

A
Project Report
on
**AUTOMATIC SLIDE GENERATION
SYSTEM**

Submitted in Partial Fulfillment of
the Requirements for the Degree
of

B. E. of Engineering
in
Computer Engineering

to
North Maharashtra University, Jalgaon

Submitted by

Apeksha Todkar
Snehal Patil
Lina Sonawane
Mahesh Saindane

Under the Guidance of

Mrs. Shital A. Patil



DEPARTMENT OF COMPUTER ENGINEERING
SSBT's COLLEGE OF ENGINEERING AND TECHNOLOGY,
BAMBHORI, JALGAON - 425 001 (MS)
2016 - 2017

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

CERTIFICATE

This is to certify that the project entitled *Automatic Slide Generation System*,
submitted by

**Apeksha Todkar
Snehal Patil
Lina Sonawane
Mahesh Saindane**

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

Date: October 10, 2016

Place: Jalgaon

Mrs. Shital A. Patil
Guide

Prof. Dr. Girish K. Patnaik
Head

Prof. Dr. K. S. Wani
Principal

Acknowledgements

We take this opportunity to express our gratitude to all those who have rendered co-operation and guidance throughout our project Automatic Slide Generation System(ASGS). We want to express our sincere thanks to Dr. K. S. Wani. Principal of SSBT COET , Bambhori and Dr.Girish Kumar Patnaik (Head of the Department,Computer Engineering, SSBT COET, Bambhori).

We also extend our sincere thanks to our project guide Mrs. Shital Patil to guide us and provide us the opportunity to undertake this project and their valuable advices. And all other members of the faculty of Computer Engineering Department.

Last but not the least, We extremely indebted to our parents and friends and friends without whose support this report could have not reach its successful completion.

Thank You...!!!

Apeksha Todkar

Snehal Patil

Lina Sonawane

Mahesh Saindane

Contents

| | |
|---|-----------|
| Acknowledgements | ii |
| Abstract | 1 |
| 1 Introduction | 2 |
| 1.1 Background | 2 |
| 1.2 Motivation | 3 |
| 1.3 Problem Definition | 3 |
| 1.4 Scope | 4 |
| 1.5 Objective | 4 |
| 1.6 Summary | 5 |
| 2 System Analysis | 6 |
| 2.1 Literature Survey | 6 |
| 2.1.1 Existing Manual Slide Generation System | 6 |
| 2.1.2 How ASGS is better | 7 |
| 2.2 Proposed System | 8 |
| 2.3 Feasibility Study | 8 |
| 2.3.1 Economical Feasibility | 8 |
| 2.3.2 Operational Feasibility | 9 |
| 2.3.3 Technical Feasibility | 9 |
| 2.4 Risk Analysis | 10 |
| 2.5 Project Scheduling | 10 |
| 2.6 Effort Allocation | 11 |
| 2.7 Summary | 12 |
| 3 System Requirement Specification | 13 |
| 3.1 Hardware Requirements | 13 |
| 3.2 Software Requirements | 13 |
| 3.2.1 JDK 1.6 And Above | 13 |
| 3.2.2 OS-Open System(Linux) | 15 |

| | | |
|----------|--|-----------|
| 3.3 | Summary | 16 |
| 4 | System Design | 17 |
| 4.1 | System Archietecture | 17 |
| 4.2 | E-R Diagram | 18 |
| 4.3 | Data Flow Diagram | 19 |
| 4.4 | Interface Design | 20 |
| 4.4.1 | Module to Module Interaction | 20 |
| 4.5 | UML Diagram | 21 |
| 4.5.1 | Use Case Diagram | 22 |
| 4.5.2 | Sequence Diagram | 22 |
| 4.5.3 | Class Diagram | 24 |
| 4.5.4 | State Diagram | 25 |
| 4.5.5 | Activity Diagram | 26 |
| 4.5.6 | Component Diagram | 27 |
| 4.5.7 | Deployment Diagram | 27 |
| 4.6 | Summary | 28 |
| 5 | Conclusion | 29 |
| | Bibliography | 30 |
| | Index | 32 |

List of Figures

| | | |
|------|--|----|
| 2.1 | Gantt Chart for Project Scheduling | 11 |
| 2.2 | Effort Allocation Among Project Partners | 11 |
| 4.1 | System Architecture | 18 |
| 4.2 | ER Diagram | 19 |
| 4.3 | DFD Level 0 | 19 |
| 4.4 | DFD Level 1 | 20 |
| 4.5 | DFD Level 2 | 20 |
| 4.6 | Collaboration Diagram | 21 |
| 4.7 | Use Case Diagram | 22 |
| 4.8 | Sequence Diagram | 23 |
| 4.9 | Class Diagram | 24 |
| 4.10 | State Diagram | 25 |
| 4.11 | Activity Diagram | 26 |
| 4.12 | Component Diagram | 27 |
| 4.13 | Deployment Diagram | 28 |

Abstract

In most of the fields for sharing information, slide presentation plays an important role. The slides for presentation are traditionally prepared using various tools. The traditional way of presenting slides is labor-intensive. Labor-intensive nature leaves scope for human-errors. Also, for lengthy documents there is a chance of some vital information being missed out. Preparing slides manually consume more time. The drawbacks of the traditional way lead to need for intelligent system. The intelligent system needs to be capable of generating slides with minimum human interference. The existing automatic tools fail to fetch the graphical elements from a given input. Hence proposed an Automatic Slide Generation System. The proposed system fetches the graphical elements as well as text from a document. The proposed system is more reliable than the existing system.

Chapter 1

Introduction

Introduction chapter is introduce the work, It is focusses exactly on what is the area of project and explains what is actually be done in the work. All ideas about project work are cleared.

Chapter is of 7 sections. First section 1.1 describes background of the project, motivation is discuss in section 1.2, section 1.3 tell the problem denition behind the project, scope of the project is dened in section 1.4, section 1.5 gives objectives of the project, organization of the whole project is given in section 1.6 and section 1.7 gives summary.

1.1 Background

Presentation slides have been a popular and effective means to present and transfer information, especially in academic conferences. Many softwares such as Microsoft PowerPoint and OpenOfce to help to prepare slides. These tools only help in the formatting of the slides, but not in the content.

Proposed system, ***Automatic Slide Generation System(ASGS)*** automatically generates slides containing graphical elements as well as text data. The proposed system focuses on developing a data-mining technique, which is help in scoring the sentences as well as in generating slides with graphical elements. System is designed by applying Natural Language Processing (NLP) for scoring the sentences. While obtaining the expected importance score for every sentence from the given input paper, the Integer Linear Programming method (ILP) is utilized to generate well-structured slides. It is done by choosing and bringing into line key phrases and sentences along with diagrams. Different from the existing schemes produce slides by purely selecting significant sentences and placing sentences on the slides with only text element, Proposed system selects both key phrases and sentences for constructing well structured slides along with graphical elements. System uses key phrases as bullet points, and sentences applicable to the phrases are positioned below these bullet points. In order to extract key phrases, chunking implemented by the Open NLP library

is applied to the sentences and noun phrases are extracted as the candidate key phrases. Images are extracted from the whole document by using the PDF box and PDF documents. Finally the slides are generated and images are added to the slide.[1]

1.2 Motivation

In section discuss about why is the problem important to solve, It briey mention about the purpose, like how it is going to be helpful to the users. In Motivation, It write about why project is important, relevant or necessary.

Slide presentations play an important role in most fields for sharing information in an easy-to-present and visually-appealing format. Presentation slides are an effective medium to convey and share information and deliver key-messages across the audience at professional as well as educational meetings.

The task of producing presentation slides from one or several written materials is both tedious and time-consuming. The information encapsulated in slides typically includes key terminologies or summaries taken from a number of written materials. Many softwares to help the presenter to generate the slides. These tools only help them in the formatting of the slides, but not in the content. It still takes presenters much time to write the slides from scratch. The traditional tools thus require a lot of investment, both in terms of time and efforts In the work, It propose a method of automatically generating presentation slides. Automating the task of creating a slides from a given source would save presenters valuable time allowing them to concentrate on other aspects of the presentation such as preparing their speech. Project is extend the ones knowledge of Artificial Intelligence, focussing heavily on the field of Natural Language Processing.[2]

1.3 Problem Definition

The section describes which are the problem are about to solve. Provided sucient detail so someone can understand both, why and how it has been solved in the past. It denes what exactly the problem is trying to solve is signicant.

Presentation slides are an effective medium to convey and share information and deliver key-messages across the audience at professional as well as educational meetings. Traditional tools help users in setting up the theme and outline of the presentation; They do not help them in selecting the content for the slides. The traditional tools thus require a lot of investment, both in terms of time and efforts, from the users. The primary aim is to generate slides for the user so as to reduce their time and efforts in setting up presentation slides.

proposed which automatically generates slides containing graphical elements as well as

text data. The proposed system focuses on developing a data-mining technique, which is help in scoring the sentences as well as in generating slides with graphical elements. User primarily uploads the input file and system preprocess the input file in which the process of tokenization is done. In tokenization sentences are splits up into tokens. Stemming converts sentences into their first normal form. Post tagging gives score to sentences using Natural Language Processing. System finds graphical element of each page and stores the image into map data. System checks the label of each image and adds image to the slide according to the label of each image. The graphical elements along with the text data make the generated slides look more comprehensible and vivid.

1.4 Scope

In the section the scope And boundaries of project is to clearly described. Briey mentioned about the scope of the document and what it contains and to whom it is useful.

The main objective of Automatic Slide Generation System is to develop a student and company information system. Automating the task of creating a slides from a given source would save presenters valuable time allowing them to concentrate on other aspects of the presentation such as preparing their speech. Although the ideal solution would be able to cope with a multitude of subject-domains and produce presentations of varying length, due to the time scale given for completing the project, a number of constraints need to be outlined to keep the scope of the solution manageable. The solution is work specifically on articles in the scientific domain. The reason for choosing a scientific corpus is purely down to ease of availability and as the solution uses domain-independent techniques, corpora from other domains could easily be handled. The duration of the generated slides is limited to talks no greater than 15-20 minutes in length. It is the normal length of a conference talk and producing content for longer durations would be more problematic for the task of topic/text segmentation. The project is extend the ones knowledge of Artificial Intelligence, focussing heavily on the field of Natural Language Processing.[1]

1.5 Objective

In section clear picture of the project is given and objective is clearly specied. The system is proposed to generate well-structured presentation slides for pdf. The aim is to automatically generate well-structured slides and provide draft slides as a basis to reduce the presenters time and effort when preparing their nal presentation slides. The project is to design, implement and evaluate a method for automatically generating slides from text, to assist presenters in conferences in producing a presentation. The proposed system is capable of generating slides

with minimum human interference. The existing automatic tools fail to fetch the graphical elements from a given input. It is suggesting an automatic slide generation system. The proposed system fetches the graphical elements as well as text from a document. The proposed system is more reliable than the existing system.[10]

1.6 Summary

In this chapter, an overview of the problem statement along with its solution for the work contained in this dissertation is provided. In the next chapter, System analysis is presented.

Chapter 2

System Analysis

System analysis is the process of gathering and interpreting facts, diagnosing problems and using the facts to improve the system. It specifies what system should do. The same is discussed in chapter.

Chapter is of 7 sections. First section describes literature survey of the project, proposed system has discussed in section 2.2, section 2.3 tell the feasibility study behind the project, risk analysis of the project is dened in section 2.4, section 2.5 gives scheduling of the project, eort allocation of the whole project is given in section 2.6 and section 2.7 gives summary.

2.1 Literature Survey

The section has mentioned why application is needed, how it is better than the currently existing manual slide generation system. In section It has also thought the dimensions of comparison of work with others work.

2.1.1 Existing Manual Slide Generation System

Manual systems put pressure on people to be correct in all details of their work at all times, the problem being people are not perfect, Much each of us wishes were. With manual systems the level of service is dependent on individuals and puts a requirement on management to run training continuously for staff to keep them motivated and to ensure they are following the correct procedures. It can be all to easy to accidentally switch details and end up with inconsistency in data entry or in hand written orders. It has the effect of not only causing problems with customer service but also making information unable be used for reporting or finding trends with data discovery. Reporting and checking data is robust can be timely and expensive. It is often an area where significant money can be saved by automation.[4]

It takes more effort and physical space to keep track of paper documents, to find information and to keep details secure. When mistakes are made or changes or corrections are

needed, often a manual transaction must be completely redone rather than just updated. With manual or partially automated systems information often has to be written down and copied or entered more than once. Systemization can reduce the amount of duplication of data entry.

Disadvantages of manual slide generation system as follows:

1. Inconsistency in data entry, room for errors, miskeying information.
2. Large ongoing staff training cost.
3. System is dependent on good individuals.
4. Reduction in sharing information and customer services.
5. Time consuming and costly to produce reports.
6. Lack of security.
7. Duplication of data entry.

2.1.2 How ASGS is better

The task of producing presentation slides from one or several written materials is both tedious and time-consuming. The information encapsulated in slides typically includes key terminologies or summaries taken from a number of written materials. Many softwares to help the presenter to generate the slides. These tools only help them in the formatting of the slides, but not in the content. It still takes presenters much time to write the slides from scratch. The traditional tools thus require a lot of investment, both in terms of time and efforts, It propose a method of automatically generating presentation slides. Automating the task of creating a slides from a given source would save presenters valuable time allowing them to concentrate on other aspects of the presentation such as preparing their speech. Project extend the ones knowledge of Artificial Intelligence, focussing heavily on the field of Natural Language Processing. The primary aim is to generate slides for the user so as to reduce their time and efforts in setting up presentation slides.[5]

The project named Automatic Slide Generation System, is a window based application created in Core Java. In system, It propose a framework to naturally produce slides have great structure and content quality from scholarly papers and pdfs. The construction modeling of framework is appeared It utilize the SVR-based sentence scoring model to appoint a significance score for every sentence in the given paper, where the SVR model is prepared on a corpus gathered on the web. At point, It produce slides from the given paper by utilizing ILP. More subtle elements of every part has talked about in the accompanying areas.

2.2 Proposed System

In section describes about proposed system and techniques

The main purpose of the proposed system is to improve the eciency and reduce the time required for automatic slide generation while maintaining the graphical elements with the text elements.

Proposed system primarily uploads the input le and preprocess the input le in which the process of tokenization is done wherein sentences are splits up into tokens. Stemming is done in which the sentence is converted into their rst normal form. Stop words such as like but, also, etc are removed from the data. Postagging of procedure is applied and score is evaluated by using the NLP procedure. The most important objective of NLP is to intend and form such a computer scheme which is observe, recognize and construct NLP. Image is extracted by using PDFbox and PDF document from the input le. Finally, the slides with graphical images are generated.

The project named Automatic Slide Generation System, is a window based application created in Core Java. In system, It propose a framework to naturally produce slides have great structure and content quality from scholarly papers and pdfs. The construction modeling of framework is appeared. Utilize the SVR-based sentence scoring model to appoint a significance score for every sentence in the given paper, where the SVR model is prepared on a corpus gathered on the web. At point, It produce slides from the given paper by utilizing ILP. More elements of every part has talked about in the accompanying areas.

2.3 Feasibility Study

In section shows the all the aspects of the project and It can know whether the project is practically possible to develop worth limited resources and time.

The feasibility study is an evaluation and analysis of the potential of a proposed project which is based on extensive investigation and research to support the process of decision making. Feasibility studies aim to objectively and rationally uncover the strengths and weaknesses of an existing business or proposed venture, opportunities and threats present in the environment, the resources required to carry through, and ultimately the prospects for success.

2.3.1 Economical Feasibility

In subsection involves the feasibility of the proposed project to generate economic benets.

Economic feasibility is a cost benet Feasibility studies are crucial during the early development of any project and form a vital component in the business development process.

Accounting and Advisory feasibility studies enable organizations to assess the viability, cost and benefits of projects before financial resources are allocated. It also provides independent project assessment and enhances SSBT's College of Engineering and Technology, Bambhori, Jalgaon project credibility.

In current research, it is common to use a Recruitment System to model how once a country reaches a specific level of economic development, the configuration of structural factors, such as size of the commercial bourgeoisie, the ratio of urban to rural residence, the rate of political mobilization, etc., generate a higher probability of transitioning from authoritarian to democratic regime.

2.3.2 Operational Feasibility

In subsection define how proposed system is beneficial if they can be turned into information system that meets the organization operating requirements, defines how the system is userfriendly, and approved by the user showing no resistance and how it can be provided more facility when user wants.

The main purpose of the proposed system is to improve the efficiency and reduce the time required for automatic slide generation while maintaining the graphical elements with the text elements. Consequently, instead of finding single isolated patterns, focus on understanding the relationships between these patterns. The proposed approach was evaluated with a case study. The target of the case study was to classify real drilling data generated by rig sensors. Experimental evaluation proves the feasibility and effectiveness of the approach.

2.3.3 Technical Feasibility

In subsection shows the technology and the tools used to develop the system and describes to the ability of the process to take advantage of the current state of the technology in pursuing further improvement.

The main purpose of the proposed system is to improve the efficiency and reduce the time required for automatic slide generation while maintaining the graphical elements with the text elements.

Proposed system primarily uploads the input file and preprocess the input file in which the process of tokenization is done wherein sentences are split up into tokens. Stemming is done in which the sentence is converted into their first normal form. Stop words such as like but, also, etc are removed from the data. Postagging of procedure is applied and score is evaluated by using the NLP procedure. The most important objective of NLP is to intend and form such a computer scheme which observe, recognize and construct NLP. Image is extracted by using PDFbox and PDF document from the input file. Finally, the slides with

graphical images are generated.

Tool Used:

- JDK1.6- Several distinct features of Java are used which enhance usability of the project.

2.4 Risk Analysis

In section the analysis of the risk is discussed.

Project Risk Analysis and Management is a process which enables the analysis and management of the risks associated with a project. Properly undertaken it increase the likelihood of successful completion of a project to cost, time and performance objectives. The framework suggests an operational retrieval model extends recent developments in the language modeling approach to information retrieval. A language model for each document is estimated, as well as a language model for each query, and the retrieval problem is cast in terms of risk minimization. The query Language model can be exploited to model user preferences, the context of a query, synonymy and word senses. While recent work has incorporated word translation models for purpose.[7]

2.5 Project Scheduling

In section timeline chart (Gantt chart) for project activities is given, giving complete scheduling of all phases of project. Analysis of the project schedule and examination of the sequences, durations, resources and inevitable scheduling constraints is discussed. The purpose of section is to outline start and nish dates for project activities.

In project management, a schedule is a listing of a projects milestones, activities, and deliverables, usually with intended start and nish dates. Those items are often estimated in terms of resource allocation, budget and duration, linked by dependencies and scheduled events. Analysis of the project schedule and examination of the sequences, durations, resources and inevitable scheduling constraints is discussed. The purpose of section is to outline start and nish dates for project activities. A schedule is commonly used in project planning and project portfolio management parts of project management. Elements on a schedule is closely related to the work breakdown structure (WBS) terminal elements, the Statement of work, or a Contract Data Requirements List.

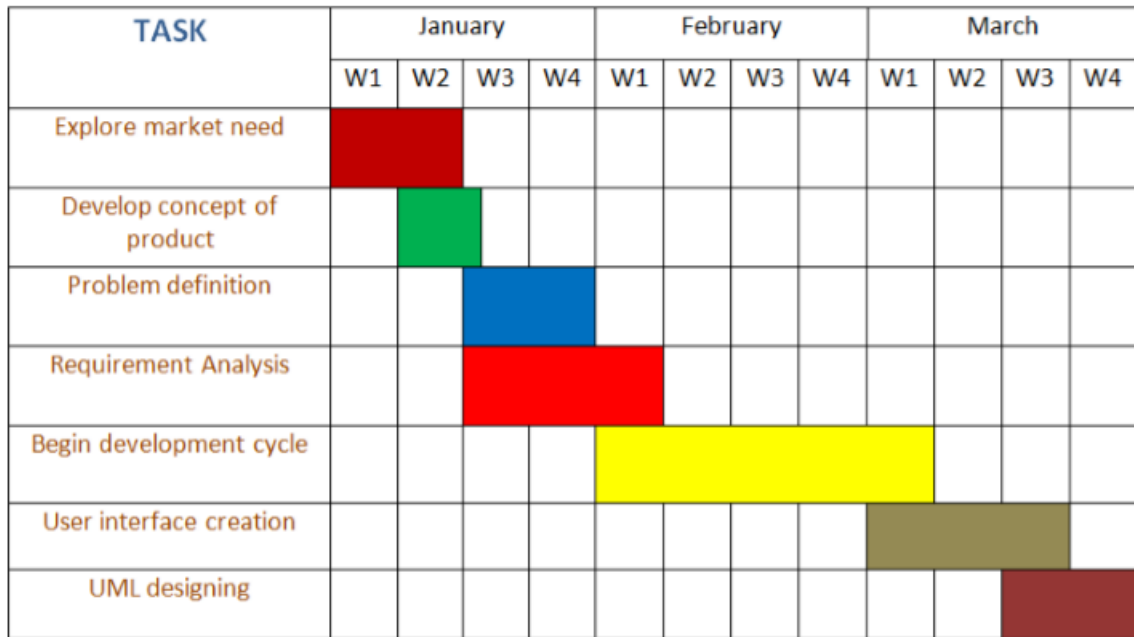


Figure 2.1: Gantt Chart for Project Scheduling

2.6 Effort Allocation

In section allocation of eorts to all project partners is given in the form of the table. Also general explanation based on Software engineering is done here.

A criterion was presented to dene the most ecient strategy for the exploration and main-tenance of plant genetic resources. All of the three factors composing the eciency. i.e., multiplicity of target populations.

| Activity | Apeksha Todkar | Snehal Patil | Lina Sonawane | Maresh Saindane |
|-----------------------|----------------|--------------|---------------|-----------------|
| Project Planning | 25% | 25% | 25% | 25% |
| Requirement Gathering | 20% | 25% | 27% | 28% |
| Design | 35% | 25% | 20% | 20% |

Figure 2.2: Effort Allocation Among Project Partners

The amount of expenses, and goodness for individual populations of the conservation manipulation adopted, were incorporated in the present criterion. Sample size per target

population for eld collection was investigated on the basis of criterion, leading to the conclusion the number of visited populations rather than sample size per population determines the overall efficiency of a collection project as a whole. Without any particular reason, intensive sampling for a limited number of populations is not logical.

2.7 Summary

In this chapter discussed about literature review of related work in image and document retrieval, existing system propose system and feasibility study of the project. Next chapter describes the Hardware and software requirement.

Chapter 3

System Requirement Specification

System requirement Specification is the ocial statement of what is required of the system developers. It include both user requirements and a detailed specication of the system requirements. Requirement analysis is done in order to understand the problem the software system is to solve.In chapter describes the system requirements of the project. Section 3.1 describe Hardware Requirements. Software Requirements describe in section 3.2.

3.1 Hardware Requirements

The various hardware requirements of the system are summarized in section.

The hardware requirements are as follow:

- Processor: Intel(R)Core(TM)2 Duo and Intel Core(TM) i5
- Hard-disk: 40GB
- RAM: 2GB (2048MB)
- Monitor : 15 VGA color

3.2 Software Requirements

The various software requirements of the system is summarized in section. The Software requirements are as follow:

3.2.1 JDK 1.6 And Above

JDK stands for Java Development Kit. It is a collection of software allows a developer to create and deploy an application written in Java. JDK 7 is a superset of JRE 7, and contains everything that is in JRE 7, plus tools such as the compilers and deBuggers necessary for developing applets and applications. JRE 7 provides the library the Java Virtual

Machine (JVM), and other components to run applets and applications written in the Java programming language.

following features of JDK 1.6:

1. **Collections Framework Enhancements:** Java SE 6 API provides bi-directional collection access. New collection interfaces includes Deque, NavigableSet, NavigableMap.
2. **java.io Enhancements:** New class Console is added and it contains methods to access a character-based console device. The readPassword() methods disable echoing thus they are suitable for retrieval of sensitive data such as passwords. The method System.console () returns the unique console associated with the Java Virtual Machine.
3. **GUI:** - JFC and Swing integration with desktop by using Windows API.
-Java 2D integration with desktop such as using desktop anti-aliasing font settings
Splash screen direct support and can be shown before JVM started
System tray support with ability to add icons, tool tips, and pop-up menus to the Windows or any other system tray (such as Gnome).
4. **Security Features and Enhancements:** Native platform Security (GSS/Kerberos) integration.
Java Authentication and Authorization Service (JAAS) login module that employs LDAP authentication
-New Smart Card I/O API
5. **Java™ API for XML Processing (JAXP):** The Java API for XML Processing (JAXP) enables applications to parse, transform, validate and query XML documents using an API that is independent of a particular XML processor implementation. JAXP provides a pluggability layer to enable vendors to provide their own implementations without introducing dependencies in application code. Using software, application and tool developers can build fully-functional XML-enabled Java applications for e-commerce, application integration, and web publishing.

The Java Platform, Standard Edition version 6.0 includes JAXP 1.4. JAXP 1.4 is a maintenance release of JAXP 1.3 with support for the Streaming API for XML (StAX).
6. **JDBC 4.0 Enhancements :** Java SE 6 includes several enhancements to the Java Database Connectivity (JDBC) API. These enhancements has released as JDBC version 4.0. The main objectives of the new JDBC features are to provide a simpler design and better developer experience

The major features added in JDBC 4.0 include:

- Auto-loading of JDBC driver class
- Connection management enhancements
- Support for RowId SQL type
- DataSet implementation of SQL using Annotations
- SQL exception handling enhancements

7. **Auto-Loading of JDBC Driver:** In JDBC 4.0, No longer need to explicitly load JDBC drivers using Class.forName(). When the method getConnection is called, the DriverManager attempt to locate a suitable driver from among the JDBC drivers that were loaded at initialization and those loaded explicitly using the same class loader as the current application.

3.2.2 OS-Open System(Linux)

Linux was originally developed as a free operating system for personal computers based on the Intel x86 architecture, but has since been ported to more computer hardware platforms than any other operating system. Because of the dominance of Android on smart phones, Linux has the largest installed base of all general-purpose operating systems. Linux is also the leading operating system on servers and other big iron systems such as mainframe computers and virtually all fastest supercomputers, but is used on only around 2.3 percent of desktop computers when not including Chrome OS, which has about 5 percent of the overall and nearly 20 percent of the sub- 300 notebook sales. Linux also runs on embedded systems, which are devices whose operating system is typically built into the firmware and is highly tailored to the system; It includes smart phones and tablet computers running Android and other Linux derivatives, TiVo and similar DVR devices, network routers, facility automation controls, televisions, video game consoles and smartwatches.

The development of Linux is one of the most prominent examples of free and open-source software collaboration. The underlying source code is used, modified and distributed commercially or non-commercially by anyone under the terms of its respective licenses, such as the GNU General Public License. Typically, Linux is packaged in a form known as a Linux distribution (or distro for short) for both desktop and server use. Some of the most popular mainstream Linux distributions are Arch Linux, CentOS, Debian, Fedora, Gentoo Linux, Linux Mint, Mageia, openSUSE and Ubuntu, together with commercial distributions such as Red Hat Enterprise Linux and SUSE Linux Enterprise Server. Distributions include the Linux kernel, supporting utilities and libraries, many of which are provided by the GNU

Project, and usually a large amount of application software to fulfil the distribution's intended use.

Distributions oriented toward desktop use typically include a windowing system, such as X11, Mir or a Wayland implementation, and an accompanying desktop environment such as GNOME or the KDE Software Compilation; some distributions also include a less resource-intensive desktop, such as LXDE or Xfce. Distributions intended to run on servers omit all graphical environments from the standard install, and instead include other software to set up and operate a solution stack such as LAMP. Because Linux is freely redistributable, anyone create a distribution for any intended use.

3.3 Summary

The chapter presented about hardware software requirement. In the next chapter System design is presented.

Chapter 4

System Design

System design provides the understanding and procedural details necessary for implementing the system recommended in the system study. In chapter, Making the design of the system. Flowchart, how the data flows from the system architecture, and all the UML diagrams including class diagram, use case diagram, sequence diagram, activity data diagram, component diagram, deployment diagram of the project.

In Section 4.1 gives system architecture, ER-diagrams is presented in section 4.2, 4.3 is presented database design, data flow diagrams are presented in section 4.4, section 4.5 presented interface design, and UML diagrams are in section 4.6.

4.1 System Architecture

The basic architecture for the proposed system is shown in section. The section defines overall structure and the components that make up the system.[1]

The following figure shows the architectural view of the proposed system. The description of the system is as follows:

Proposed system primarily uploads the input file and preprocess the input file in which the process of tokenization is done wherein sentences are splits up into tokens. Stemming is done in which the sentence is converted into their first normal form. Stop words such as like but, also, etc are removed from the data. Postagging of procedure is applied and score is evaluated by using the NLP procedure. The most important objective of NLP is to intend and form such a computer scheme which observe, recognize and construct NLP. Image is extracted by using PDFbox and PDF document from the input file. Finally, the slides with graphical images are generated.

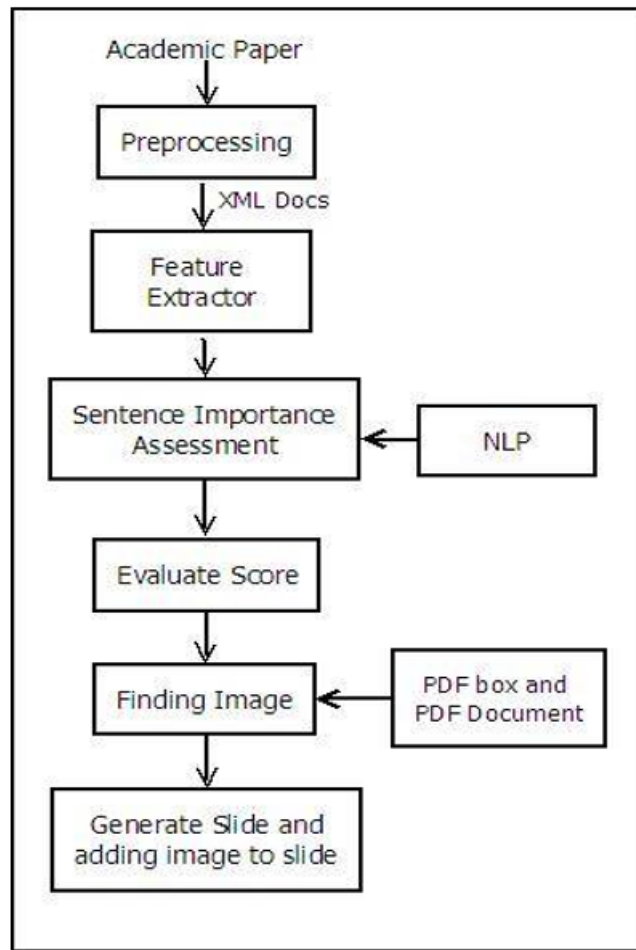


Figure 4.1: System Architecture

4.2 E-R Diagram

Entity-Relationship model and the operations to be done on those entities are shown by E-R diagram in section. A primary goal of database design is to decide what tables to create. Usually, Two principles:

- Capture all the information that needs to be captured by the underlying application.
- Achieve the above with little redundancy.

The first principle is enforced with an entity relationship (ER) diagram, while the second with normalization.

An ER diagram is a pictorial representation of the information can be captured by a database. Such a picture serves two purposes: It allows database professionals to describe an overall design concisely yet accurately. It can be easily transformed into the relational schema.

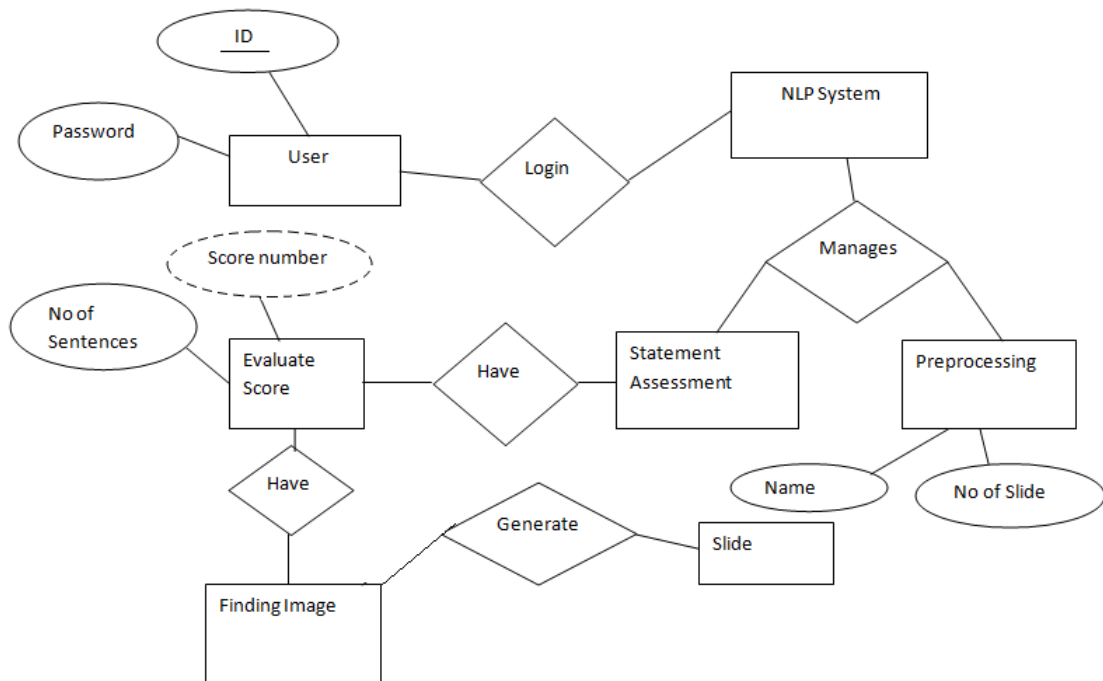


Figure 4.2: ER Diagram

4.3 Data Flow Diagram

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design). A DFD Shows what kind of information has input to and output from the system, where the data is come from and go to, and where the data is store. Section includes DFD0, DFD1 and DFD2.

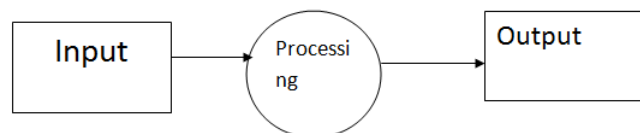


Figure 4.3: DFD Level 0

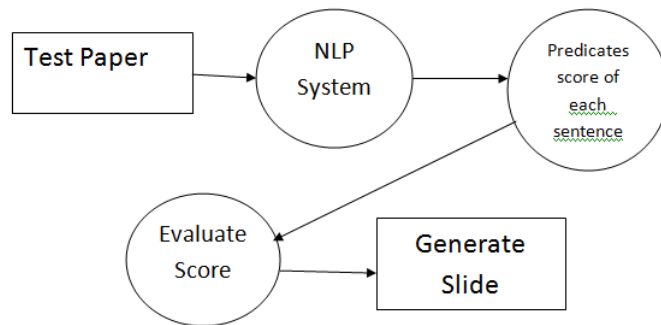


Figure 4.4: DFD Level 1

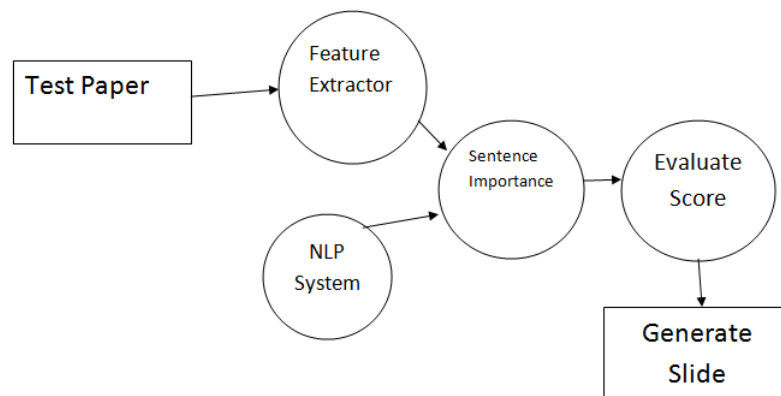


Figure 4.5: DFD Level 2

4.4 Interface Design

The section includes the description of module to module interaction using Collaboration Diagram.

4.4.1 Module to Module Interaction

A collaboration diagram, also called a communication diagram or interaction diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML) as shown in Figure 4.6. The concept is more than a decade old although it has been refined as modeling paradigms have evolved.

A collaboration diagram resembles a flowchart that portrays the roles, functionality and behavior of individual objects as well as the overall operation of the system in real time. Objects are shown as rectangles with naming labels inside. These labels are preceded by colons and underlined.

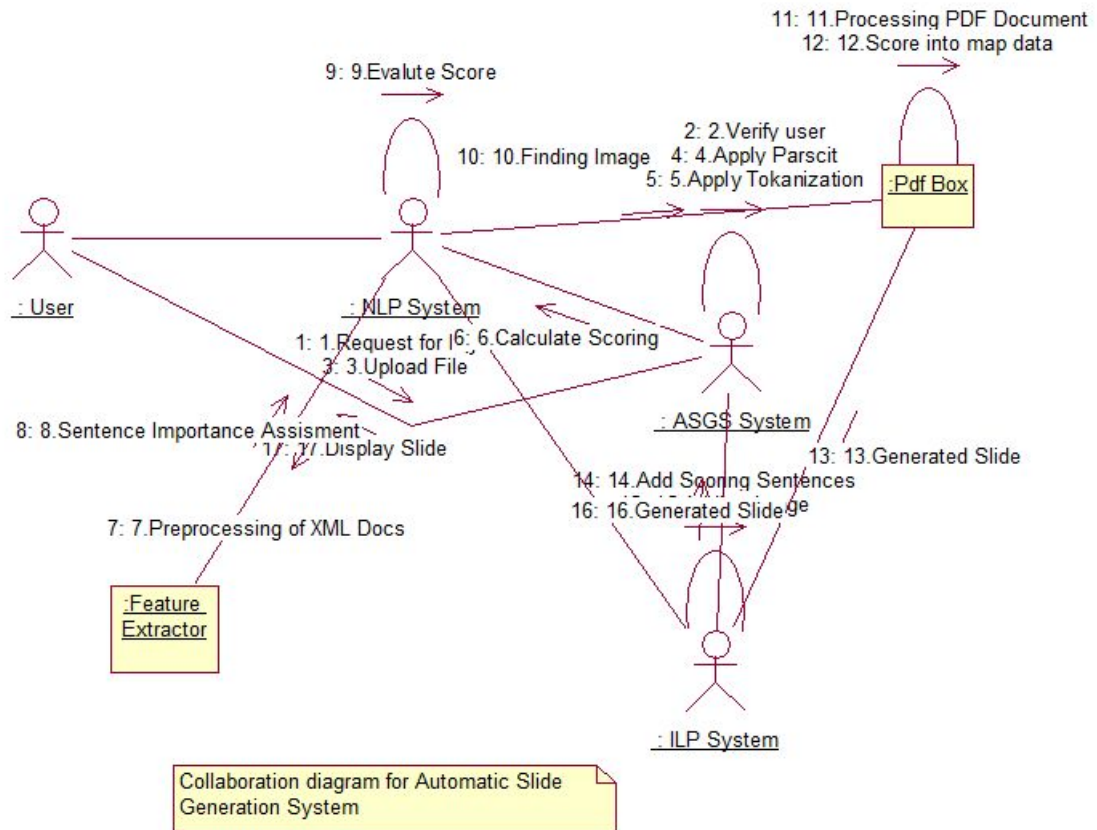


Figure 4.6: Collaboration Diagram

4.5 UML Diagram

Unified Modeling Language is a standard models application and business structures, processes and data. Standard helps design large and complex software programs before the code is created. UML allows users to visualize the big picture without worrying about the many small details required while programming. UML is a standard created by the not-for-profit technology standards consortium Object Management Group Inc. The standard includes different diagram types separated into three categories: structure, behavior and interaction. These include the use case diagram, class diagram, sequence diagram, activity diagram, component diagram, deployment diagram and state diagram. The section gives all these diagrams.

4.5.1 Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and often be accompanied by other types of diagrams as well. Section gives the Use Case Diagram of the proposed system as shown in Figure 4.7.

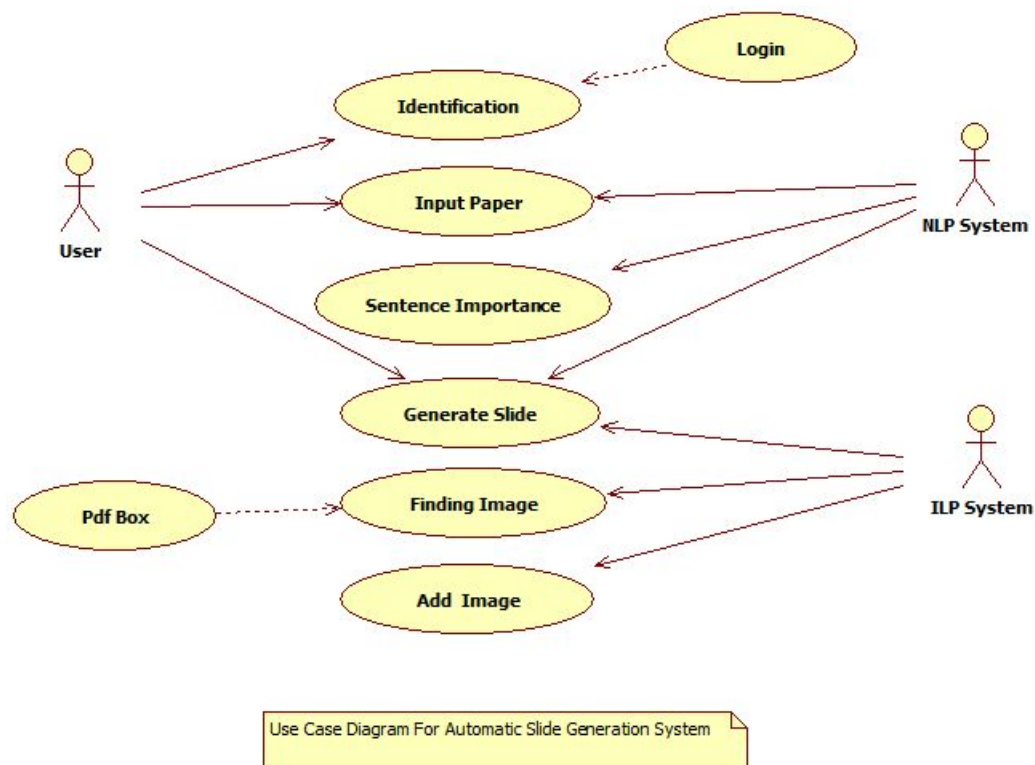


Figure 4.7: Use Case Diagram

4.5.2 Sequence Diagram

A Sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart.

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry

out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios. Section gives the sequence diagram as shown in Figure 4.8.

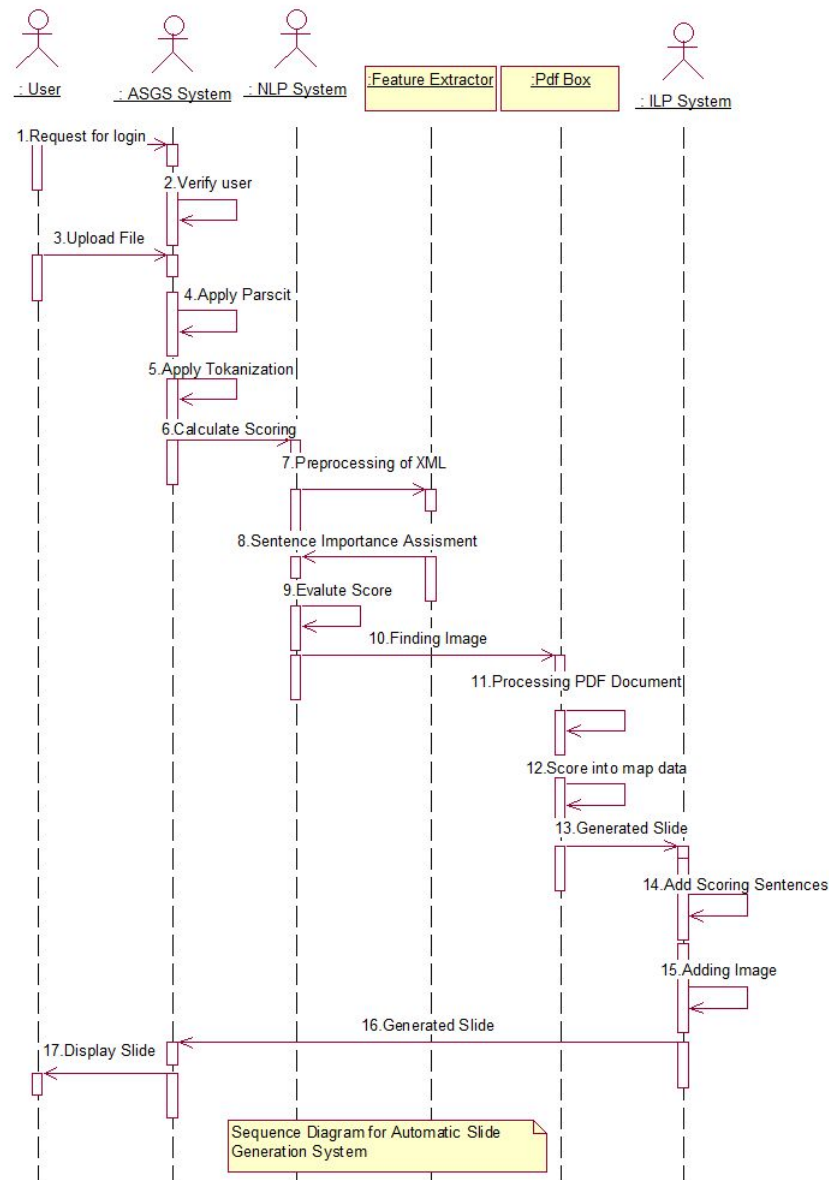


Figure 4.8: Sequence Diagram

4.5.3 Class Diagram

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

The class diagram is the main building block of object-oriented modelling. It is used both for general conceptual modelling of the systematics of the application, and for detailed modelling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed. Section gives the class diagram as shown in Figure 4.9.

In the diagram, classes are represented with boxes that contain three compartments:

The top compartment contains the name of the class. It is printed in bold and centered, and the first letter is capitalized. The middle compartment contains the attributes of the class. Left-aligned and the first letter is lowercase. The bottom compartment contains the operations the class can execute. It is also left-aligned and the first letter is lowercase.

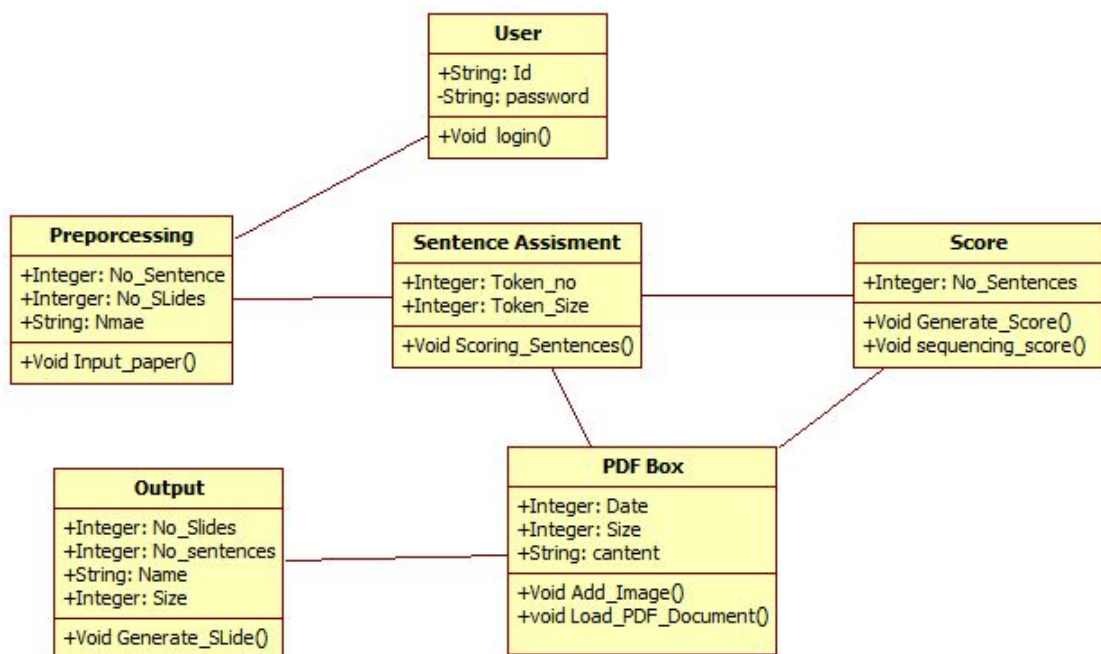


Figure 4.9: Class Diagram

4.5.4 State Diagram

A state diagram is a type of diagram used in computer science and related fields to describe the behavior of systems. State diagrams require the system described is composed of a finite number of states; sometimes, It is indeed the case, while at other times It is a reasonable abstraction .

UML state machine is an object-based variant of Harel statechart, adapted and extended by UML. The goal of UML state is to overcome the main limitations of traditional finite-state machines while retaining main benefits. UML state charts introduce the new concepts of hierarchically nested states and orthogonal regions, while extending the notion of actions. Many forms of state diagrams exist, which differ slightly and have different semantics. Section gives the state diagram as shown in Figure 4.10.

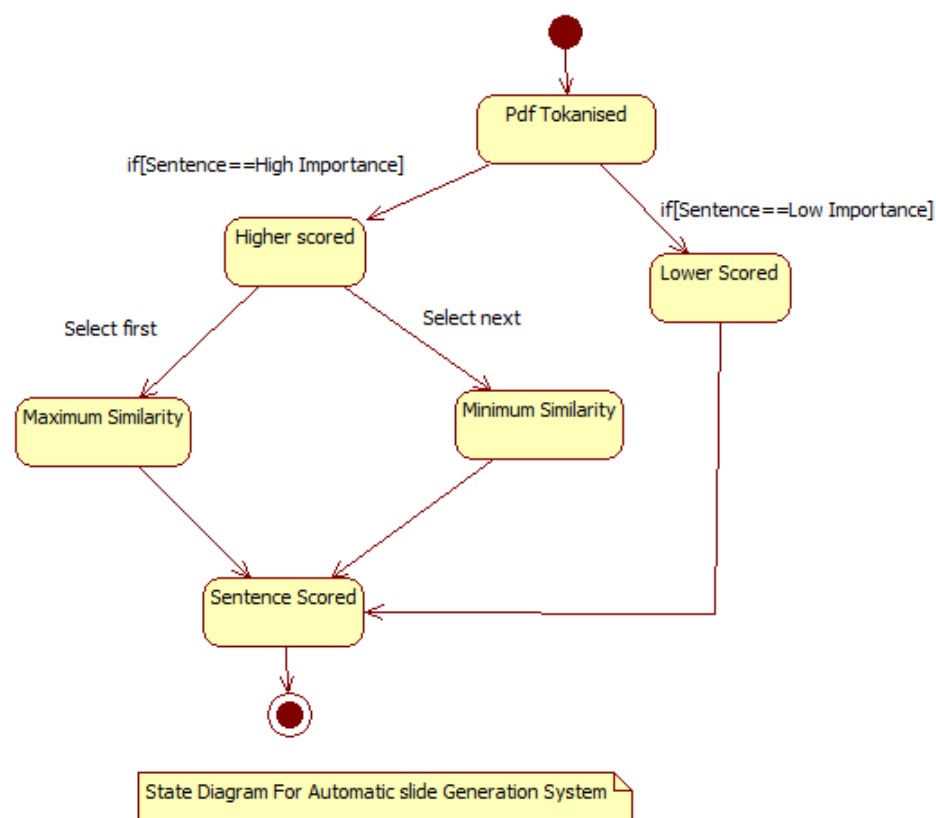


Figure 4.10: State Diagram

4.5.5 Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e. workflows). Activity diagrams has regarded as a form of flowchart. Typical flowchart techniques lack constructs for expressing concurrency as shown in Figure 4.11.

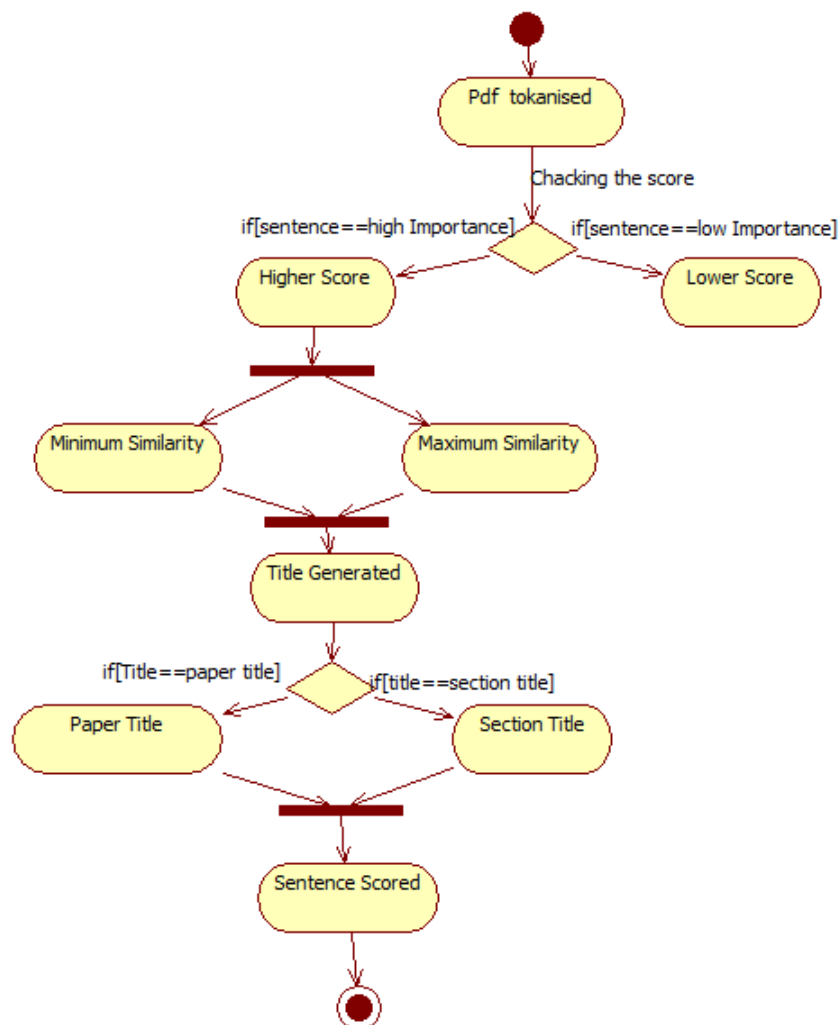


Figure 4.11: Activity Diagram

Activity diagrams are constructed from a limited number of shapes, connected with arrows. The most important shape types:

- Rounded rectangles represent actions
- Diamonds represent decisions

- Bars represent the start (split) or end (join) of concurrent activities
- A black circle represents the start (initial state) of the workflow
- An encircled black circle represents the end (final state)

The join and split symbols in activity diagrams only resolve for simple cases; the meaning of the model is not clear when they are arbitrarily combined with decisions or loops. Activity diagrams show the overall flow of control. Section gives the activity diagram.

4.5.6 Component Diagram

In the Unified Modeling Language, a component diagram depicts how components are wired together to form larger components and or software systems. Using a component diagram to show the internal structure of a component, the provided and required interfaces of the encompassing component can delegate to the corresponding interfaces of the contained components. A delegation connector is a "connector that links the external contract of a component (as specified by its ports) to the internal realization of that behavior by the components parts. It is used to illustrate the structure of arbitrarily complex systems. Section gives the component diagram as shown in Figure 4.12.

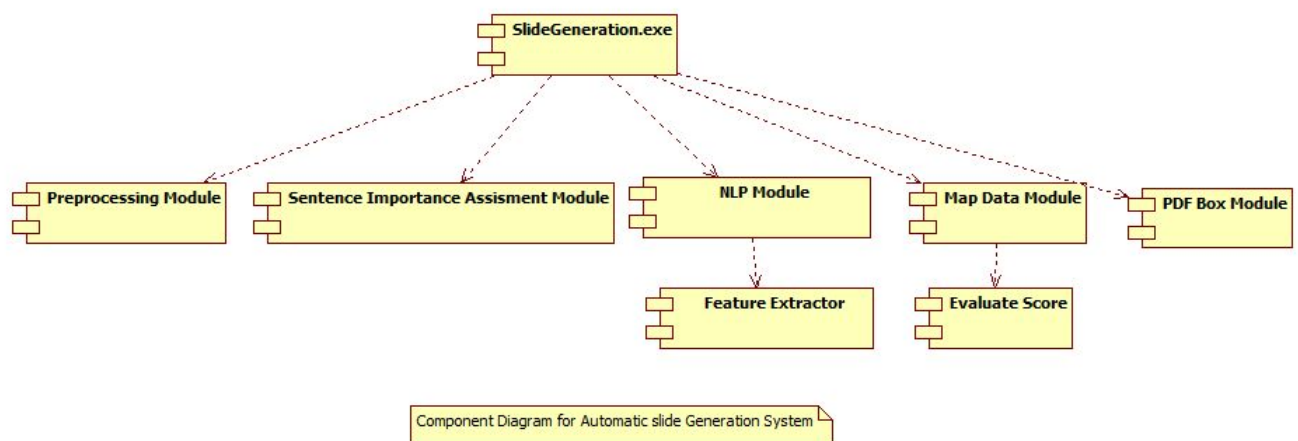


Figure 4.12: Component Diagram

4.5.7 Deployment Diagram

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes. To describe a web site, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server,

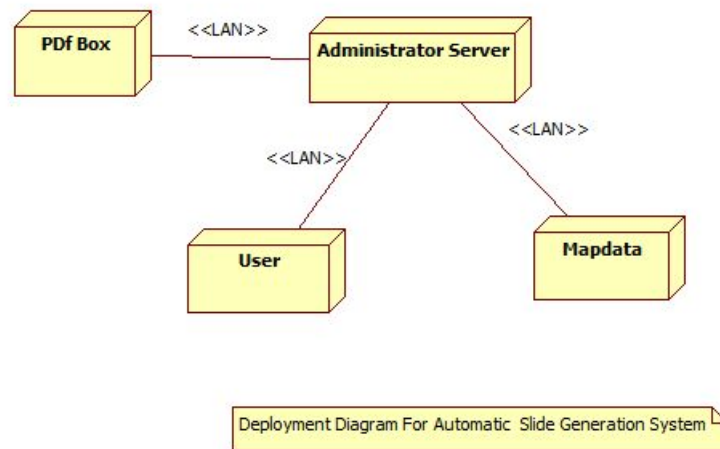


Figure 4.13: Deployment Diagram

and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI. Section gives the deployment diagram as shown in Figure 4.13. The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes have subnodes, which appear as nested boxes. A single node in a deployment diagram conceptually represent multiple physical nodes, such as a cluster of database servers.

Two types of Nodes:

- Device Node
- Execution Environment Node

4.6 Summary

In this chapter described the overall system architecture, E-R diagram, DFDs, interface design and UML diagrams. The next chapter is give the project implementation in detail.

Chapter 5

Conclusion

The system is proposed for automatically generating the slides from the pdf. The system generates the slides which includes text and graphical element. The graphical elements along with the text data make the generated slides look more comprehensible and vivid. System initially finds the graphical elements of each page from the paper and after that system stores the image to map data. System performs the same operation for each page and finally checks the label of each image and adds that image to the slide according to the label of each image. NLP method is used for sentence scoring and the ILP method is used for slide generation which contain key phrases and the relevant sentences. Presently, the system generate slides on the basis of only one given pdf, but in future extra information like additional relevant pdfs and information of citation can be utilized for improving the slides generation.

Bibliography

- [1] Ektaa Meshram, D. A. Phalke, "Technique for Generating Automatic Slides on the basis of Paper Structure Analysis", International Journal of Innovative Research in Science, Engineering and Technology, Vol. 5, Issue 6, June 2016.
- [2] Yue Hu and Xiaojun Wan, "PPSGen: Learning-Based Presentation Slides Generation for Academic Papers, in IEEE transactions on knowledge and data engineering, vol. 27, no. 4, april 2015.
- [3] V. Qazvinian, D. R. Radev, S. M. Mohammad, B. J. Dorr, D. M. Zajic, M. Whidby, and T. Moon, Generating extractive summaries of scientic paradigms, J. Artif. Intell. Res., vol. 46, pp. 165 201, 2013.
- [4] U. Masao and H. Koiti, Automatic slide presentation from semantically annotated documents, in Proceedings of the Workshop on Coreference and its Applications, pp. 2530, Association for Computational Linguistics, 2012.
- [5] Y. Yasumura, M. Takeichi, and K. Nitta, A support system for making presentation slides, Transactions of the Japanese Society for Artificial Intelligence, vol. 18, pp. 212220, 2011.
- [6] T. Hayama, H. Nanba, and S. Kunifuji, Alignment between a technical paper and presentation sheets using a hidden markov model, in Active Media Technology, 2005.(AMT 2005). Proceedings of the 2005 International Conference on, pp. 102106, IEEE, 2005.
- [7] T. Shibata and S. Kurohashi, Automatic slide generation based on discourse structure analysis, in Natural Language ProcessingIJCNLP 2005, pp. 754766, Springer, 2005.
- [8] M.-Y. Kan, Slideseer: A digital library of aligned document and presentation pairs, in Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries, pp. 81 90, ACM, 2007.
- [9] S. M. A. Masum and M. Ishizuka, Making topic specific report and multimodal presentation automatically by mining the web resources, in Proc. IEEE/WIC/ACM Int. Conf. Web Intell., 2006, pp. 240246.

- [10] D. Galanis, G. Lampouras, and I. Androutsopoulos, Extractivemulti-document summarization with integer linear programmingand support vector regression, in Proc. COLING, 2012,pp. 911926.

Index

ASGS, 3

ILP, 8

NLP, 8

Postagging, 17

SVR Model, 7

systemization, 7

Tokenization, 10