9.1 CONCLUSION 52
9.2 FUTURE WORK 52
9.3 APPLICATIONS 52

Annexure A Feasibility Assessment 53

Annexure B Plagiarism Report 55

10 References 59

Chapter 1

Introduction

1.1 OVERVIEW

Voice-based identity verification to voice pathology detection, is nowadays ubiquitous in our daily life. A significant attention has been paid to the science of voice pathology's diagnostic and monitoring as it offers solution to companies seeking for efficiency enhancement with simultaneous cost saving, the market of speech technology seems to be particularly promising in the coming years. The voice disorders are caused due to defects in the speech organs, mental illness, hearing impairment, autism, paralysis or multiple disabilities.

The main aim of this system is to help patients with pathological problems for monitoring their progress over the course of voice therapy. Currently, patients are required to routinely visit a specialist to follow up their progress. Furthermore, the existing ways to detect the voice pathology are subjective, invasive methods such as the direct inspection of the vocal folds and the observations of the vocal folds by endoscopic instruments are done. In addition, the present systems are usually based on information related to the vocal tract configuration, the airflow passing through the vocal folds, and called glottal flow. These techniques are expensive, risky, time consuming, discomfort to the patients and require costly resources, such as special light sources, endoscopic instruments and specialized video-camera equipment.

In order to avoid the above problems, a robust system is implemented to detect vocal fold pathology at an early stage from set of features like 12 Mel-Frequency Filter Bank Cepstral Coefficients (MFCC) and zero crossing rate (ZCR) derived from simple voice sample. The

system helps the clinicians and speech therapists for early detection of vocal fold pathology and can improve the accuracy of the assessments using Saarbruecken Voice Database.

The proposed system can analyze voice source features in speech data for detection of glottal pathology using data mining techniques. SVM classifier is developed for various feature combinations to classify the glottal pathologic voice from normal voice. The system is implemented using 12 Mel-Frequency Filter Bank Cepstral Coefficients (MFCC) and zero crossing rate (ZCR).

1.2 MOTIVATION

- The voice disorders are mainly caused due to defects in the speech organs, mental illness, hearing impairment, autism, paralysis or multiple disabilities. The traditional ways to diagnose voice pathology like direct inspection of the vocal folds and the observations of the vocal folds by endoscopic instruments are done.
- Such existing techniques are expensive, risky, time consuming, discomfort to the patients and require costly resources, such as special light sources, endoscopic instruments and specialized video-camera equipment.
- The proposed system can give the low cost and quick solution by analyzing voice source features like 12 Mel-Frequency Filter Bank Cepstral Coefficients (MFCC) and zero crossing rate (ZCR) in speech data for detection of glottal pathology.

1.3 PROBLEM DEFINITION AND OBJECTIVES

To implement a system i.e. “Glottal Pathology Detection System Using Machine Learning” to differentiate pathological voice from normal voice using Support Vector Machine (SVM).

Objectives

- To analyze and to differentiate pathological voice from normal voice using data mining technique.
- To detect vocal fold pathology at an early stage from set of features like MFCC and ZCR, which are vocal tract parameters.

1.4 PROJECT SCOPE & LIMITATIONS

The proposed system can help to detect glottal pathologies from the voice input. The system uses MFCC which are vocal tract parameters, in detection of the glottal pathologies and ZCR for the detection. The SVM classifier is used to train the data. The classification of pathologies is carried out with the help of SVM classifier.

Limitation

Internet connectivity should be always on at all the times.

1.5 METHODOLOGIES OF PROBLEM SOLVING

- This platform is rapidly growing with user's need which overcomes the issues of differentiating pathological voice from normal voice using data mining technique.
- Software project estimation is form of problem solving used here. The complex software is hard to estimate hence it is divided into smaller pieces. The estimation of project will be correct only when the estimation of size of the project is correct. In the context of project planning size refers to quantifiable outcome of project.
- Here, the direct approach is selected and hence, the size is estimated in Line of Codes. The feasibility study comprise of an initial investigation into personnel will be required.
- Feasibility study will enable us to make informed and straightforward choice at crucial points while developing phase. All projects are feasible given unlimited time and resources. But, the development of computer-based system is more likely to be plagued to scarcity of resources. It is both essential and prudent to evaluate the feasibility of project at earliest possible time.
- This particular system helps to detect vocal fold pathology at an early stage from set of features. The system analyzes and differentiates pathological voice from normal voice using data mining technique from set of features like MFCC and ZCR, which are vocal tract parameters.

Chapter 2

LITERATURE SURVEY

Pranav S. Deshpande et al.[2] presents a method for reliable detection of glottal instants and EGG parameters from an electroglottographic (EGG) signal composed of voiced and nonvoice segments. An adaptive variational mode decomposition-based algorithm is used or suppressing low-frequency artifacts and additive high-frequency noises. Depending on the center frequency criterion, the proposed system first constructs a candidate EGG feature signal for the determination of glottal closure and opening instants. In the second stage, the candidate glottal instants are determined by detecting the positive and negative zero crossings in normalized candidate EGG feature signals [15], respectively. Finally, an autocorrelation features based post-processing algorithm is presented to reject nonglottal instants from the nonspeech production segments.

S.C.Punitha et al. [3] proposed Voice Pathology Identification System using SVM Classifiers. The experiments are performed on their Database. Mel-frequency Cepstrum was taken as Feature Extraction Technique.

D. Pravena et al. [4] used the MFCC as a main feature extraction technique and to classify the voice signals by using the Gaussian Mixture Model (GMM).

Anis Ben Aicha [6] proposed an automatic detection method of premalignant lesions based on human voice production theory. The non-invasive process based on the recorded speech is used for experiments that get an accuracy of 92%. The source signal is first extracted namely the glottal flow signal from the acoustic speech using the IAIF technique.

The relevance of used descriptors using boxplot and PCA [6] analysis is also done. SVM module is used to classify and discriminate premalignant lesions from normal voices.

Vahid et al., [7] give the study of feature extraction and feature reduction in the task of vocal fold pathology diagnosis. A new type of feature vector, based on wavelet packet decomposition and Mel-Frequency-Cepstral-Coefficients (MFCCs), was proposed. Also, Principal Component Analysis (PCA) was used for feature reduction. An Artificial Neural Network was used as a classifier for evaluating the performance of the proposed method. The database was created by specialists from the Belarusian Republican Center of Speech, Voice and Hearing Pathologies. The algorithm gives the best result of accuracy.

V. Srinivasan et al., [8] used a method of finding the ability of acoustic parameters in discrimination of normal voices from pathological voices that were analyzed and classified. The classification of pathological voice from a normal voice was implemented using a support vector machine (SVM) and the classifiers were trained and tested. The dataset was recorded by speech utterances of a set of Tamil phrases containing speech samples of 10 distinct subjects (5 normal, 5 pathological children). The speech signals were analyzed and were extracted. A Genetic Algorithm (GA) [9] based feature selection has improved the classification accuracy of this work. The support vector machine shows better performance in terms of classification accuracy.

J. Hernando and P. Ejarque [9] give the study of jitter and shimmer which have been largely used for the description of pathological voice quality. Several types of jitter and shimmer measurements have been analysed .

Markaki et al. [10] give a joint acoustic and modulation frequency representation, i.e. Modulation Spectrum, of sustained vowel /AH/ for detection and discrimination of voice disorders. The database of sustained vowel recordings from healthy and pathological voices is used, with support vector machines (SVM) for classification with accuracy of 94.1%.

V. Sellam, J. Jagadeesan [11] provides a classification of pathological voice from normal voice using Support Vector Machine (SVM) and Radial Basis Functional Neural Network

(RBFNN) with the dataset of Tamil phrases. The voice features like Signal Energy, pitch, formant frequencies, Mean Square Residual signal, Reflection coefficients, Jitter and Shimmer are taken into consideration to detect voice disorders in children.

Korutla Sudhir Sai et al. [12] presents solutions to companies seeking for efficiency enhancement in the field of speech technology. This work involves the advancement in glottal analysis in order to bring down new techniques within the speech processing advancements.

Christophe Dalessandro and Nicolas Sturmel et al. [13] provides introduction to Line of Maximum Amplitude (LoMA) method when time-scale representation of voiced speech is applied to voice quality. This representation takes advantage of the tree patterns observed for voiced speech periods in the time-scale domain.

L Gavidia-Ceballos and J H Hansen [14] used speech production parameters as complete glottal closure is very hard to obtain in vocal fold pathology. The method proves advantageous in the estimation of Enhanced Spectral Pathology Component (ESPC) instead of glottal flow waveform, which varies considerably between pathology and health conditions. The meaningful measures Mean- Area-Peak-Value (MAPV) and Weighted-Slope (WSLOPE) were derived from the ESPC feature.

Chapter 3

SOFTWARE REQUIREMENTS SPECIFICATION

3.1 ASSUMPTIONS AND DEPENDENCIES

Let us Assume:

- User must have basic knowledge of computer.
- User must have basic knowledge of handling Webpages.

Dependencies:

- Only Administrators will be able to edit main configurations.
- Administrators will communicate among themselves while executing the application.

3.2 SYSTEM FEATURE 1 (FUNCTIONAL REQUIREMENTS)

- **Speech Input:** Speech Data is given to the system.
- **Preprocessing:** Input data is preprocessed to get the fine-tuned data.
- **Feature Extraction:** Features like MFCC and ZCR are extracted from input.
- **Classification:** SVM classifier classify the glottal pathologic voice from normal voice.

3.3 EXTERNAL INTERFACE REQUIREMENTS

3.3.1 User Interfaces

- The system will provide the access control over the functionality according to the user roles like students and staff.
- User interface will provide good look and provides user friendly environment to its users.
- User interface of an application/software follows world user interface standard thus he or she can operate system very efficiently.

3.3.2 Hardware Interfaces

The minimum configuration required on computer.

- **Processor:** 1 gigahertz (GHz) or faster processor or SoC.
- **RAM:** 1 gigabyte (GB) for 32-bit or 2 GB for 64-bit.
- **Hard disk space:** 16 GB for 32-bit OS 20 GB for 64-bit OS.

3.3.3 Software Interfaces

- **IDE:** Eclipse Luna
- **Platform:** Microsoft Windows 7 Professional or greater
- **Language:** Java 1.8, Python

3.3.4 Communication Interfaces

- Communication Interface process is intended to give an approach to archive and track extend interfaces from Planning stage (FEP) to the end of the project.
- The system uses the HTTP protocol for communication over the internet and for the intranet communication it will be done with TCP/IP protocol suite.

3.4 NON-FUNCTIONAL REQUIREMENTS

3.4.1 Performance Requirements

High Speed

The system should process the requested task in parallel for various activities to give a quick response then the system must wait for process completion.

Accuracy

The system should correctly execute the process; i.e. display the result according to the particular parameter.
System output should be in user required format.

Interoperability

System should have the ability to exchange information and communicate with internal and external applications and systems. It must be able exchange information both internally and externally.

Response Time:

The response time of the system should be deterministic at all times and very low, i.e. it should meet every deadline. Thus, the system will work in real time.

3.4.2 Safety Requirements

- The data safety must be ensured by arranging for a secure and reliable transmission media.
- The source and destination information must be entered correctly to avoid any misuse or malfunctioning.
- Safety requirements against the natural disaster and accidents.
- Failures due to technical issues.

3.4.3 Security Requirements

- All the user details shall be accessible to only high authority persons.
- Access will be controlled with usernames and passwords.

3.4.4 Software Quality Attributes

- Software should be Maintainable.
- Encourage in-code documentation (XML docs in javadoc, etc.)
- Use a wiki to maintain the documentation.
- Unit Tests = Good for documenting specifications.
- Comments = Good for documenting design decisions
- Unit Tests + Comments = Good for documenting specifications and design decisions. = Easily maintainable software.
- Faster feedback from any changes made to the system.
- Providing better transparency into the changes happening to the system.
- Propagating environmental changes and code changes more rapidly while maintaining control.

3.5 SYSTEM REQUIREMENTS

3.5.1 Database Requirements

The database is required to be created and maintained in MySQL Server. Stored procedures are also created to retrieve and operate on data.

3.5.2 Hardware Requirements

The minimum configuration required on server platform.

- **Processor:** 1 gigahertz (GHz) or faster processor or SoC.

- **RAM:** 1 gigabyte (GB) for 32-bit or 2 GB for 64-bit.
- **Hard disk space:** 16 GB for 32-bit OS 20 GB for 64-bit OS.

3.5.3 Software Requirements

- Eclipse Luna is an open source community whose projects building tools and frameworks are used for creating general purpose application. The most popular usage of Eclipse is as a Java development environment.
- Eclipse Luna is an open source community, whose projects are focused on building an open development platform comprised of extensible frameworks, tools and runtimes for building, deploying and managing software across the lifecycle. The Eclipse Foundation is a not-for-profit, member supported corporation that hosts the Eclipse Luna projects and helps cultivate both an open source community and an ecosystem of complementary products and service.
- The independent not-for-profit corporation was created to allow a vendor neutral and open, transparent community to be established around Eclipse. Today, the Eclipse community consists of individuals and organizations from a cross section of the software industry.

Python

- Python is a programming language used on a server to create web applications.
- Python works on different platforms like Windows, Mac, Linux, and Raspberry Pi and has a simple syntax similar to the English language.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly brackets for this purpose.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.

Java 1.8

- The Java Development Kit (JDK) is a software development environment used for developing Java applications and applets. It includes the Java Runtime Environment (JRE), an interpreter/loader (java), a compiler(javac), an archiver (jar), a documentation generator (javadoc)and other tools needed in Java development.

- A Java virtual machine (JVM) is an abstract computing machine that enables a computer to run a Java program. There are three notions of the JVM: specification, implementation, and instance. The specification is a document that formally describes what is required of a JVM implementation. Having a single specification ensures all implementations are interoperable.

- A JVM implementation is a computer program that meets the requirements of the JVM specification. An instance of a JVM is an implementation running in a process that executes a computer program compiled into Java bytecode.

MySQL

- Java web application will require storing large amounts of metadata and keep data organized. Therefore, there was a need to host a Java web application with MySQL. It is the safest relational database currently in use making it ideal for e-commerce sites that handle frequent online transactions and other sensitive data.

- Mysql can be built to handle the most demanding websites with the heaviest traffic, it's not bogged down by high usage. Even when it's used by traffic-heavy sites like Twitter and Facebook, MySQL maintains its lightning fast performance speeds.

- A few other benefits of using MySQL as opposed to other database software are as follows:

- State-of-the-art security: MySQL's reputation as the safest relational database currently in use makes it ideal for e-commerce sites that handle frequent online transactions and other sensitive data.

- High-quality performance: Built to handle the most demanding websites with the heaviest traffic, it's not bogged down by high usage. Even when its used by traffic-

heavy sites like Twitter and Facebook, MySQL maintains its lightning fast performance speeds.

- More uptime: MySQL guarantees 100 percent uptime so that you never have to worry about surprise software crashes.
- Easy maintenance: Because its open-source, the software is constantly being upgraded and debugged, which means less maintenance for you to worry about, all you have to worry about is your Java site or web application.

3.6 NON-FUNCTIONAL REQUIREMENTS

3.6.1 Performance Requirements

High Speed

System should process requested task in parallel for various action to give quick response then system must wait for process completion.

Accuracy

System should correctly execute process, display the result accordingly. System output should be in user required format.

Interoperability

System should have the ability to exchange information and communicate with internal and external applications and systems. It must be able exchange information both internally and externally.

Response Time:

The response time of the system should be deterministic at all times and very low, i.e. it should meet every deadline. Thus, the system will work in real time.

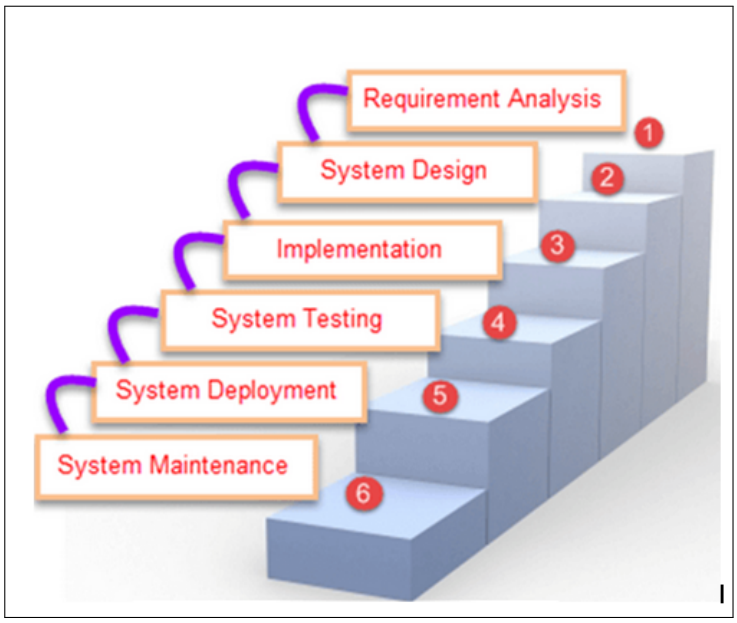

Figure 3.1: Iterative SDLC Model

3.7 ANALYSIS MODELS: SDLC MODEL TO BE APPLIED

Iterative SDLC Model

The development process starts with the requirements to the functional part, which can be expanded later. The process is repetitive, allowing to make new versions of the product for every cycle. Every iteration includes the development of a separate component of the system, and after that, this component is added to the functions developed earlier. The major steps of the SDLC model are given below:

- **Requirement gathering:** All the functional and non-functional requirements of the project were identified. Interaction with the users and all other stakeholders of the project was conducted to identify all the requirements starting from important features like maintaining audit trail, security parameters etc. to the very basic features like the look and the feel of user interface. The different requirements mainly fall into categories:

    1. System features
    2. Security parameters
    3. User requirements
    4. Administrator requirements

5. User interface

- **Design:** The first step was database design. A complete database required for the implementation of this project was designed. The second step was project design. The project was designed based on a framework. The framework uses three layers:
  a. Business entities layer: It identifies all the entities used in the project.
  b. Business logic layer: This layer operates on the business entity to achieve the goals.
  c. Data access layer: This layer serves as an interface between backend and the services.

- **Construction:** All modules and user interface were built in this step. Development was done using Java. Database was constructed in MySQL.

- **Integration and system testing:** All the modules were integrated together. The user interface was integrated with the modules which made the use web services. Data flow originated from the database built in MySQL. In testing phase project was tested and debugged. Various test cases were developed and the project was tested at the developers end as well as users end. Debugging was done to discover errors and exception which were corrected.

- **Installation and maintenance:** Our system is installed on one dedicated machine and it is accessible to admin and all authenticated users. Maintenance of our system is done on regular basis. New requirements and features can be added as and when required as long as they do not conflict with the existing features

Chapter 4

SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

We have implemented a system that can analyze and differentiate pathological voice from normal voice using data mining technique. Here to detect vocal fold pathology at an early stage from set of features like MFCC and ZCR which are vocal tract parameters used. Below figure shows the architecture of the proposed system.

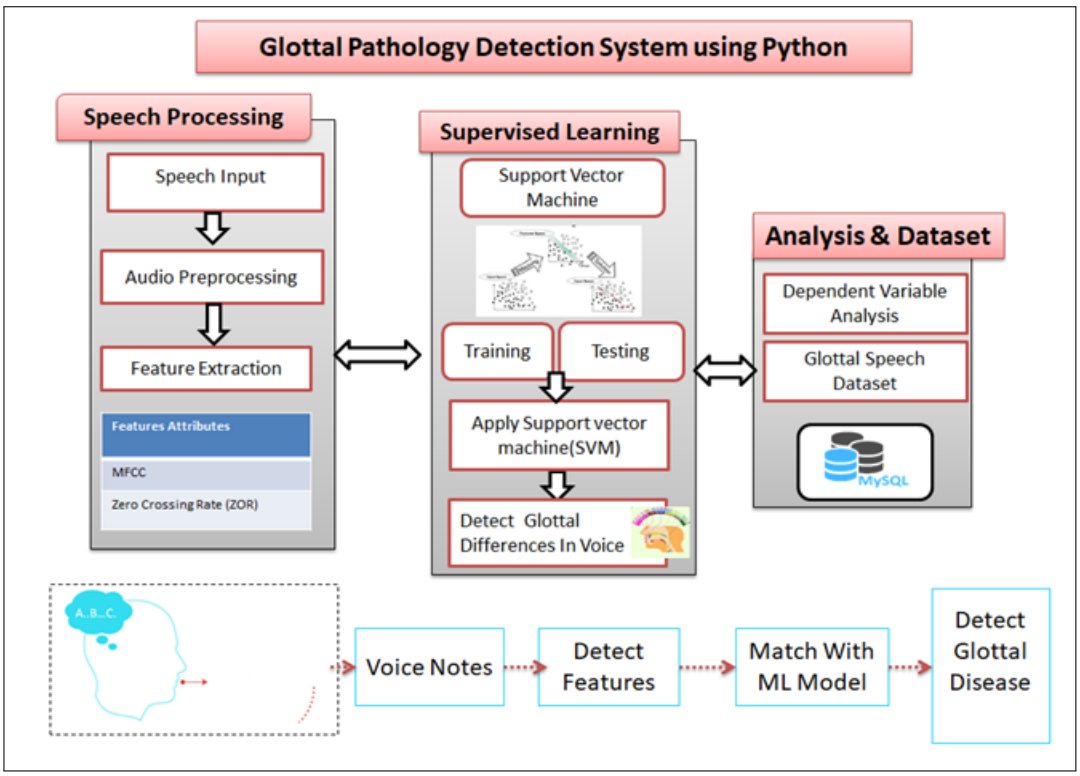

Figure 4.1: System Architecture

- **Speech Input:**

  The main aim of the proposed system is the extraction of glottal parameters from speech signal for the distinction between glottal pathological voice and normal voice. Firstly, input speech data is given to the system which contain database of patients suffering from glottal and supra-glottal cancer and from normal persons.

- **Preprocessing:**

  The speech data contain lot of noise. By using noise removal techniques or removing any other disturbances present in the data, it is preprocessed to get the fine-tuned data.

- **Feature Extraction:**

  The system can detect vocal fold pathology at an early stage from set of features like MFCC and ZCR, which are vocal tract parameters, in detection of the glottal pathologies.

  a. MFCC (Mel-Frequency Cepstrum (MFC))

  The first step in any automatic speech recognition system is to extract features i.e. identify the components of the audio signal that are good for identifying the linguistic content and discarding all the other stuff which carries information like background noise, emotion.

  The Mel-Frequency Cepstral Coefficients (MFCC) features are the most commonly used features in speaker recognition. It combines the advantages of the cepstrum analysis with a perceptual frequency scale based on critical bands. MFCC is based on Human hearing perceptions which cannot perceive frequencies over 1 khz. In other words, MFCC is based on known variation of the human ear's critical bandwidth with frequency.

  b. Zero Cross Rate (ZCR):

  The zero-crossing rate is the rate of sign-changes along a signal, i.e., the rate at which the signal changes from positive to zero to negative or from negative to zero to positive. The zero-crossing rate is strongly correlated with the spectral centroid, which can be computed using the MFCC spectrum, and is a measure for the high frequency content of a signal.

- **Classification:**

  Once the features extracted and normalized, we train a Support Vector Machine model.

According to preliminary tests, the best results were achieved. SVM classifier is used for various feature combinations to classify the glottal pathologic voice from normal voice.

4.2 MATHEMATICAL MODEL

Mathematical Model:

Let us consider S be a Systems such that

S= {U,FT,P S,T,Ss,Ds}, where

- U= {U1, U2, U3.....Un | 'U' is a Set of all Voters }
  There may be number of users for making use of system. So this is the Infinite Set.
- I = {I | is the speech input of the system }
- FT= {F1, F2, F3.... Fn | F are the features extracted }
  FT is the feature extracted from speech input.
- T={ T is the technique used to process Input voice }
  P is the technique used to pre-process the input voice.
  T is the technique used for detecting glottal differences from voice.
- SS = {S_REG, S_LOGIN, S_Feature Data| SS is a Set of Storage Service }
  STORAGE SERVER will provide four services like Registration, Login, and glottal parameters As this set also has finite attributes, so this is also Finite Set.
- DS = {Train_Data| DS is a Set of data table for permanent storing of data on server }

EVENTS and ACTIVITIES:

- **EVENT 1**
  User will make registration on SYSTEM & Storage Server.
  Let f(U) be a function of User
  Thus, f(U)$\rightarrow$ {Ss}
- **EVENT 2**
  Speech Input is given to the system.

Let f(U) be a function of User.

Thus, f(U) $\rightarrow$ {I1,I2,I3.....In } ε S

- **EVENT 3**

  Pre Processing is performed on Input to remove noise.

  Let f(P) be an image processing function.

  Thus, f(S) $\rightarrow$ {P $\cup$ I}ε P

- **EVENT 4**

  Let f(FT) be the feature extraction function of the system.

  Let f(S) is a function of the System

  Thus, f(S) = {FT1, FT2, FT3..... FTn } $\rightarrow$ I

- **EVENT 5**

  Let f(E) be a function used to classify glottal pathological voice using SVM classifier.

  Let f(R) be a Prediction function

  Thus, f(S) $\rightarrow$ {E} ε T

4.3 DATA FLOW DIAGRAMS

A data flow diagram (DFD) is a graphical representation of the flow of data through an information system, modeling its process aspects. It shows data is processed by a system in terms of inputs and outputs.

4.3.1 DFD Level-0

It only contains one process node (Process 0) that generalizes the function of the entire system in relationship to external entities.

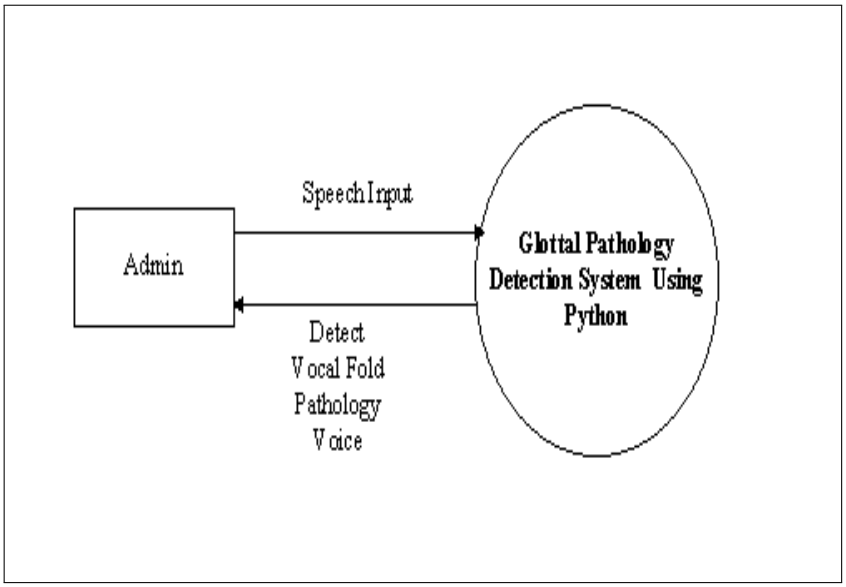

Figure 4.2: DFD Level-0

4.3.2 DFD Level-1

DFD level 1 diagram expands the DFD 0 and shows the detailed flow of the proposed system.

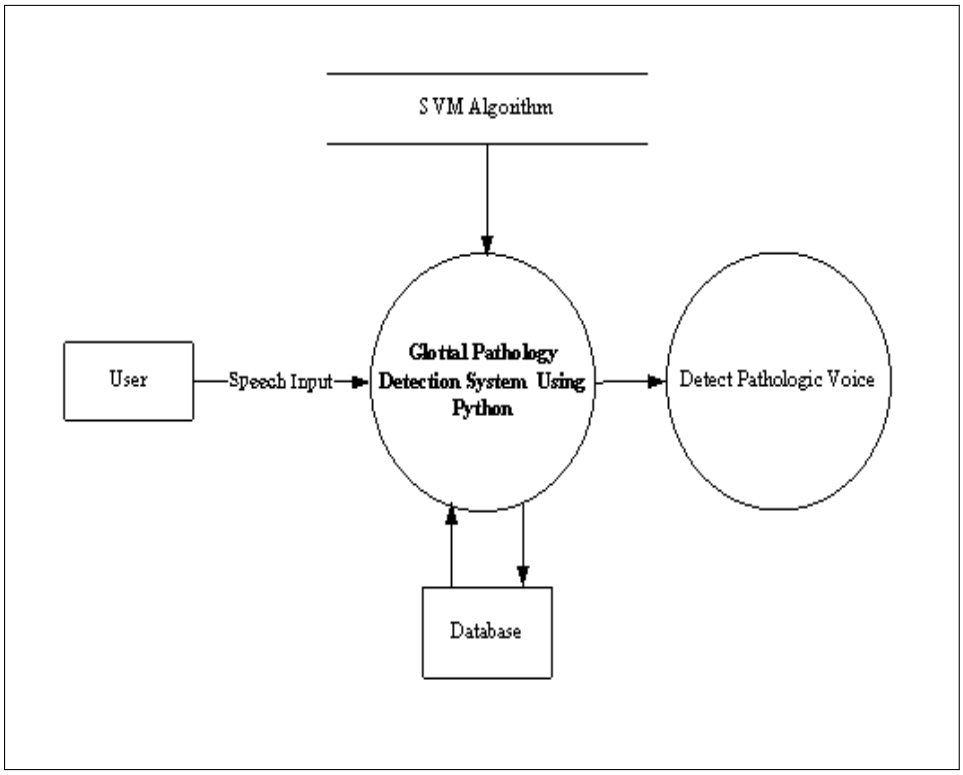

Figure 4.3: DFD Level-1

4.3.3 DFD Level-2

DFD level 2 diagram expands the DFD 1 and shows the detailed flow in the proposed system. It shows the different processes that take place to perform the authentication.

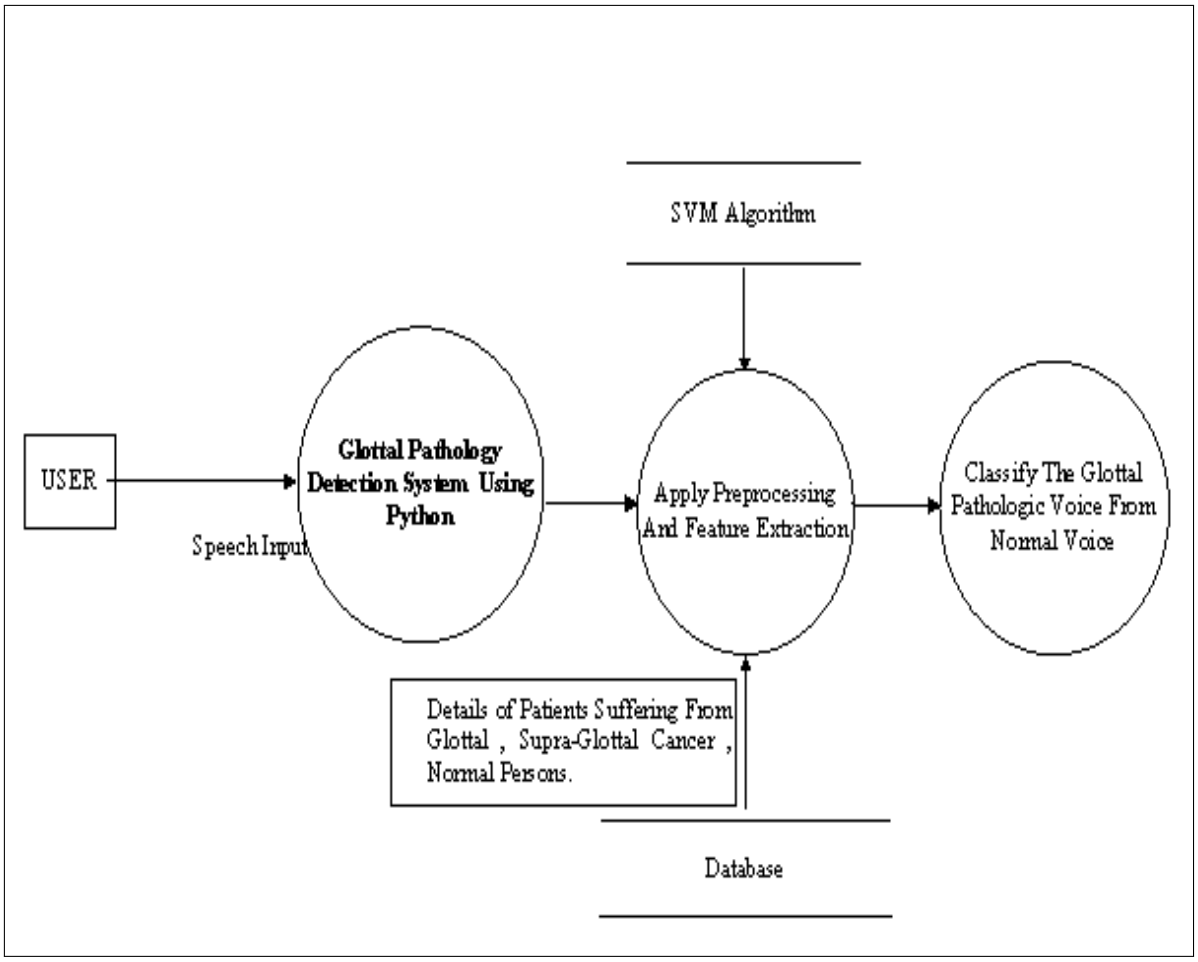

Figure 4.4: DFD Level-2

4.4 ENTITY RELATIONSHIP DIAGRAMS

Data objects and their major attributes and relationships among data objects are described using an ER - like form. ER diagram is a data model for describing the data or information aspects of a software system. The main components of ER models are entities and the relationships that exist among them. The various entities are system and admin.

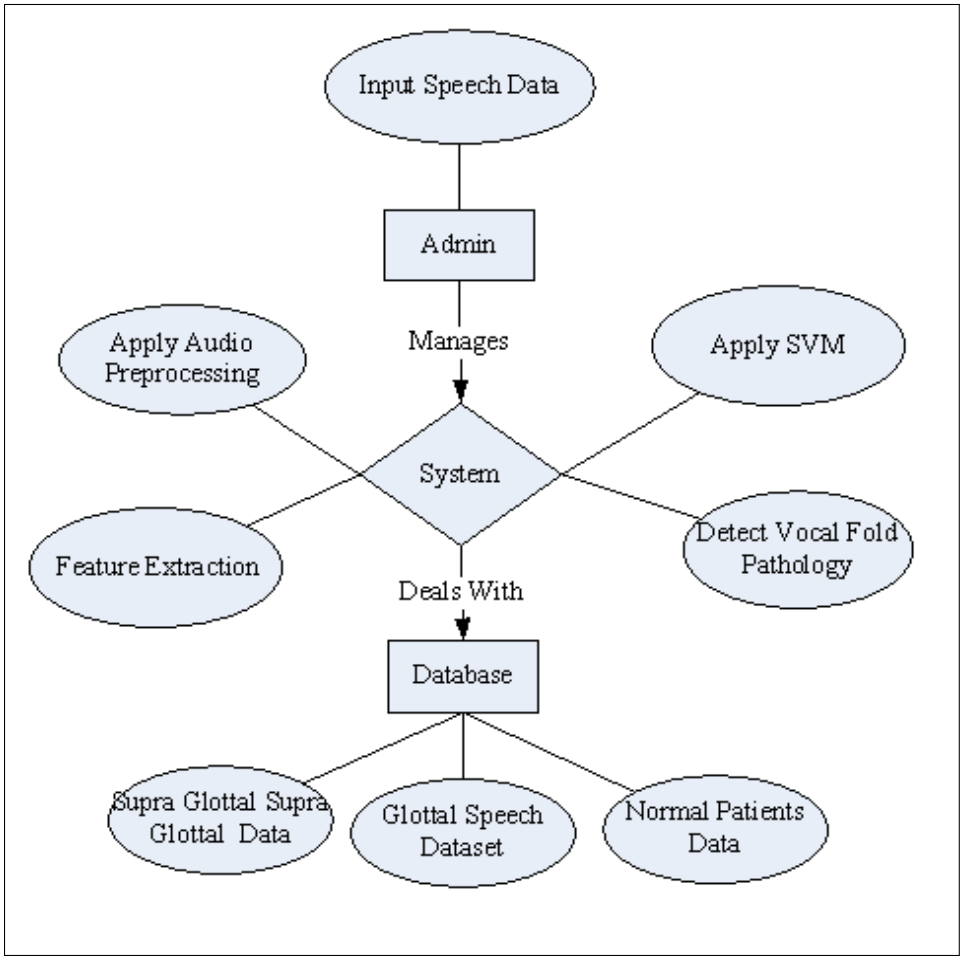

Figure 4.5: ER Diagram

4.5 UML DIAGRAMS

4.5.1 Use Case Diagram

Use case diagram is a simple representation of a users interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. Here the actors are: The voter and the system.

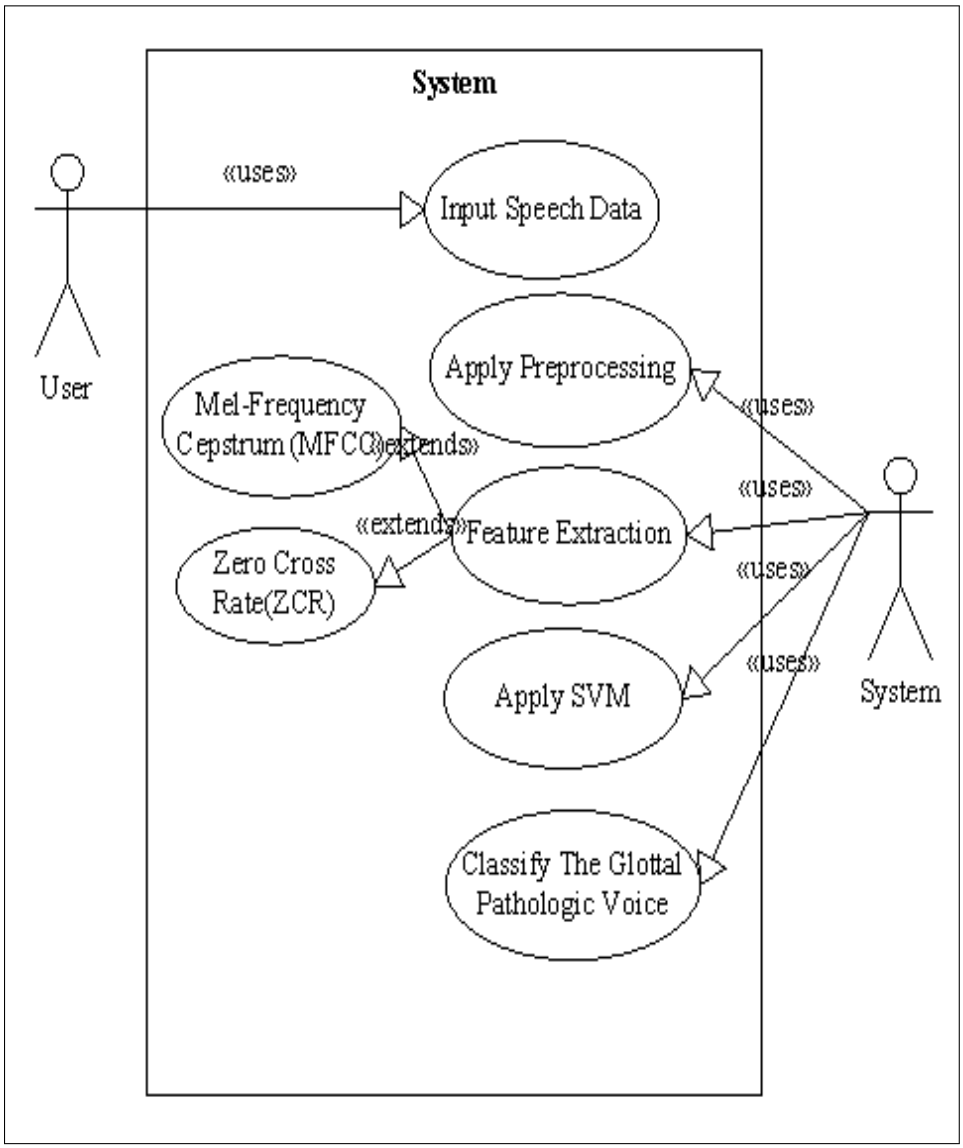

Figure 4.6: Use Case Diagram

4.5.2 Sequence Diagram

A Sequence diagram is an interaction diagram that shows how processes operate with one another and in what order. Sequence diagrams are sometimes called event diagrams or event scenarios. The sequence diagram for the proposed system shows the interaction in between admin, system and database.

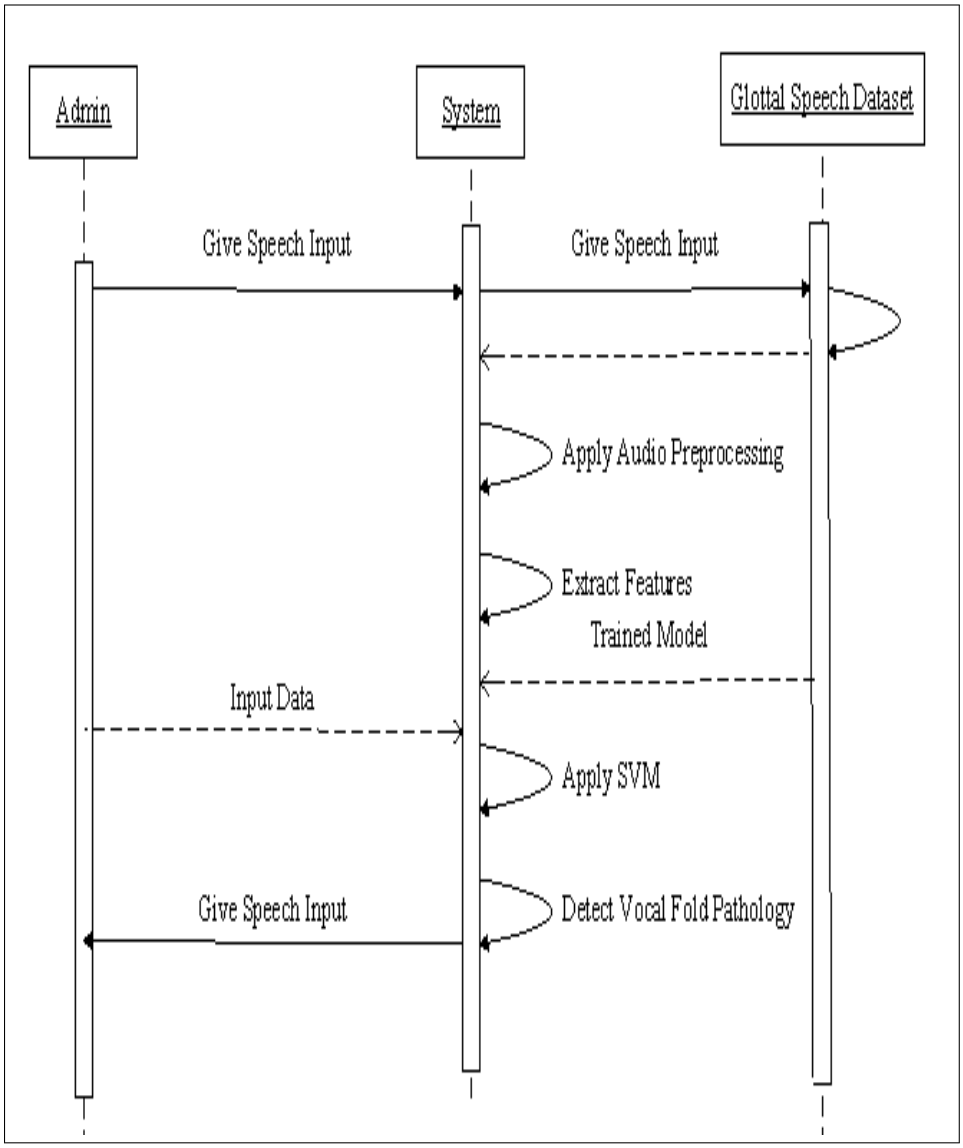

Figure 4.7: Sequence Diagram

4.5.3 Class Diagram

Class diagram is a type of structure diagram that shows the structure of the classes, attributes, operations and relationship among them. Given below is the class diagram of the proposed system which shows 6 classes such as user, system, database.

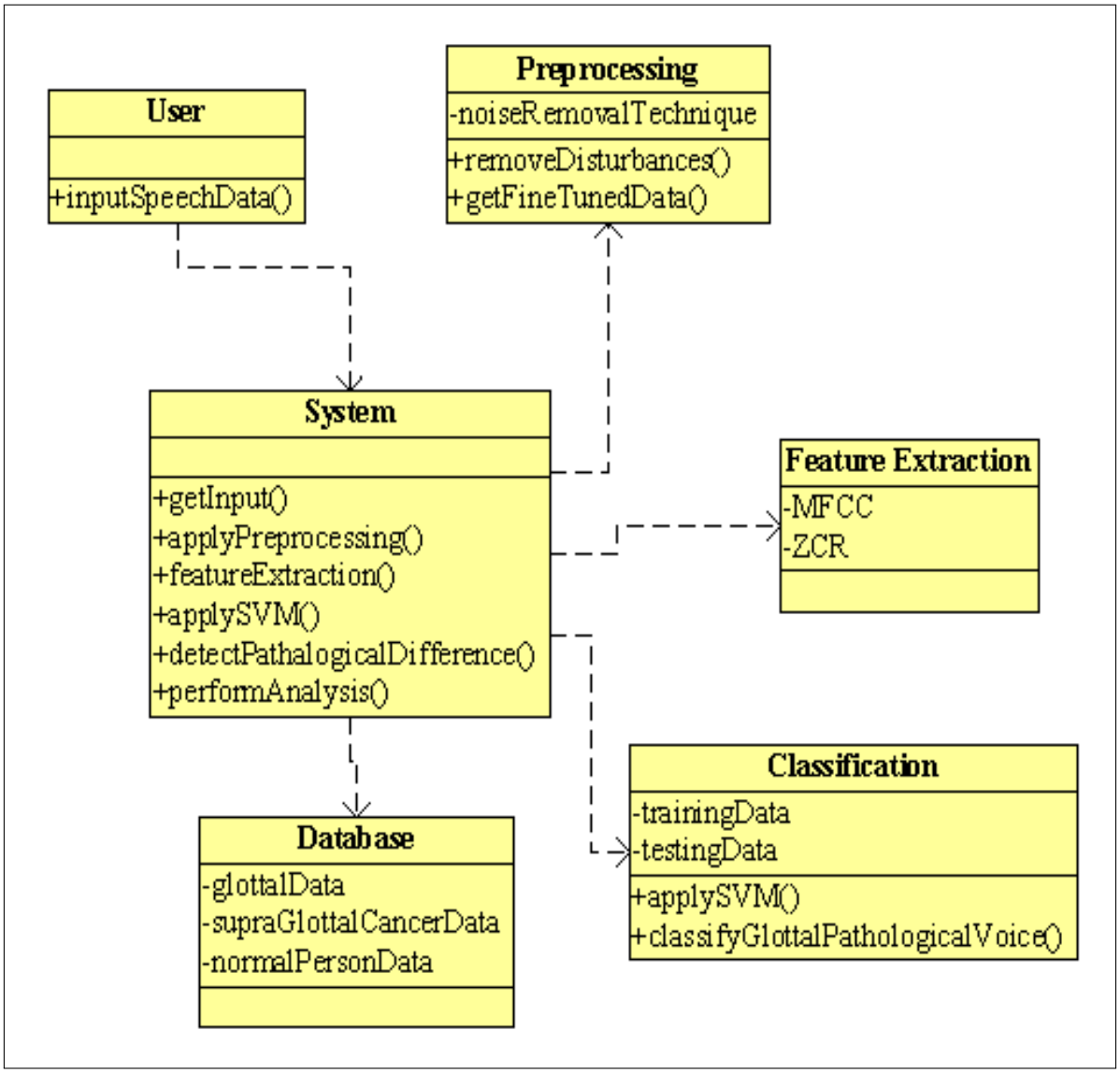

Figure 4.8: Class Diagram

4.5.4 Component Diagram

A component diagram depicts how components are wired together to form larger components and or software systems. A component is something required to execute a stereotype function. Examples of stereotypes in components include executable, documents, database, tables, files.

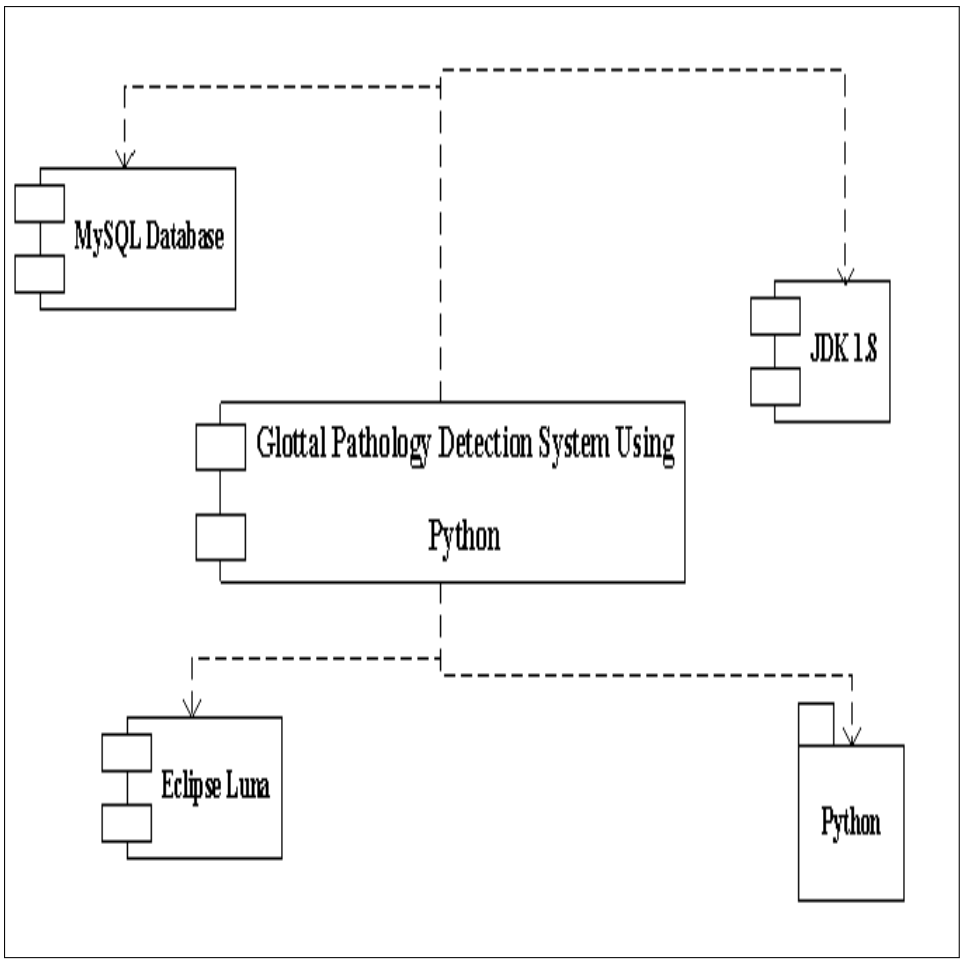

Figure 4.9: Component Diagram

4.5.5 Deployment Diagram

Deployment diagrams are used to visualize the topology of the physical components of a system where the software components are deployed. The deployment diagram for the proposed system is shown below. It shows the physical or the hardware components on which the software components run. The physical components include the Server, Client, Windows JVM and the Database.

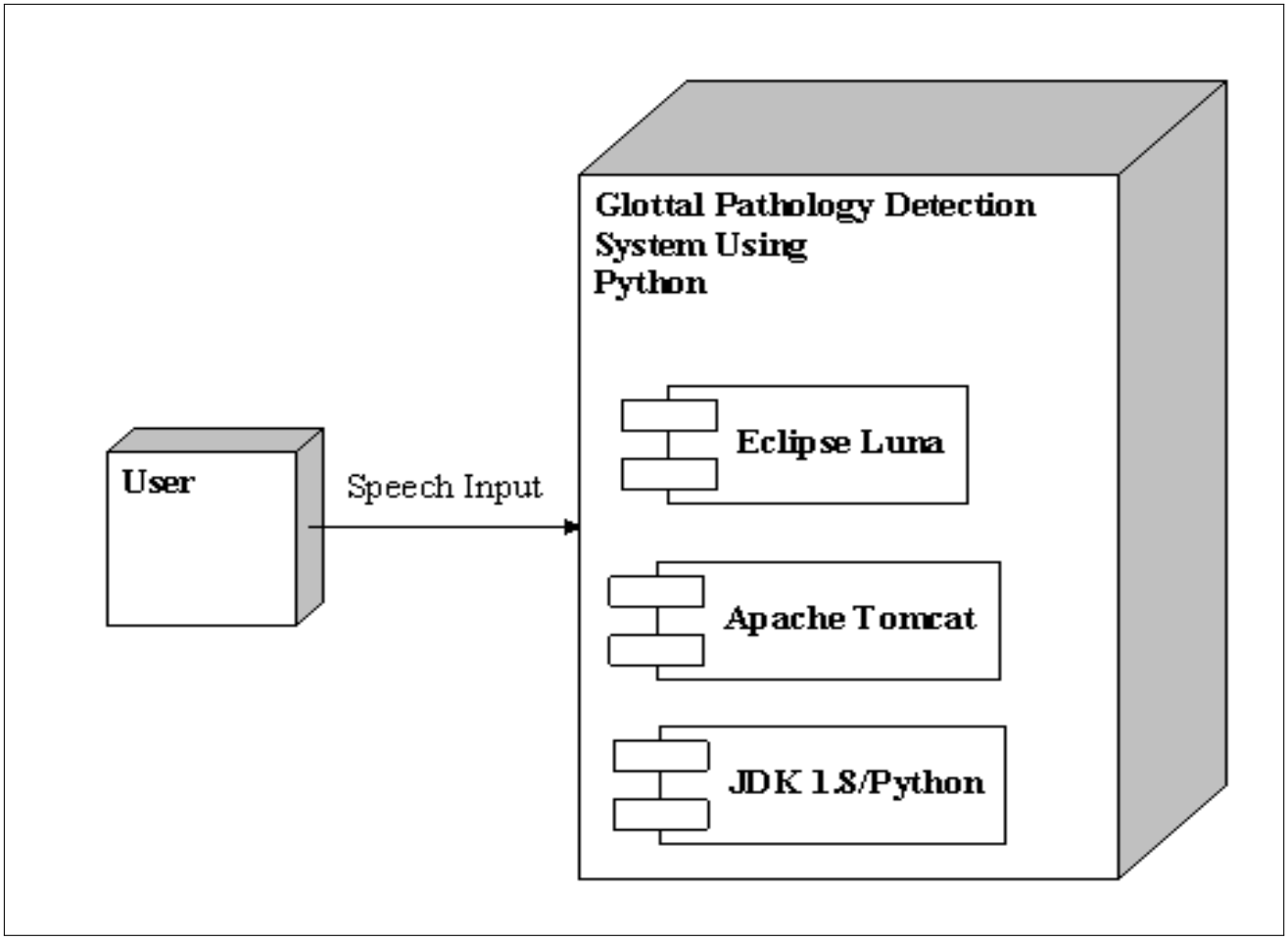

Figure 4.10: Deployment Diagram

4.5.6 State Diagram

State diagrams are used to give an abstract description of the behavior of a system. This behavior is analyzed and represented by a series of events that can occur in one or more possible states. Hereby “each diagram usually represents objects of a single class and track the different states of its objects through the system”.

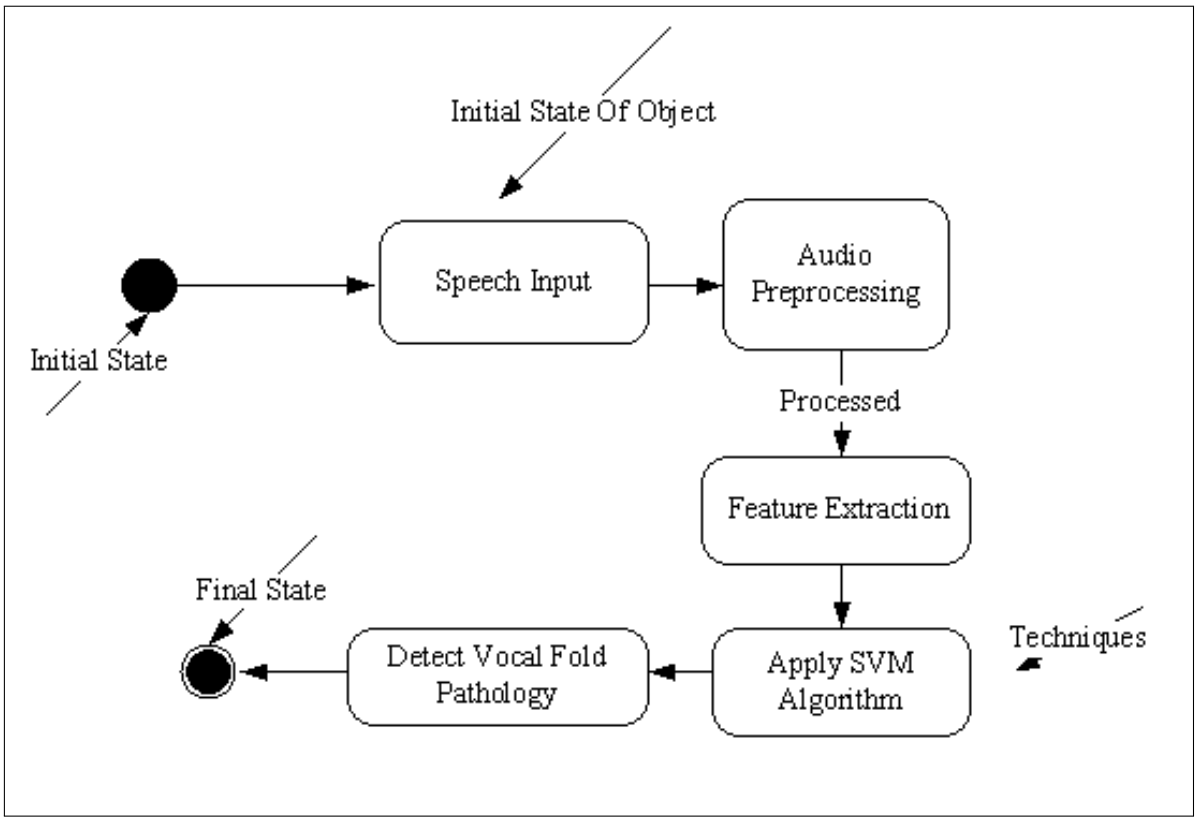

Figure 4.11: State Diagram

4.5.7 Activity Diagram

Activity diagrams are graphical representations of workflows of step-wise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.

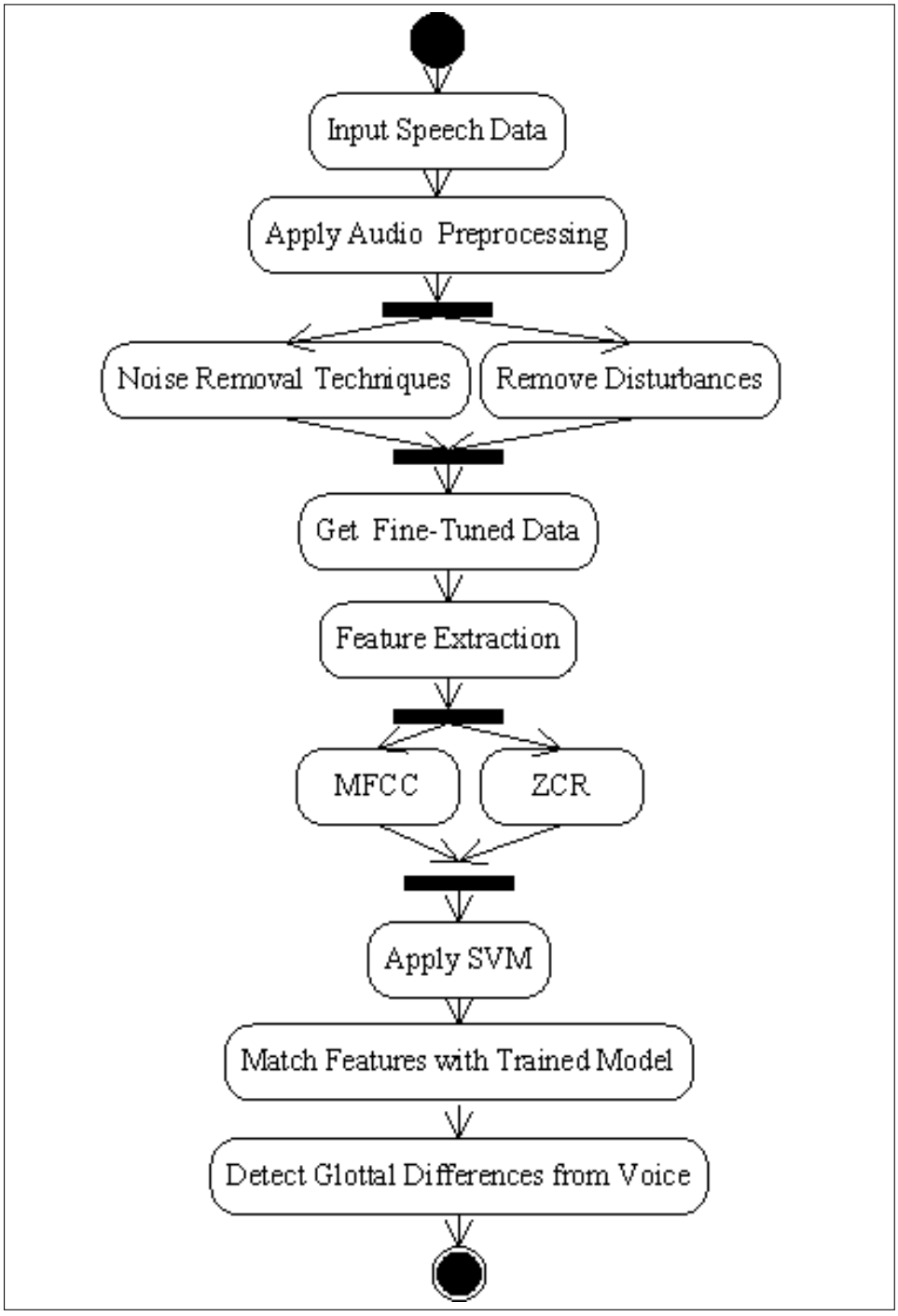

Figure 4.12: Activity Diagram

Chapter 5

PROJECT PLAN

5.1 PROJECT ESTIMATE

Use of Waterfall model and associated streams derived for estimation.

5.1.1 Reconciled Estimates

Cost Estimate

The initial cost estimate of the project before beginning the implementation process is INR 15000 for in-house resources. This cost may vary. This estimate is subject to change according to the availability and/or need of a particular item.

Time Estimates

The initial time estimate for the complete implementation of the primary objectives is 45-50 days depending on the schedule of the developers. The secondary objectives require an additional 25 days to be completed. Also, depending on the stage of development, the testing and debugging would require an additional of 15 days.

5.1.2 Project Resources

Hardware

- **Processor:** 1 gigahertz (GHz) or faster processor or SoC.
- **RAM:** 1 gigabyte (GB) for 32-bit or 2 GB for 64-bit.

- **Hard disk space:** 16 GB for 32-bit OS 20 GB for 64-bit OS.

Software

- **IDE:** Eclipse Luna
- **Platform:** Microsoft Windows 7 Professional or greater
- **Language:** Java 1.8, Python

5.2 RISK MANAGEMENT

During different phases of project there can be several threats like uncertainty in financial markets, threats from project failures. These threats if not attended can later cause poor performance and project failure. By knowing predictable risks, the project manager takes first step towards avoiding them and controlling them when necessary. Two types of different risks are:
Generic risks: Generic risks are a potential threat to every software project.
Product-specific risks: Product-specific risks can be identified only by those with a clear understanding of the technology, the people, and the environment that is specific to the project at hand.

5.2.1 Risk Identification

It is the process of determining potential threats which can later harm the performance of the project. One method of identifying risks is to create a risk item checklist. The checklist can be used for risk identification and focuses on some subset of known and predictable risks in the following generic subcategories:

- Product size: Risks associated with the overall size of the software to be built or modified.
- Business Impact: Risks associated with constraints imposed by management or the market place.
- Customer characteristics: Risks associated with the sophistication of the customer and the developer's ability to communicate with customer in a timely manner.

- Process definition: Risks associated with the degree to which the software process has been defined and is followed by the development organization.
- Development Environment: Risks associated with the ability and quality of the tools to be used to build the product.
- Technology to be built: Risks associated with the complexity of the system to be built.
- Staff size and experience: Risks associated with the overall technical and project experience of the software engineers who will do the work.

5.2.2 Risk Analysis

Risk management is concerned with identifying risks and drawing up plans to minimize their effect on a project. A risk must also have a probability. It must be a chance to happen or it is not a risk. The risks for project can be analyzed within the constraint of time and quality.

ID	Risk Description	Risk Description	Impact
1	Privacy	Low	–
2	Connectivity failure	Low	— .

5.2.3 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

Risk ID	1
Risk Description	Third party access
Category	Networking Environment
Source	Internet.
Probability	High
Impact	High
Response	Mitigate
Strategy	Break security
Risk Status	Occurred

Risk ID	2
Risk Description	User can make fake profile
Category	Requirements
Source	Software Design Specification documentation review.
Probability	Low
Impact	High
Response	Mitigate
Strategy	Better testing will resolve this issue.
Risk Status	Identified

Risk ID	3
Risk Description	Server crash
Category	Technology
Source	This was identified during early development and testing.
Probability	Low
Impact	Very High
Response	Accept
Strategy	Example Running Service Registry behind proxy balancer
Risk Status	Identified

5.3 PROJECT SCHEDULE

5.3.1 Project Task Set

Major Tasks in the Project stages are:

- Task 1: Requirement Gathering
- Task 2 : Literature Survey
- Task 3 : Applications and Objectives
- Task 4 : Platform/Technology Selection
- Task 5 : Internal Presentation - 1
- Task 6 : Study of Algorithms
- Task 7 : Mathematical Model
- Task 8 : Software Requirements Specification
- Task 9 : UML Diagrams
- Task 10 : Problem Definition using NP Hard/ NP Complete
- Task 11 : System Architecture
- Task 12 : Testing phase
- Task 13 : Internal Presentation - 2
- Task 14 : Report Preparation
- Task 15 : Installation
- Task 16 : Overview of Project Model
- Task 17 : Construction of GUI
- Task 18 : Module Identification
- Task 19 : Module 1 - User Authentication

- Task 20 : Module 2 - Database generation
- Task 21 : Module 3 - Connection of GUI to Database
- Task 22 : Module 4 - Testing and Result
- Task 23 : Test Planning
- Task 24 : Testing
- Task 25 : Poster Presentation
- Task 26 : Research of Journals for Final Report

5.3.2 Task Network

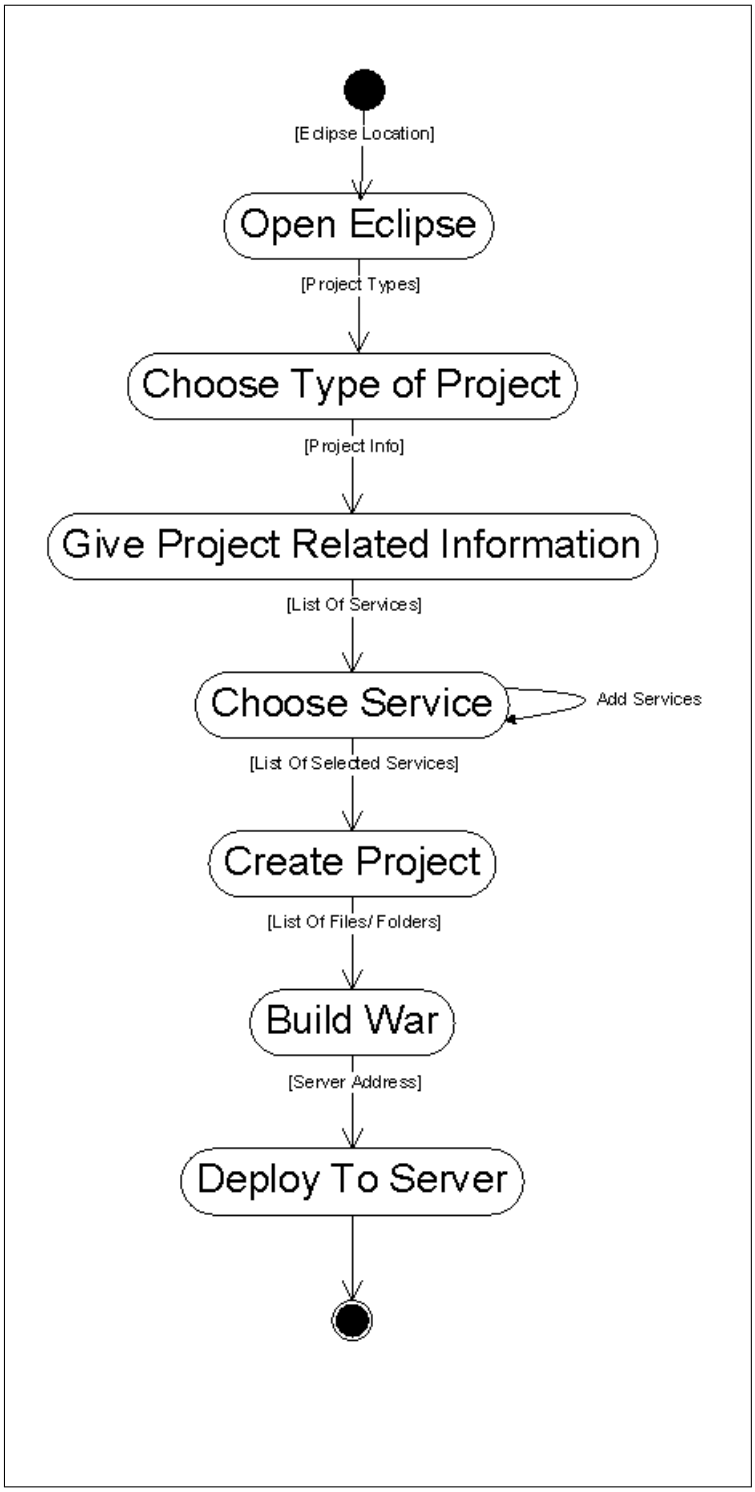

Figure 5.1: Task Network Diagram

5.3.3 Timeline Chart

Project planning is part of project management, which relates to the use of schedules such as Gantt charts to plan and subsequently report progress within the project environment. A project management plan is the planning document, capturing the entire project end-to-end, covering all project phases, from initiation through planning, execution and closure.
Analysis or prototyping should increase in direct proportion with project size and complexity. 20 to 25 % of effort is normally applied to software design.
1. Requirement gathering
2. Literature Survey of existing systems
3. Requirement Modeling and training
4. Development of mock screens
5. Actual Implementation
The Gantt chart for the project is drawn below. The Gantt chart shows the project planning right from the beginning when the topic was finalized.
It depicts the software development life cycle (SDLC). The milestones in the project include Topic Selection, Requirements Gathering, Software Requirements Specification, Hardware Requirements Specification.
The milestones also depict the project planning stage. In the Gantt chart below the milestones are represented according to the months in the development lifetime.

5.4 TEAM ORGANIZATION

The team for B.E. final year project consists of a team of college students, a college professor as an internal guide and industry professionals as external guide making collaborative efforts for fulfillment and implementation of project problem statement.

5.4.1 Team Structure

Each and every member of the team is responsible for the identification of problems, proposing problem solving methodologies, identifying approaches for implementation and documentation.

Prof P K Kosamkar is the internal college guide for providing thorough domain guidance, doubt removal and suggesting approaches and ensuring timely completion of activities.

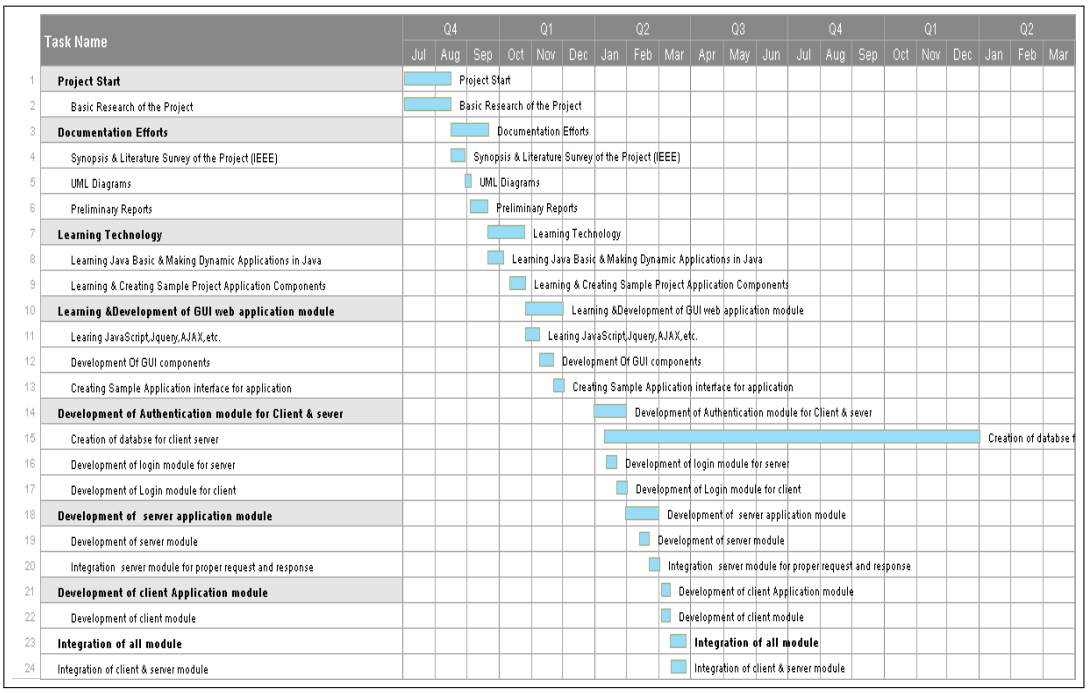

Figure 5.2: Timeline Chart

Sr. No.	Member	Responsibilities
1	ANUJ CHAVAN	Project analysis, Developer and Design
2	FEBIN AHMED	Requirement Gathering and Developer
3	MANMOHAN CHAKOLE	Requirement Gathering and Developer
4	PRANAV JADHAV	Testing and Design

5.4.2 Management Reporting and Communication

We report the progress of our project to our internal guide twice a week. We show our weekly status to our guide and incorporate the necessary changes. We communicate among ourselves in case we want suggestions while executing our tasks.

Chapter 6

PROJECT IMPLEMENTATION

6.1 OVERVIEW OF PROJECT MODULES

The voice disorders are caused due to defects in speech organs, mental illness, hearing impairment, autism, paralysis or multiple disabilities. Vocal cord lesions affect the vocal system and lead to voice disorders which are intensely affecting a large number of people. The main concern of this system is to detect vocal fold pathology at an early stage from some set of features. The project modules involved in the system is mentioned below:

1. Speech Input
2. Preprocessing
3. Feature Extraction

a. MFCC (Mel-Frequency Cepstrum (MFC))

b. Zero Cross Rate (ZCR)

4. Voice Classification

6.2 TOOLS AND TECHNOLOGIES USED

JDK 1.8:

- The Java Development Kit (JDK) is a software development environment used for developing Java applications and applets.
- It includes the Java Run- time Environment (JRE), an interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator (javadoc) and other tools

needed in Java development.

Databases

- The database basically used for user storing user details like Username and Password. The tool used for database functionalities was MySQL GUI Browser.

Python

- Python is a programming language used on a server to create web applications. Python has syntax that allows developers to write programs with fewer lines than some other programming languages.

6.3 ALGORITHM DETAILS

6.3.1 Support Vector Machine

- SVM is a powerful classifier that is able to distinguish two classes. SVM classifies the test image in to the class with highest distance up to the neighboring point in the training.

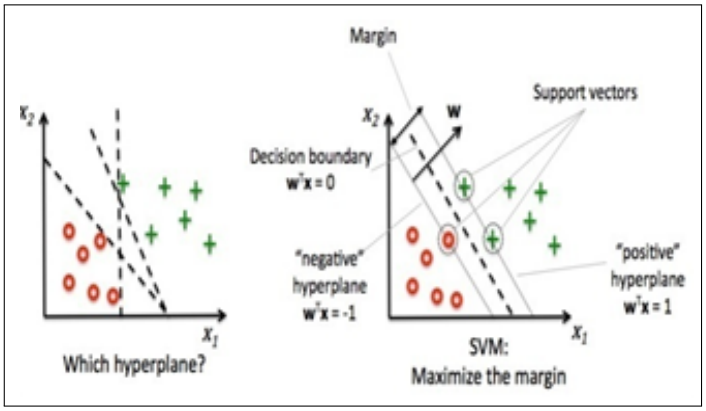

Figure 6.1: Distinguishing Hyper Plane to Minimize the Error

- SVM training algorithm built a model that predicts whether the test image falls into this class or another.

- SVM necessitate a vast training data to decide a decision boundary and computing cost is very high although we are using single pose (frontal) detection.

- The SVM is a learning algorithm for classification which attempts to discover the finest distinguishing hyper plane which minimize the error for unseen patterns.
- For the data which cannot be distinguished, the input is mapped to high-dimensional attribute space where they can be separated by a hyper plane. This projection is well performed by means of kernels.

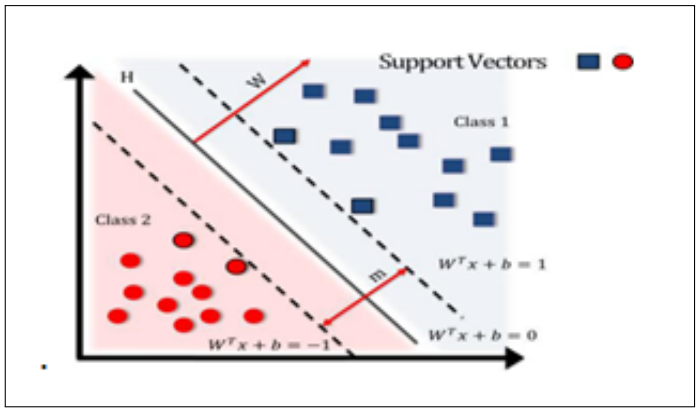

Figure 6.2: Separating Hyper Plane by Equation

Chapter 7

SOFTWARE TESTING

Software testing is an activity aimed at evaluating an attribute or capability of a program or system and determining whether it meets its required results. It is more than just running a program with the intention of finding faults. Every project is new with different parameters. No single yardstick maybe applicable in all circumstances. This is a unique and critical area with altogether different problems. Although critical to software quality and is widely deployed by programmers and testers. Software testing still remains an art, due to limited understanding of principles of software. The difficulty stems from complexity of software. The purpose of software testing can be quality assurance, verification and validation or reliability estimation. Testing can be used as a generic metric as well. Software testing is a trade-off between budget, time and quality.

7.1 TYPE OF TESTING

- **Integration testing**

  Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successful unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

- **System Test**

  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 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. 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.

- **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.

7.2 TEST CASES & TEST RESULTS

Test ID	**Test Case Title**	**Testcase Des**	**Expected Result**	**Actual Result**
T01	To verify Login Name and Password	Enter login name and password	If user name and Password are Correct then message "Login successful and redirected to the next page."	Login Successful

	-	Enter Invalid login name and Password	If username or password are incorrect or any one of them is blank then show error message as “Please enter correct Password”	Display Error Message
T02	Speech Input	Speech Data is Given as Input to the system	System will Process the Data	Same as expected
T03	Preprocessing	Apply Noise Removal Techniques on speech Data	Get fine Tuned Data	Same as expected
T04	Feature Extraction	The Glottal Pathological Features are Extracted form Input	MFCC and ZCR Features are Extracted	Same as expected
T05	Classification	Apply Support Vector Machine	Apply Support Vector Machine	Same as expected

Table 7.1

Chapter 8

RESULTS

8.1 OUTCOMES

The proposed system can help to detect glottal pathologies from the voice input. The system uses MFCC which are vocal tract parameters, in the detection of the glottal pathologies and ZCR for the detection. The SVM classifier is used to train the data. The classification of pathologies is carried out with the help of the SVM classification algorithm.
In our proposed system we used a large Saarbruecken Voice Database. Where accuracy and precision are calculated based on false positives images, i.e. which are items incorrectly labelled as belonging to the class and false negatives, which are items which were not labelled as belonging to the positive class but should have been.
1. TP: Positive samples classified as positive.
2. TN: Negative samples classified as negative.
3. FP: Negative samples classified as positive.
4. FN: Positive samples classified as negative

For the mentioned classes the accuracy and precision are calculated by using the formula. The precision is the percentage of documents that are correctly classified as positive out of all the documents that are classified as positive. Where, TP, FP, and FN are truly positive, false positive and false negative images.

$$\boxed{Precision = \frac{TP}{(TP+FP)}}$$

$$\boxed{Recall = \frac{TP}{(TP+FN)}}$$

Confusion Matrix

	Class a	Class b	Class c	
Class a	27	1	0	96.42%
Class b	4	24	0	85.71%
Class c	1	0	27	96.42%

Figure 8.1: Confusion Matrix

The precision obtained by using above for the classes mentioned above is given in below table.

Class Label	Precision
a	0.84
b	0.96
c	1

Figure 8.2: Precision Table

Recall Calculation

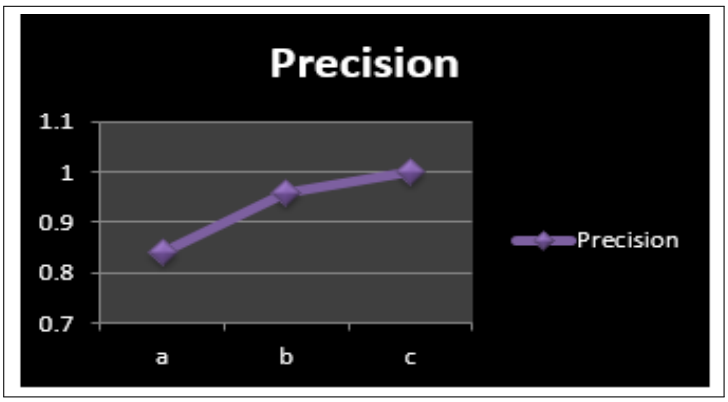

Figure 8.3: Plot for Precision

Class Label	Recall
a	0.96
b	0.85
c	0.96

Figure 8.4: Table for Recall

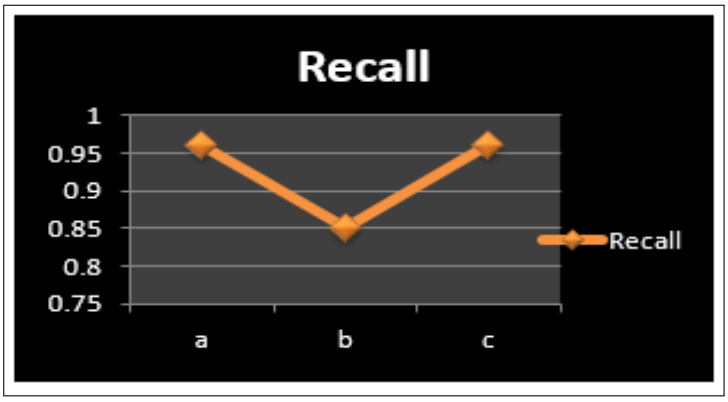

Figure 8.5: Plot for Recall

The overall accuracy obtained with the help of the confusion matrix is 92.8571%, where out of 84 i.e. total number of instances. The correctly classified instances are 78 and incorrectly classified are 6. We get the 92% accuracy by using SVM with features like MFCC and ZCR.

8.2 SCREEN SHOTS

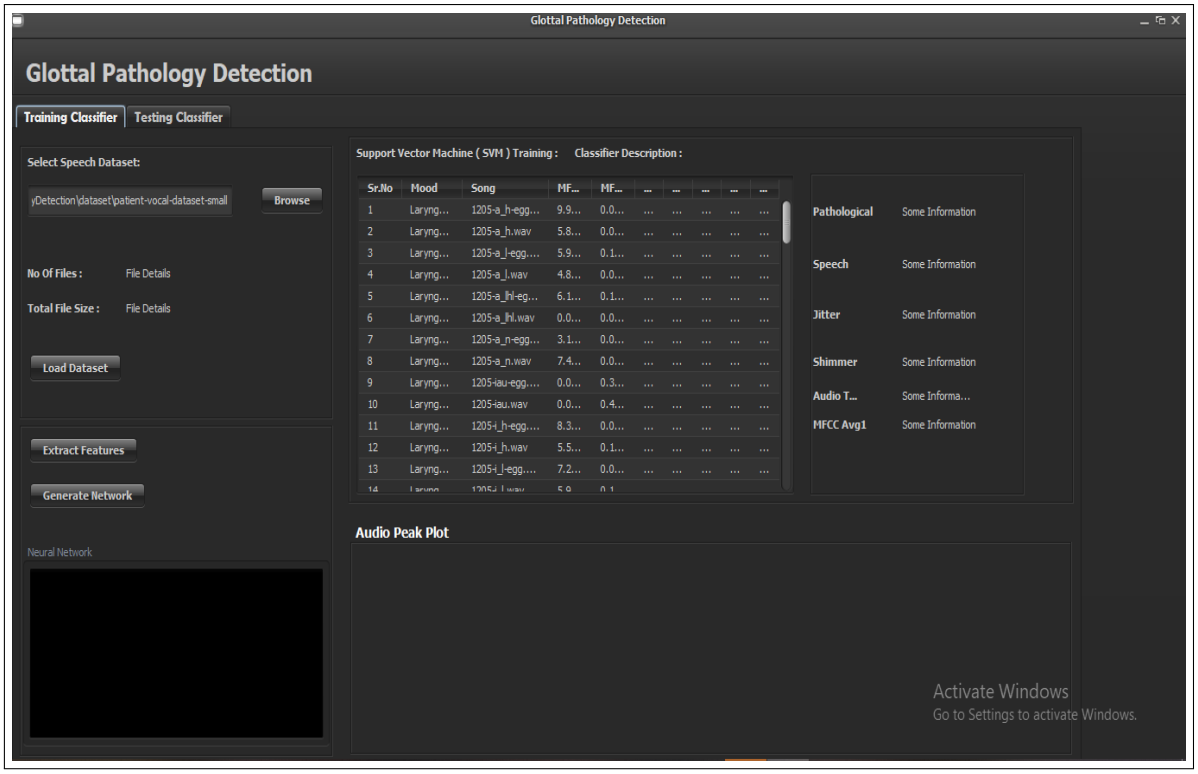

Figure 8.6: Training

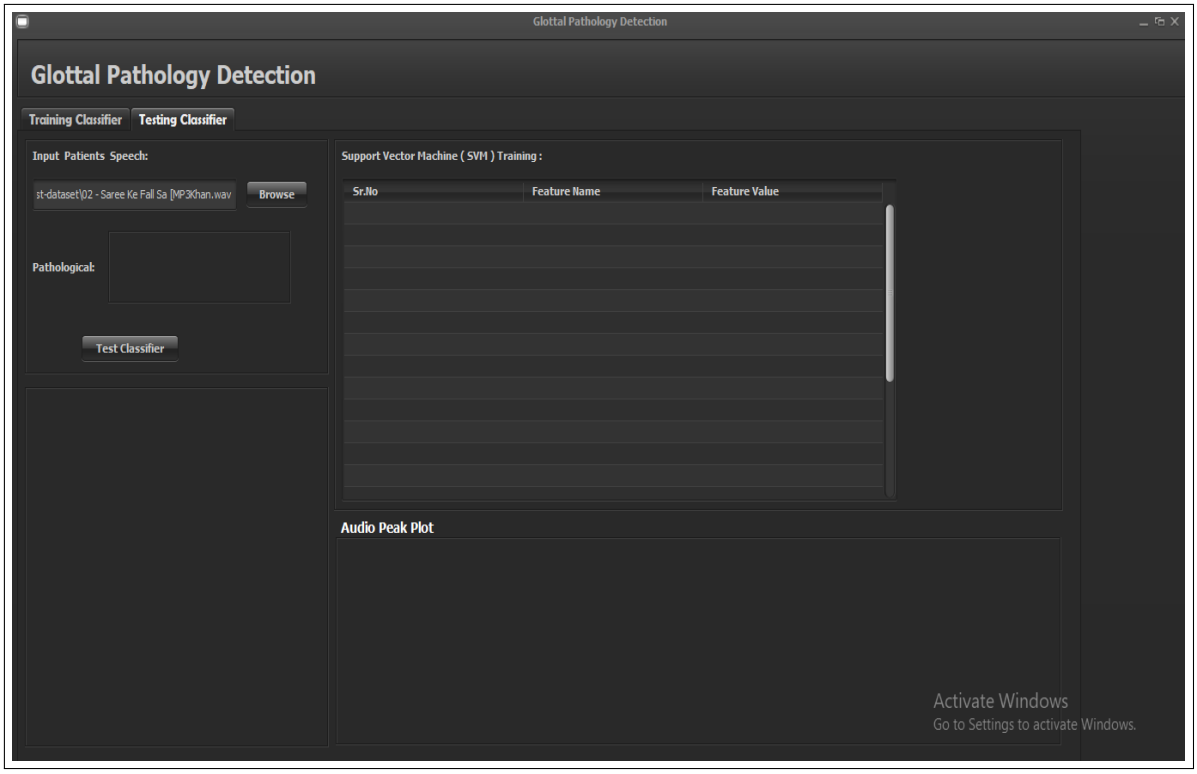

Figure 8.7: Testing

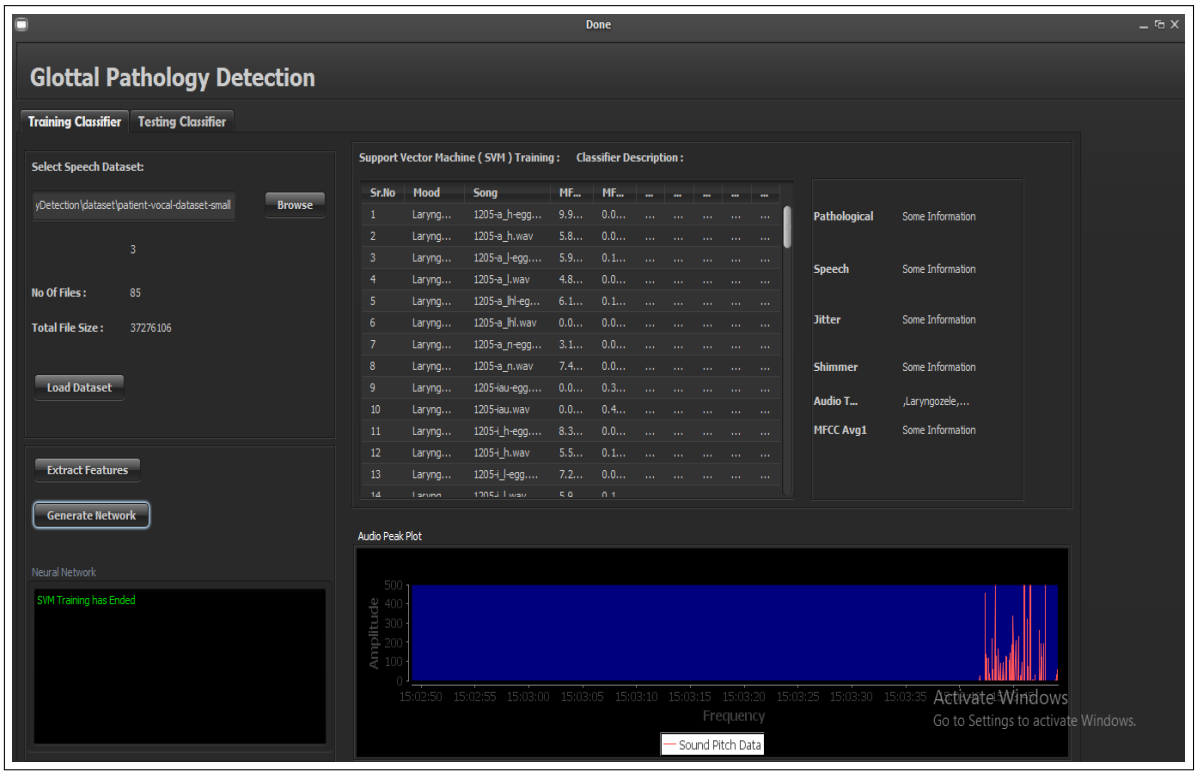

Figure 8.8: SVM Train

Chapter 9

CONCLUSION

9.1 CONCLUSION

The voice disorders are mainly caused due to defects in the speech organs, mental illness, hearing impairment, autism, paralysis or multiple disabilities. The traditional ways to diagnose voice pathology like direct inspection of the vocal folds and the observations of the vocal folds by endoscopic instruments are done. Such existing techniques are expensive, risky, time-consuming, discomfort to the patients.

The proposed system can give a low cost and quick solution by analyzing voice source features like 12 Mel-Frequency Filter Bank Cepstral Coefficients (MFCC) and zero-crossing rate (ZCR) in speech data for detection of glottal pathology.

9.2 FUTURE WORK

The classification accuracy can be increased in future by extracting additional features from the speech input.

9.3 APPLICATIONS

1. Used to detect glottal pathology at an early stage.

Annexure A

Feasibility Assessment

Problem statement feasibility assessment using, satisfiability analysis and NP Hard,NP-Complete

NP-Hard problem:

- NP-Complete and NP-Hard: A decision problem is in P if there is a known polynomial-time algorithm to get that answer. The collection of all problems that can be solved in polynomial time is called P. That is, a decision question is in P if there exists an exponent k and an algorithm for the question that runs in time O (nk) where n is the length of the input.
- A decision problem is in NP if there is a known polynomial-time algorithm for a Non-deterministic machine to get the answer.
- If the estimation cannot be solved in a fixed time or if we cannot define their execution complexity with a mathematical algorithm, then such problems are called as Non-Deterministic polynomial problems. Here comparison between the two algorithms is to be done to take the decision which is to used for prediction with better accuracy. **So, it is NP.**

NP-COMPLETE:

- The collection of all problems that can be solved in polynomial time using non-deterministic is called NP. That is, a decision question is in NP if there exists an exponent k and a non-deterministic algorithm for the question that for all hints runs in time O (nk) where n is the length of the input.

- The time required to extract the glottal features from voice input and perform classification by using SVM can be done in polynomial time. **So, it is NP-Complete**.
- All project algorithms can be determined in polynomial time but requires indefinite time for db. interaction. Hence all db. file handling projects are NP-COMPLETE.

Annexure B

Plagiarism Report

Plagiarism Report

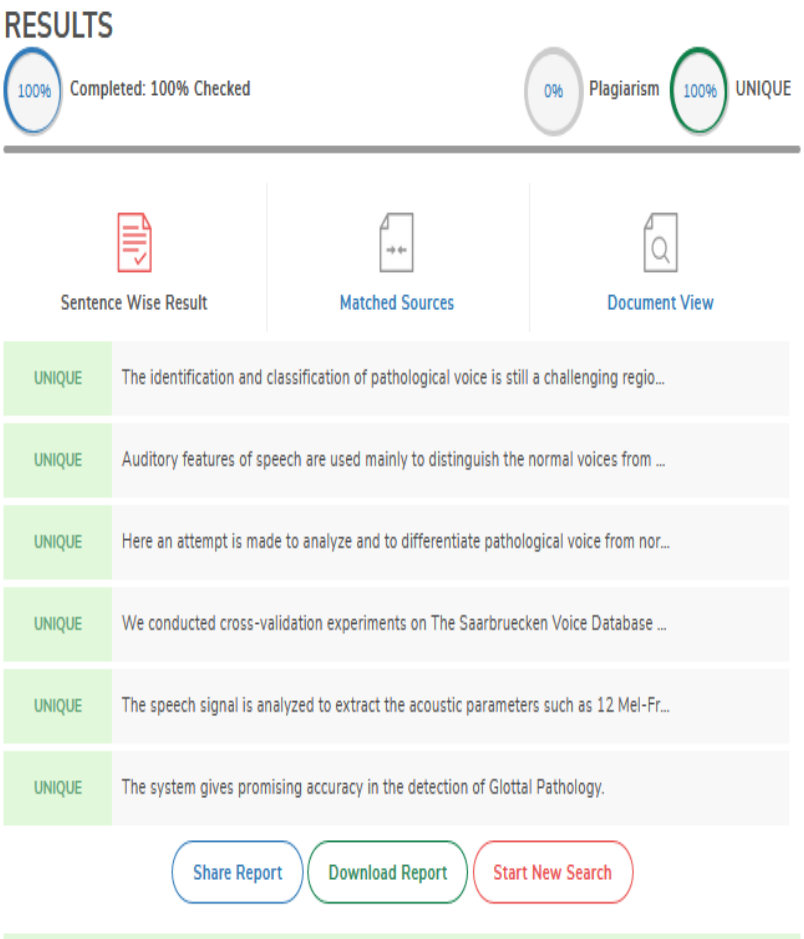

Figure B.1: Introduction Plagiarism

Figure B.2: Literature Survey Plagiarism

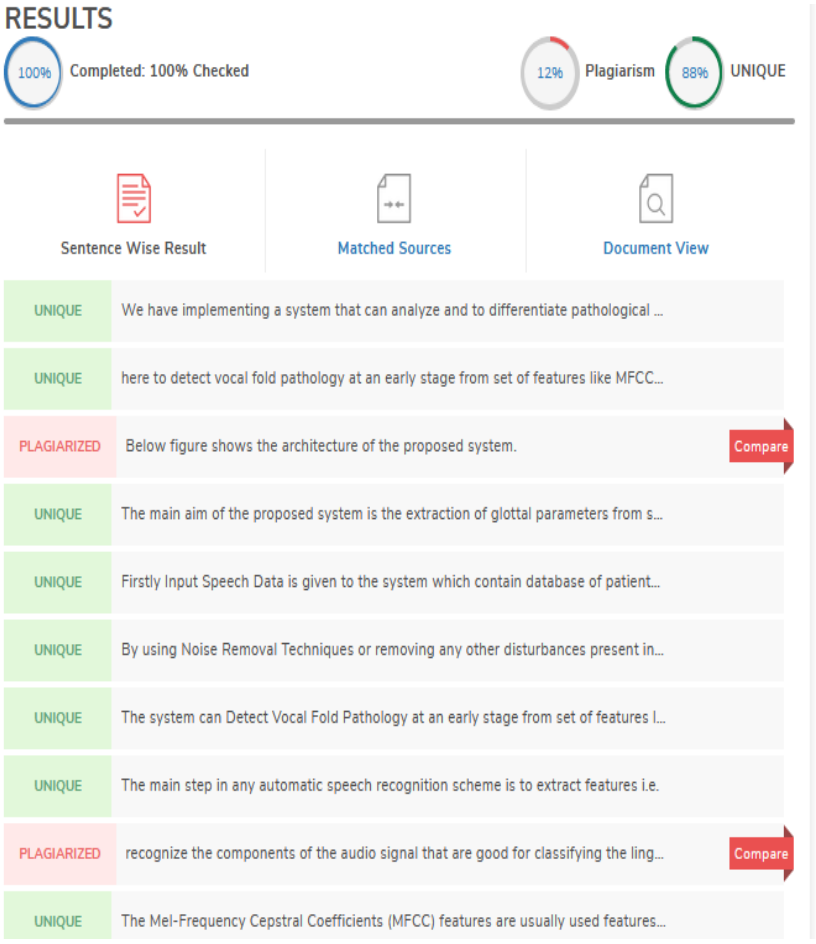

Figure B.3: Proposed System Plagiarism

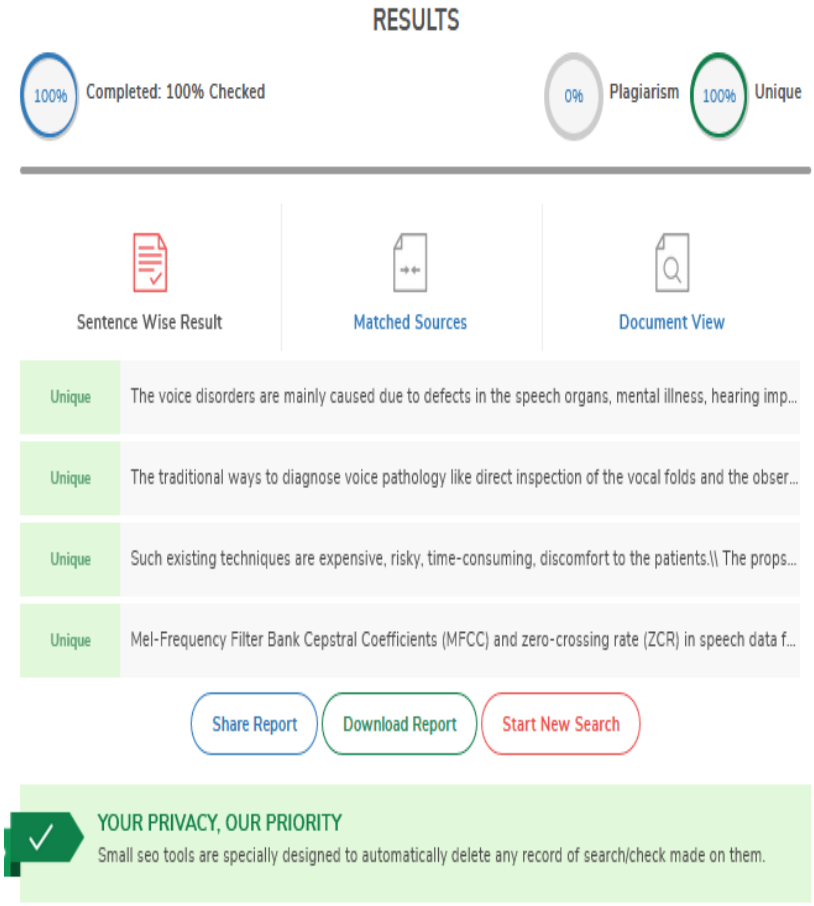

Figure B.4: Conclusion Plagiarism

Chapter 10

References

[1] Mythili J, Vijaya MS,"Pathology Voice Detection and Classification Using Ensemble Learning",International Journal of Engineering Science Invention (IJESI),2018.

[2] Pranav S. Deshpande ; M. Sabarimalai Manikandan ,"Effective Glottal Instant Detection and Electroglottographic Parameter Extraction for Automated Voice Pathology Assessment", IEEE Journal of Biomedical and Health Informatics (Volume: 22 , Issue: 2 , March 2018.

[3] S.C.Punitha,N.A. Sheela Selvakumari, "Voice Pathology Identification System using SVM Classifier" ,International Journal of Advance Research in Computer Science and Management Studies,2017.

[4] D. Pravena, S. Dhivya and A. Durga Devi, "Pathological Voice Recognition for Vocal Fold Didease," International Journal of Computer Applications (0975-888), Vol. 47, No. 13, 2012.

[5] N.A. Sheela Selvakumari,Dr. V.Radha, "Recent Survey on Feature Extraction Methods for Voice Pathology and Voice Disorde",International Journal of Computer Mathematical Sciences,August 2017.

[6] Anis Ben Aicha,"Noninvasive Detection of Potentially Precancerous Lesions of Vocal Fold Based on Glottal Wave Signal and SVM Approaches",International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2018, 3-5 September 2018, Belgrade, Serbia.

[7] Vahid Majidnezhad, Igor Kheidorov, “An ANN-based Method for Detecting Vocal Fold Pathology”, International Journal of Computer Applications (0975 –8887) Volume 62–No.7, January 2013.

[8] V.Srinivasan, V.Ramalingam, V.Sellam, “Classification of Normal and Pathological Voice using GA and SVM”, International Journal of Computer Applications (0975 –8887) Volume 60–No.3, December 2012.

[9] M.Farra, J. Hernando and P. Ejarque, “Jitter and Shimmer Measurements for Speaker Recognition,” TALP Research Center, Department of Signal Theory and Communications, Universitat Politecnica de Catalunya, Barcelona.

[10] Maria Markaki,Student Member, IEEE,and Yannis Stylianou,Member, IEEE,“Voice Pathology Detection and Discriminationbased on Modulation Spectral Features”,october, 2010. Aicha Bouzid and Noureddine Ellouze,“GLOTTAL OPENING INSTANT DETECTION FROM SPEECH SIGNAL”.

[11] V. Sellam, J. Jagadeesan,“Classification of Normal and Pathological Voice Using SVM and RBFNN” ,Journal of Signal and Information Processing, 2014, 5, 1-7, 2014.

[12] Korutla Sudhir Sai , Polasi Phani Kumar,“ Glottal Analysis Using Speech sig- nals” International Journal for Research in Applied Science Engineering Tech- nology (IJRASET),2017.

[13] CHRISTOPHE DALESSANDRO and NICOLAS STURMEL, “Glottal closure instant and voice source analysis usingtimescale lines of maximum amplitude”,Sadhan VOL. 36, Part 5, October 2011, pp. 601–622.c, Indian Academy of Sciences.

[14] L Gavidia-Ceballos and J H Hansen,“Direct speech feature estimation using an iterative EM algorithm for vocal fold pathology detection”. IEEE transactions on biomedical engineering, 43(4):373–83, April 1996.

[15] Hemant A. Patil,Pallavi N.Baljekar,“Classification of Normal and Pathological Voices using TEO Phase and Mel Cepstral Features”,2012 IEEE.