A

Project-I Report

on

KEYWORDS EXTRACTION FROM CRIME INFORMATION USING MISCELLANEOUS DATA SOURCES

Submitted in Partial Fulfillment of
the Requirements for the Degree
of

Bachelor of Engineering

in

Computer Engineering

to

North Maharashtra University, Jalgaon

Submitted by

Dhanashri Rajesh Chaudhari
Vrushali Rajesh Malvadkar
Sucheta Sanjay Jadhav
Ankush Babasaheb Pandit

Under the Guidance of

Prof. Dr. Girish K. Patnaik

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-I entitled *Keywords Extraction from Crime Information using Miscellaneous Data Sources* , submitted by

Dhanashri Rajesh Chaudhari
Vrushali Rajesh Malvadkar
Sucheta Sanjay Jadhav
Ankush Babasaheb Pandit

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

Date: October 7, 2016
Place: Jalgaon

Prof. Dr. Girish K. Patnaik
Guide

Prof. Dr. Girish K. Patnaik
Head

Prof. Dr. K. S. Wani
Principal

Acknowledgements

First of all we would like to extend our deep gratitude to almighty God, who has enlightened us with power of knowledge. We wish to express our sincere gratitude to the principal Prof. Dr. Kishor S. Wani for giving us such a great opportunity to develop this project. A special gratitude to our Head of Department and project guide, Prof. Dr. Girish K. Patnaik, whose contribution in stimulating suggestions and encouragement, helped us to coordinate our project especially in writing this report. He has invested his full effort in guiding the team in achieving the goal.

Furthermore we would also like to acknowledge with much appreciation the crucial role of the staff of our department, who gave the permission to use all required equipment and the necessary materials to complete this project report. We would like to thanks to our parents and all friends for their co-operation and supports in making this project report successful. We acknowledge our sincere gratitude to all who have directly or indirectly helped us in completing this report successfully.

Dhanashri Rajesh Chaudhari
Vrushali Rajesh Malvadkar
Sucheta Sanjay Jadhav
Ankush Babasaheb Pandit

Contents

List of Figures

Abstract

Much information that helps to solve and prevent crimes are never gathered. Crime reporting methods available to citizens and law enforcement personnel are not optimal. Existing text based and email system provides little support for witness memory recall. Natural language processing is used to extract crime related keywords from such lengthy police and witness narrative reports. Crime information extraction helps police investigators quickly comprehend crime incidents without having to read an entire report. The evaluation is done by analyzing precision and recall. The system uses crime information extraction rules crime information extraction rules that powers the lexicons to extract useful information from various sources and with diverse formats.

Chapter 1

Introduction

In India, a property crime is committed on average every 3 seconds; every 22 seconds, a violent crime is committed. Today, most crimes are solved after investigators interview witnesses and victims, analyze the resulting narrative reports, and combine the information. Missing, non response, or partial non response data often occurs when data is collected using questionnaires or structured interviews. Interviewers may fail to record data, want to finish an interview quickly, or record data incorrectly or illegibly. In addition, analyzing salient clues within a significant number of narrative reports is an arduous task for police investigators and efficiency could be improved if text reports could be automatically analyzed to obtain relevant crime related information, such as vehicle names, addresses, narcotics, and peoples identities. Information technology, e.g., information extraction, can be used to obtain crime-related information more efficiently.

Direct reporting by witnesses and victims is facilitated by developing a crime information extraction (IE) system. Witness narrative reports obtained from online forums and blogs and police narrative reports from police departments and newspaper articles. Automatically extracting information, combining information from crime narrative reports, and presenting a meaningful summary for police investigators will help them quickly comprehend crime incidents without having to read an entire report. Working with original witness crime reports that contain first-hand information is ideal, but difficult to achieve. In general these documents are difficult to obtain due to confidentiality or privacy concerns, and they often contain spelling and grammatical errors.

A successful and useful crime information extraction system should achieve high precision and recall regardless of the type and origin of the information. By analyzing crime corpora from heterogeneous data sources a, rich and substantial lexicon can be compiled. Crime Information Extraction rules powers the lexicons to extract useful information from various sources and with diverse formats. The performance is based on comparison of IE for police narrative reports and witness or victim narrative report.

Section 1.1 describes background. Motivation is presented in Section 1.2. Section 1.3 describes problem definition. Scope is presented in Section 1.4. Section 1.5 describes the Objective. Organization of the Report is presented in Section 1.6. Section 1.7 presents Summary.

1.1 Background

Police department and investigators comprehend crime by reading and analyzing witness and victims report. The witness and victims report are quite lengthy to go through. In police station crime reporting is done by simply interviewing the witness and victims. The reporting method is text based. Reading and summarizing the contents of large report entries of text into a small set of topic is difficult and time consuming for a human, so much so that it becomes nearly impossible to accomplish with limited manpower as the size of the information grows.

Keywords are commonly used for search engines and document databases to locate information and determine if two pieces of test are related to each other. Reading and summarizing the contents of large report entries of text into a small set of topic is difficult and time consuming for a human, so much so that it becomes nearly impossible to accomplish with limited manpower as the size of the information grows. As a result, automated systems are being more commonly used to do this task. This problem is challenging due to the intricate complexities[3] of natural language, as well as the inherent difficulty in determining if a word or set of words accurately represent topics present within the text. With the advent of the internet, there is now both a massive amount of information available, as well as a demand to be able to search through all of this information. Keyword extraction from text data is a common tool used by search engines and indexes alike to quickly categorize and locate specific data based on explicitly or implicitly supplied keywords.

Various methods of locating and defining keywords have been used, both individually and in concert.[2]

- Word Frequency Analysis
- Word Co-Occurrence Relationships
- Frequency-Based Single Document Keyword Extraction
- Keyword Extraction Using Lexical Chains
- Keyphrase Extraction Using Bayes Classifier
- Content-Sensitive Single Document Keyword Extraction

1.2 Motivation

Most of the crimes are solved after interviewing witnesses and victims by investigators. Investigators analyse the resulting narrative reports. Data is collected using questionnaires or structured interviews contains Missing, nonresponse, or partial nonresponse data in report. Interviewers may fail to record data, or record data incorrectly or illegibly. Analysing number of narrative reports is an difficult task. A meaningful summary for police investigators is generated automatically by extracting information[1] using crime narrative reports which helps police investigators quickly comprehend crime incidents without having to read an entire report.

1.3 Problem Definition

Develop a crime information extraction system. The system administrator should collect various witness and police narrative reports from data sources like newspaper articles and internet. Data should be filtered by information extraction manager and the keywords get extracted. Gold standard should be established by system administrator. The gold standard contains marked phrases according to predefined groups such as weapons, people, and vehicles. The results from the system are compared against this gold standard. Extracted Information should be evaluated by evaluation manager analyzing precision and recall. The system admin is responsible for all. He/She would appoint the evaluation manager and information extraction manager.

1.4 Scope

Extracting keywords is one of the most important tasks when working with text. Investigators get benefit from keywords because they can judge more quickly whether the report is worth reading. Keywords describe the main topics expressed in a document so it is of vital use in natural language processing, data mining, artificial intelligence, machine learning.As introduced earlier, keywords extraction is considered core for many text processing and information retrieval application.

1.5 Objective

The objective is to develop Keyword Extraction System that reduces efforts and time of police department in solving the crime cases quickly.

1.6 Organization of the Report

Chapter 1 presents the basic introduction to the Proposed System, Problem Statement, Problem Definition, Objective and Future Scope.

The system analysis is presented in second chapter, which describes Literature Survey, Existing System, Proposed System, Feasibility Study, Risk Analysis.

Chapter 3 describes the system requirements and specifications which includes Software Requirement, Functional Requirement, Non Functional requirement.

The system designing concepts are described in Chapter 4 including Proposed System Flow, Data Flow Diagrams, E-R Diagram, UML Diagrams.

1.7 Summary

This chapter describes Introduction. System Analysis is presented in next chapter.

Chapter 2

System Analysis

Analysis is a software engineering task that bridges the gap between system level requirements engineering and software design. Requirements engineering activities result in the specification of softwares operational characteristics (function, data, and behavior), indicate software's interface with other system elements, and establish constraints that software must meet. System analysis allows the software engineer (sometimes called analyst in this role) to refine the software allocation and build models of the data, functional, and behavioral domains that will be treated by software.

Section 2.1 presents Literature Survey. Proposed System is described in Section 2.2. Section 2.3 presents Feasibility Study. Risk Analysis is described in Section 2.4. Project Scheduling is presented in Section 2.5. Effort Allocation is described in Section 2.6. Summary is presented in Section 2.7.

2.1 Literature Survey

Information extraction stands for a wide range of techniques that are applied to automatically extract pre-specified elements. Most existing crime information extraction projects use narrative reports coming primarily from police departments.Details of several IE have been published in the recent past and these systems use different IE techniques. One such system is Kylin[4] , whose information extraction technique is based on classification and operates in two phases: identifying sentences in which interesting information is present and identifying words within sentences that carry the information. Kylin uses the Maximum Entropy model for the first phase and Conditional Random Fields (CRF) for the second phase. It uses a set of Wikipedia pages as its corpus and attempts to extract information presented by the "infoboxes, which provide a summary of the contents of each page. Kylin can also be considered an OBIE system because it constructs an ontology based on the structure of the infoboxes in order to aid its information extraction process.

Another information extraction technique used by many information extraction systems is known as “extraction rules. Here, the idea is to specify regular expressions that can be used to extract certain information from text. For example, the expression (belonged—belongs) to NP, where NP denotes a noun phrase, might capture the names of organizations in a set of news articles. This technique has been used by several IE systems including the ontoX[5] system and the implementations by Embley[6]. The Apache UIMA (Unstructured Information Management Architecture) project appears to be the most serious attempt so far to develop a component-based approach for information extraction. It targets analysis on all types of unstructured data including text, audio and video. It defines a common structure, known as Common Annotation Structure (CAS), to store the extracted information and provides frameworks in Java and C++ to deploy the developed components. In terms of analyzing text, UIMA components have been mostly developed for general NLP tasks such as sentence splitting, POS tagging and tokenization although some components have been developed for extracting instances of specific classes such as gene names. UIMA components do not separate the domain and corpus specific information from the underlying IE technique, which is a key idea in our component-based approach. Moreover, UIMA assumes that the developed components are interoperable or reusable whereas our approach studies the basis for successful reuse and presents methods to improve reusability. It is also interesting to note that UIMA uses UML models (through type systems) to relate the extracted information to domain models. In addition, Embley[6], Maedche et al. and Yildiz and Miksch[5] have independently worked on including extraction rules in ontologies to come up with what has been termed “extraction ontologies or “concrete ontologies. The general idea here is to include extraction rules related to a class in the ontology itself.

2.2 Proposed System

Using witness narrative reports obtained from online forums and blogs and police narrative reports from police departments and newspaper articles. Police investigators can obtain more valuable information from summarized narrative reports. Automatically extracting information, combining information from crime narrative reports, and presenting a meaningful summary for police investigators will help them quickly comprehend crime incidents without having to read an entire report. Evaluation is based on precision and recall.

2.3 Feasibility Study

The feasibility study is carried out to test whether the proposed system is worth being implemented. Feasibility study is a test of system proposed regarding its work ability, its impact on the organization ability to meet user needs and effective use of resources. The key consideration involve in the feasibility study are:

- Economical Feasibility
- Operational Feasibility
- Technical Feasibility

2.3.1 Economical Feasibility

This is the main factor in the feasibility study. When product is economically affordable then it can be used. So project must be cost saving. Establishing the cost effectiveness of the proposed system i.e. if the benefits do not outweighs the costs then it is not worth going ahead. The project involves the utilization of open source tools and softwares which indirectly decreases the release cost of system. The system is economically feasible.

2.3.2 Operational Feasibility

Operational feasibility is the ability to utilize, support and perform the necessary tasks of a system or program. It includes everyone who creates, operates or uses the system. The keywords extraction makes the operation easier for police investigators in solving crime incidents. Open source tools helps administrator to perform specified tasks. This makes the system operationally feasible.

2.3.3 Technical Feasibility

Technical feasibility centers on the existing computer system (hardware, software etc) and to what extent it can support the proposed system addition. The front end used in the system java. The platform used for developing the applications is Linux which is easily available. There is no more hardware required other than the personal system for its execution. The use of OpenNLP and General Architecture for Text Engineering(GATE), an open source softwares helps in carrying out Natural Language Processing tasks. Use of this tools make the system technically feasible and sound.

2.4 Risk Analysis

Risk analysis and management are a series of steps that help a software team to understand and manage uncertainty. Many problems can plague a software project. A risk is a potential problem might happen, it might not. But, regardless of the outcome, it is really a good idea to identify it, assess its probability of occurrence, estimate its impact, and establish a contingency plan should the problem actually occur. Everyone involved in the software process, managers, software engineers, and customers participate in risk analysis and management.

2.4.1 Software Risk

The Natural Language Processing tasks are interdependent. The output of first module is input to next module. If any one of the module fails to execute the whole process can be affected. Failure of any of the module while executing can give the wrong output.

2.4.2 Project Risk

Project risks threaten the project plan. That is, if project risks become real, it is likely that project schedule will slip and that costs will increase. Project risks identify potential budgetary, schedule, personnel (staffing and organization), resource, customer, and requirements problems and their impact on a software project. Project risk can occur if any one of member allocated is unavailable according to project plan and estimation. If project is not completed within time in this situation project risk can occurs.

2.4.3 Technical Risk

Threaten the quality and timeliness of the software to be produced. If a technical risk becomes a reality, implementation may become difficult or impossible. Technical risks identify potential design, implementation, interface, verification, and maintenance problems. In addition, specification ambiguity, technical uncertainty, technical obsolescence are also risk factors. Technical risks occur because the problem is harder to solve than thought it would be. If any module does not work properly according to expectation then technical risk may occur. The records if not maintained properly may affect the quality and accuracy.

2.5 Project Scheduling

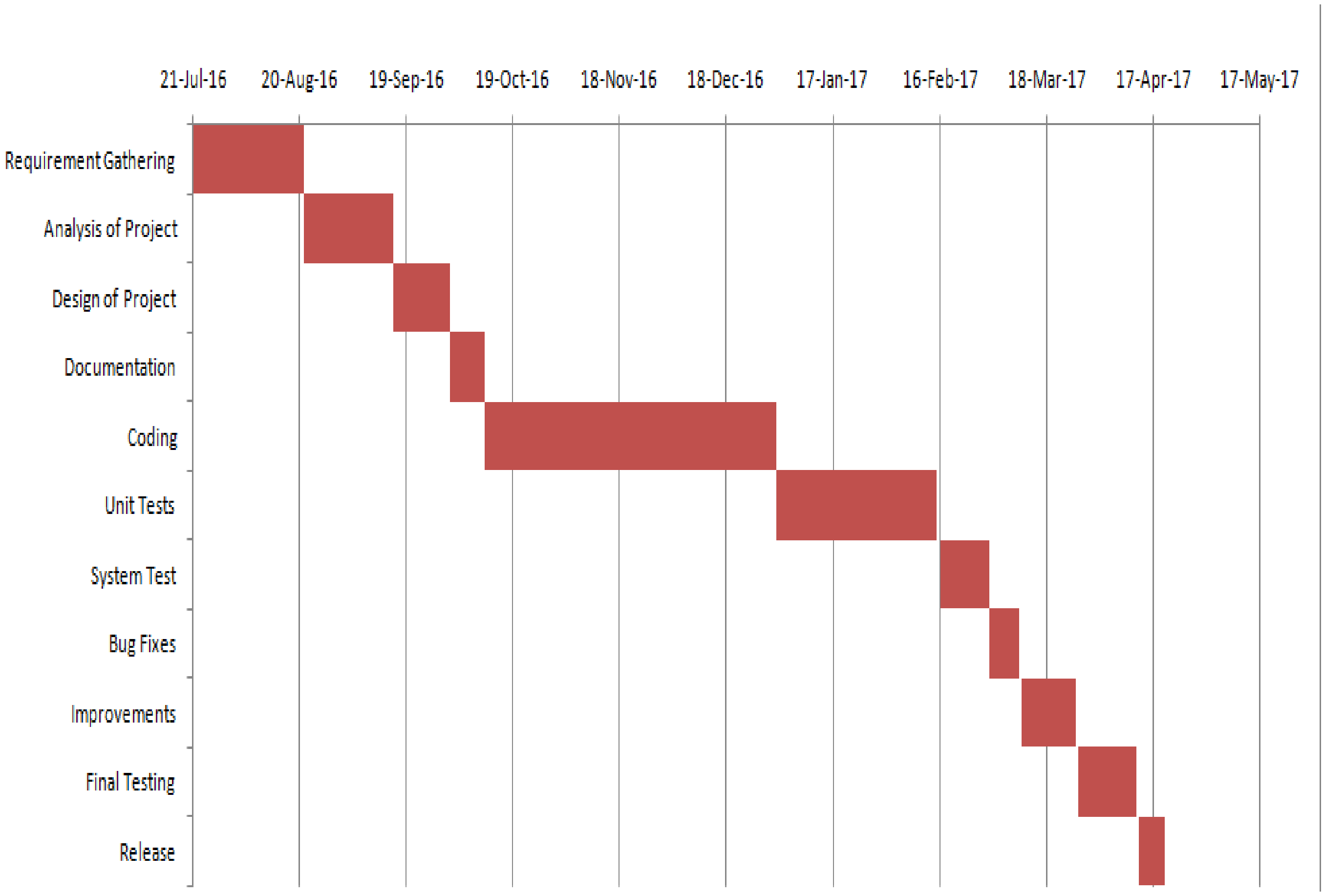
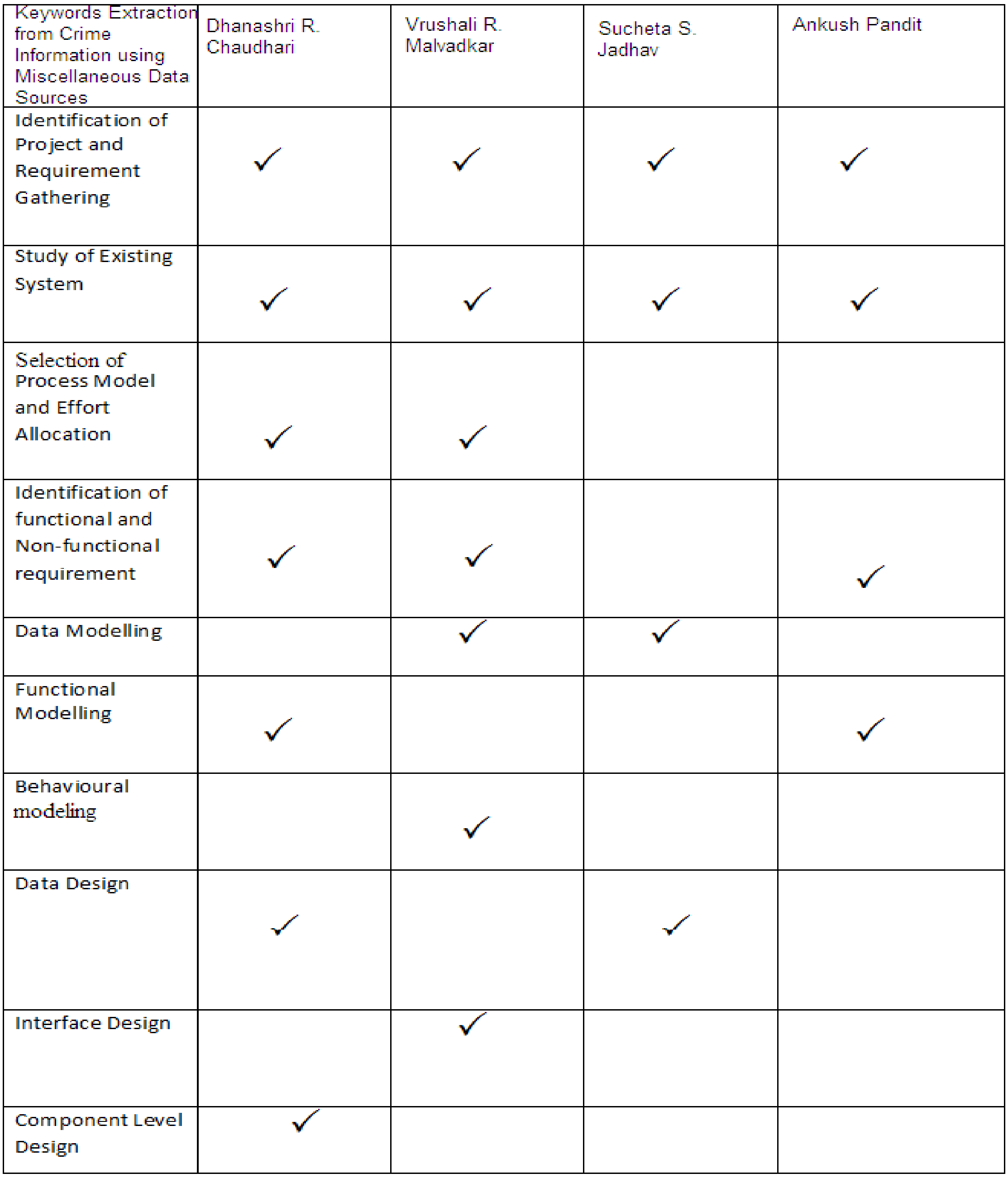


Figure 2.1: Gantt Chart

2.6 Effort Allocation

Project means team work. Project is developed by combination of effort of team. So whole project is divided into modules and number of modules is allotted to team members. After completion of each module, it will be link from one module to another module to form a complete project. This effort allocation is a guideline only. The characteristics of each project must dictate the distribution of effort.



Keywords Extraction from Crime Information using Miscellaneous Data Sources	Dhanashri R. Chaudhari	Vrushali R. Malvadkar	Sucheta S. Jadhav	Ankush Pandit
Identification of Project and Requirement Gathering	✓	✓	✓	✓
Study of Existing System	✓	✓	✓	✓
Selection of Process Model and Effort Allocation	✓	✓		
Identification of functional and Non-functional requirement	✓	✓		✓
Data Modelling		✓	✓	
Functional Modelling	✓			✓
Behavioural modeling		✓		
Data Design	✓		✓	
Interface Design		✓		
Component Level Design	✓			

Figure 2.2: Effort Allocation Table

2.7 Summary

This chapter describes the System Analysis. System Requirement Specification is presented in next chapter.

Chapter 3

System Requirement Specification

It provides requirements, needs of project and those things which help to complete project. System requirement describe a system from a technical perspective, which describe the essential characteristics of the hardware and software that will meet those needs. It should specify the capabilities, capacities and characteristics of the system in both qualitative and quantitative terms.

Section 3.1 presents Hardware Requirements. Software Requirements is described in Section 3.2. Summary is presented in Section 3.3.

3.1 Hardware Requirements

The hardware requirement includes a system with following configurations:

- Processor: Intel core i3/i5/i7
- Hard Disk: 50 GB
- RAM: 4GB RAM

3.2 Software Requirements

- Operating System: Linux
- Language: Java
- Tools:
  1. Jdk 1.8 for Linux
  2. Netbeans IDE for Linux
  3. Standard Lexicalized Parser V3.6.0

4. Ekit-Spell Checker
5. General Architecture for Text Engineering(GATE) / Open NLP
6. Java WordNet Library

3.3 Summary

This chapter describes System Requirement Specification. System design is presented in next chapter.

Chapter 4

System Design

System design provides the understanding and procedural details necessary for implementing the system. Design is an activity concerned with making major decisions, often of a structural nature. Design builds coherent, well planned representations of programs that concentrate on the interrelationships of parts at the higher level and the logical operations involved at the lower levels. Software design is the first of the three technical activities-designs, coding and test which are required to build and verify the software.

Section 4.1 presents System Architecture. E-R Diagram is presented in Section 4.2. Section 4.3 presents Database Design. Data Flow Diagram is presented in Section 4.4. Section 4.5 presents Summary.

4.1 System Architecture

The System Architecture provides the details of how the components or modules are integrated.[1]

Tokenizer: The tokenizer splits input text into tokens such as words, numbers, and symbols.

Sentence Splitter: This component separates an input text into sentences.

POS Tagger: This is a revised version of the Brill tagger. Every token is annotated with a POS (part-of-speech) tag, which is a grammatical tag such as noun, determiner, or adverb. There are 42 different possible tags as default in GATE.

Gazetteer: This component is mainly used to identify entities (e.g., act and event) and can also be used in rules to extract complex phrases.

Ortho-Matcher: This component recognizes specific names such as cities, streets, and states. These names may contain upper initials, all capitals, or all lower case letters.

JAPE Rule: JAPE (Java Annotations Pattern Engine) language is used for pattern matching and to add annotations to the matched patterns.

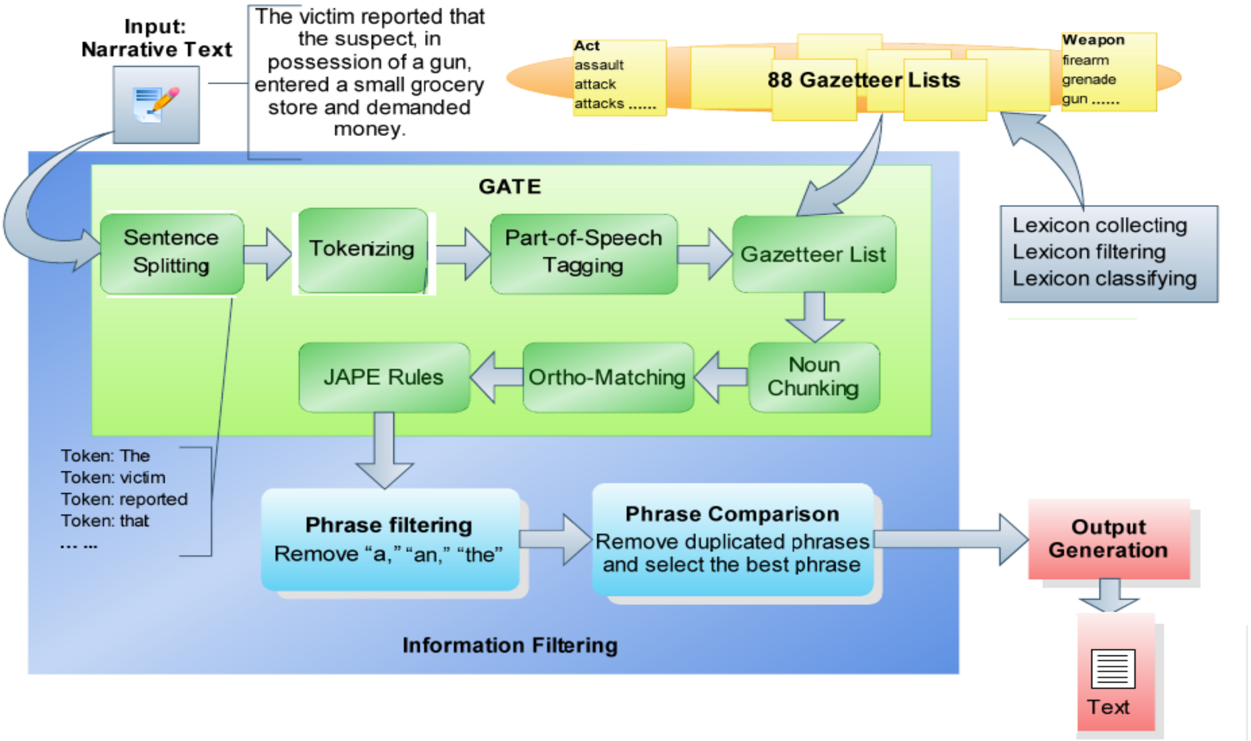


Figure 4.1: System Architecture

4.2 E-R Diagram

In software engineering, an entity relationship model (ER model) is a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a relational database. The main components of ER models are entities (things) and the relationships that can exist among them.

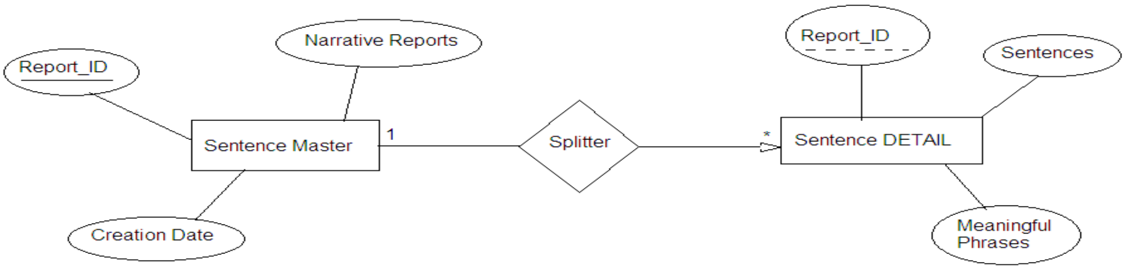


Figure 4.2: E-R Diagram

Figure 4.2 shows E-R Diagram. Sentence Master and Sentence Detail are two entities and Splitter relationship exists among them. Report ID, Narrative Text and Creation Date are attributes of Sentence Master in which Report ID is a primary key. Report ID, Sentences and Meaningful Phrases are attributes of Sentence Details.

4.3 Database Design

A database schema of a database system is its structure described in formal language supported by the database management system. The term “schema” refers to the organization of data as a blueprint of how the database is constructed(divided into the database tables in the case of relational databases).

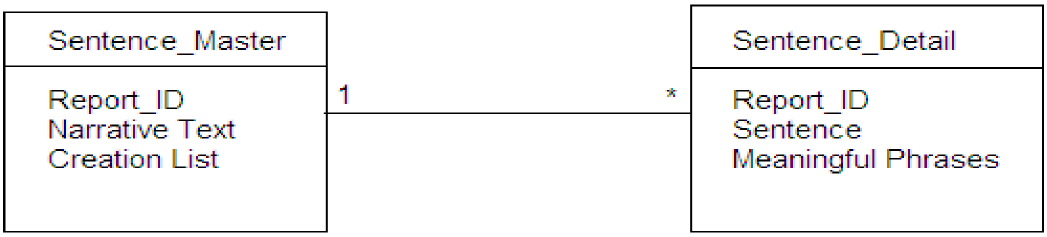


Figure 4.3: Database Schema

Figure 4.3 shows Database Schema. The database is divided into two tables-Sentence Master and Sentence Detail. Attributes of Sentence Master are Report ID(primary key), Narrative Text and Creation Date. Report ID(Foreign Key), Sentence and Meaningful Phrases are the attributes of Sentence Detail relation. Sentence Master and Sentence Detail relations are associated with one to many relationship.

4.4 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 will be input to and output from the system, where the data will come from and go to, and where the data will be stored.

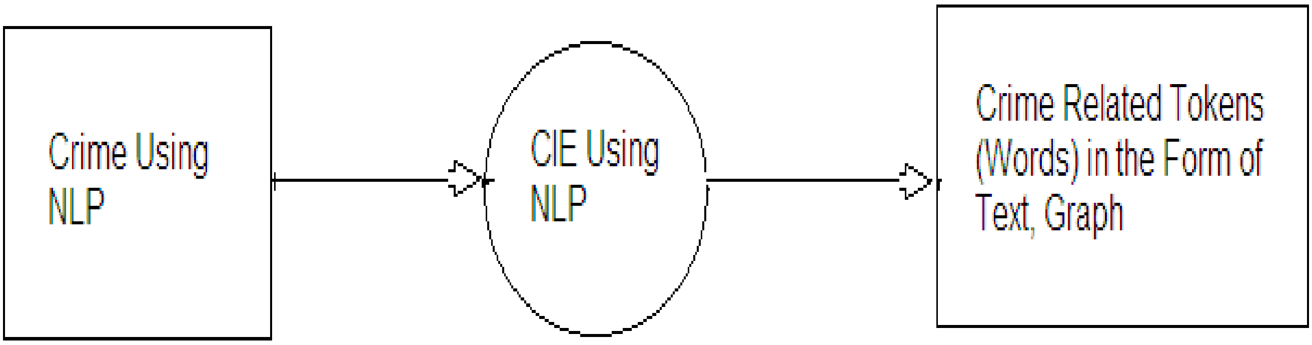


Figure 4.4: Data Flow Diagram(Level 0)

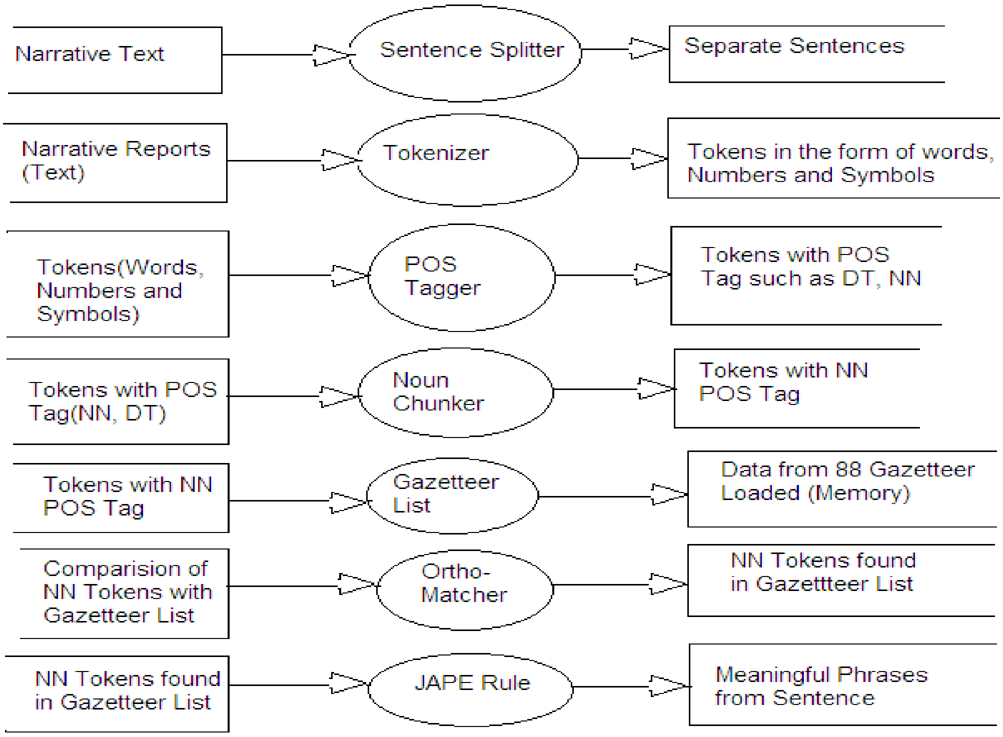

Narrative Text
Sentence Splitter
Separate Sentences
Narrative Reports (Text)
Tokenizer
Tokens in the form of words, Numbers and Symbols
Tokens(Words, Numbers and Symbols)
POS Tagger
Tokens with POS Tag such as DT, NN
Tokens with POS Tag(NN, DT)
Noun Chunker
Tokens with NN POS Tag
Tokens with NN POS Tag
Gazetteer List
Data from 88 Gazetteer Loaded (Memory)
Comparision of NN Tokens with Gazetteer List
Ortho-Matcher
NN Tokens found in Gazettteer List
NN Tokens found in Gazetteer List
JAPE Rule
Meaningful Phrases from Sentence

Figure 4.5: Data Flow Diagram(Level 1)

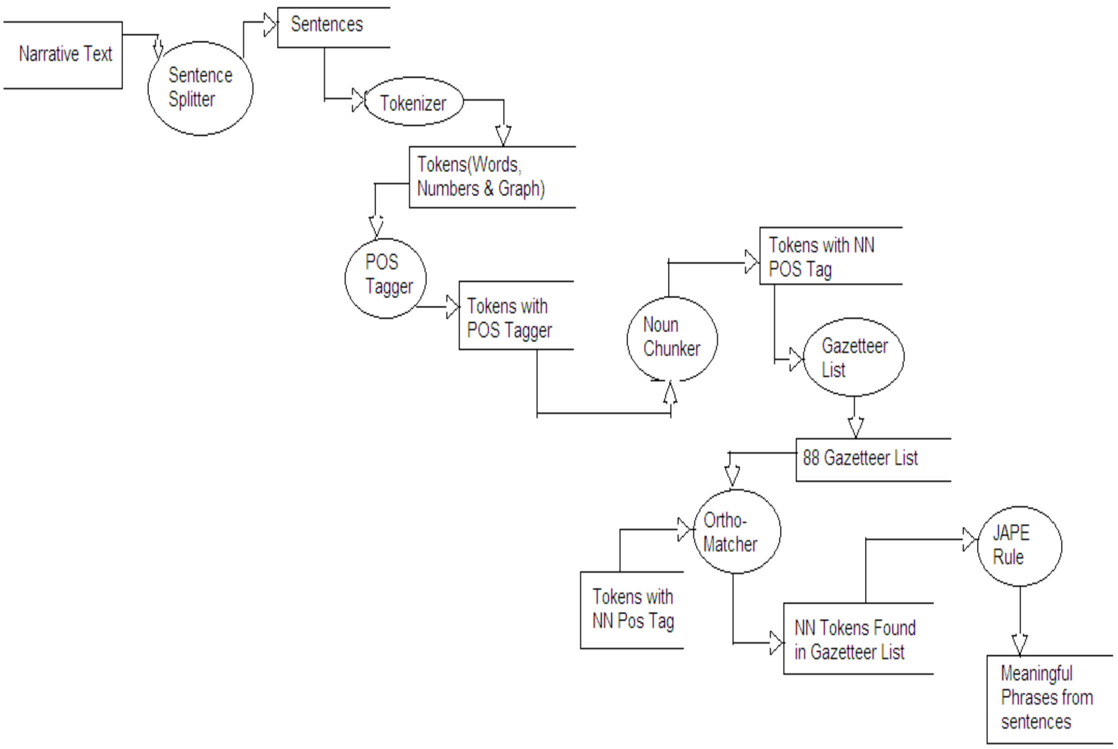


Figure 4.6: Data Flow Diagram(Level 2)

4.5 UML Diagrams

The UML is a language for: Visualizing, Specifying, Constructing and Documenting.

4.5.1 Use Case Diagram

Figure 4.7 shows use case diagram for System Administrator. It shows the actor like System Administrator. The use cases in Figure 4.7 are Collect witness narrative reports, collect police narrative reports, filter information, create gold standard, compare results, extract information, appoint evaluation manager and appoint information extraction manager. Actor is connected to use cases according to their role in system.

Figure 4.8 shows use case diagram for the information extraction management module. It shows the actors like Information Extraction Manager and System Administrator. The use cases in the Figure 4.8 are extract keywords and filter information. Actors are connected to use cases according to their role in system.

Figure 4.9 shows use case diagram for Evaluation Manager module. It shows the actor like Evaluation Manager. The use cases in Figure 4.9 are evaluation, analyze precision, analyze recall and calculate results. Actor is connected to use cases according to their role in system.

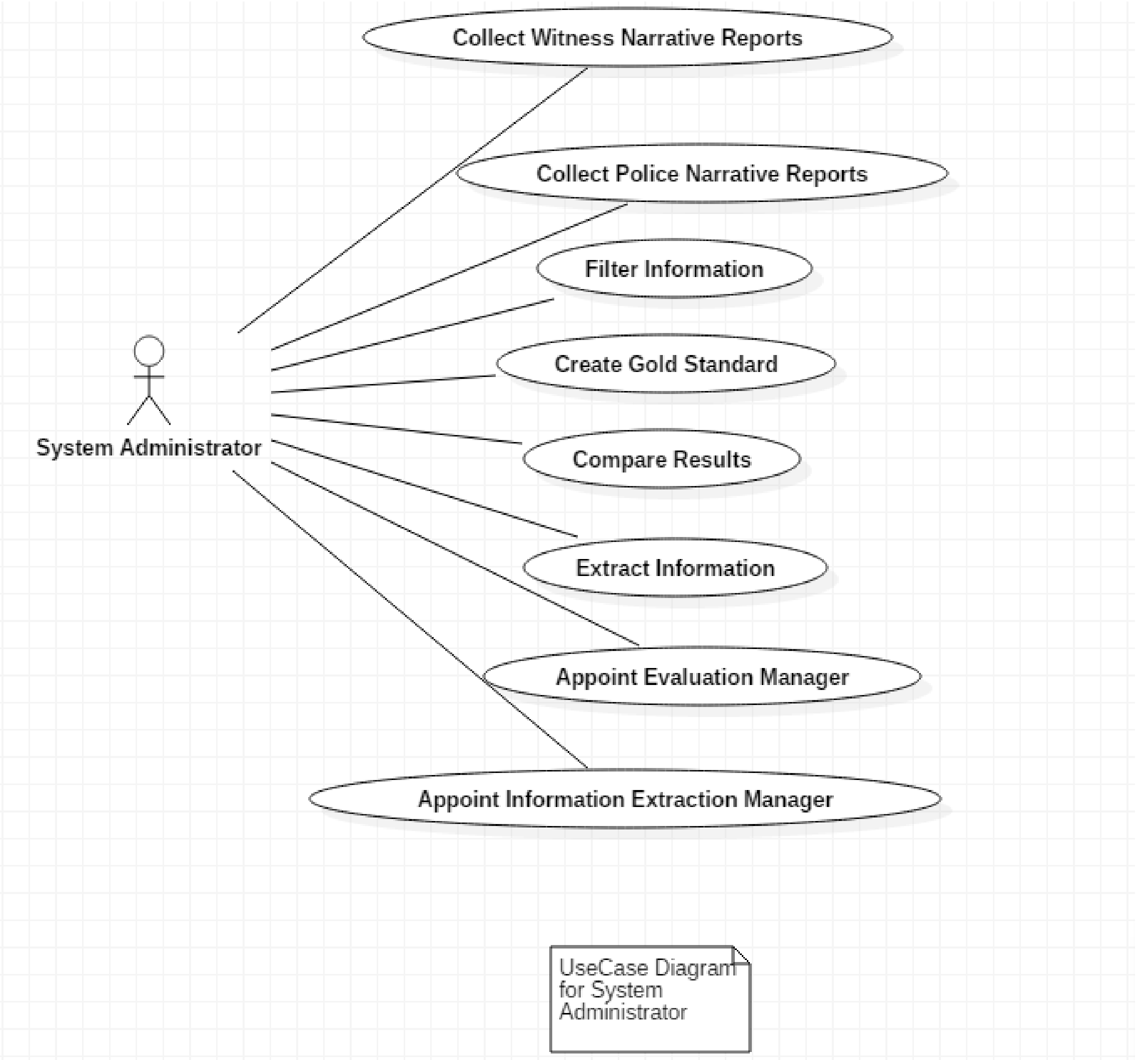

Collect Witness Narrative Reports
Collect Police Narrative Reports
Filter Information
Create Gold Standard
System Administrator
Compare Results
Extract Information
Appoint Evaluation Manager
Appoint Information Extraction Manager
UseCase Diagram for System Administrator

Figure 4.7: Use Case Diagram for System Administrator

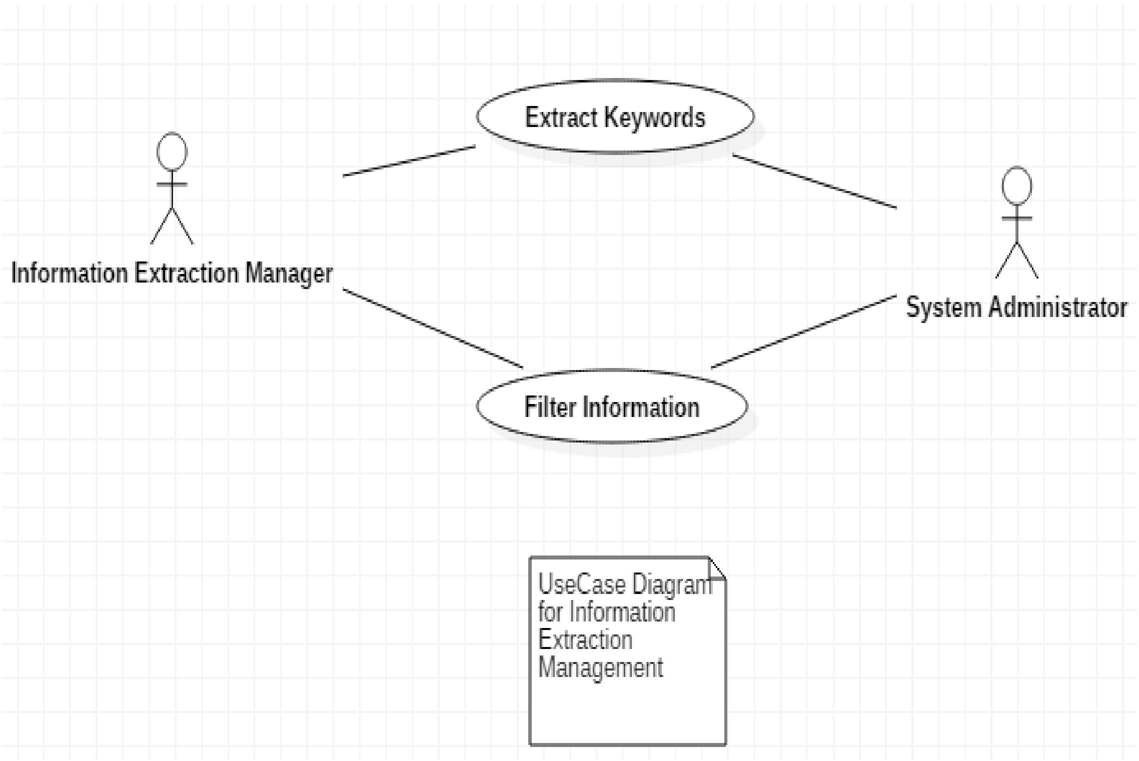

Extract Keywords
Information Extraction Manager
System Administrator
Filter Information
UseCase Diagram for Information Extraction Management

Figure 4.8: Use Case Diagram for Information Extraction Management

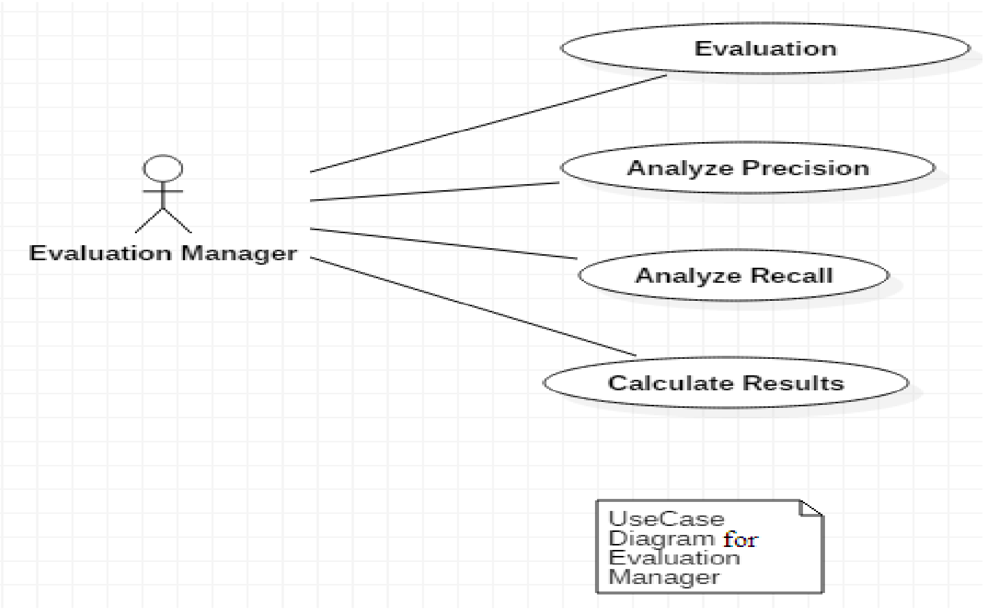


Figure 4.9: Use Case Diagram for Evaluation Management

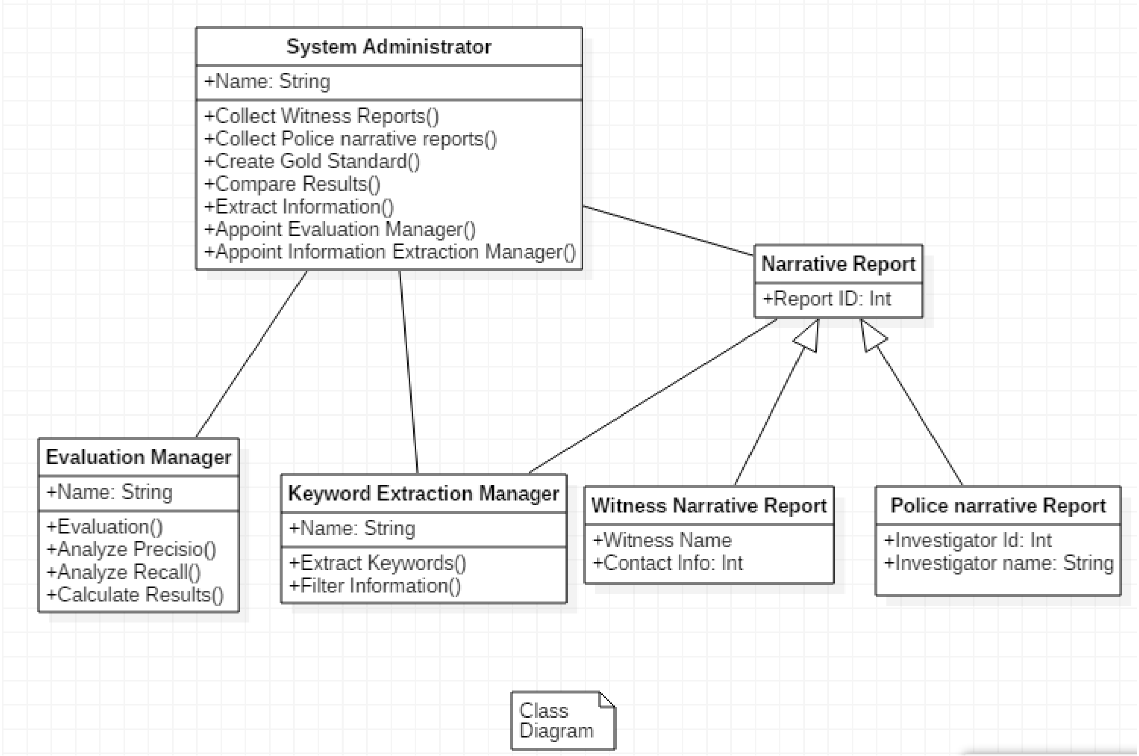


Figure 4.10: Class Diagram

4.5.2 Class Diagram

A Class diagram is used to represent the static view of the system. It mainly use Classes, interfaces and their relationships. Figure 4.10 shows Class Diagram for proposed system. System Administrator has the attributes like collect witness report, collect police narrative reports, create gold standard etc. Evaluation Manager has attributes like evaluation, analyse precision, analyze recall and calculate results. Keyword Extraction Manager has the attributes like extract keywords and filter information. Narrative Report is the parent class. Witness narrative reports and police narrative reports are the child classes.

4.5.3 Sequence Diagram

A Sequence diagram is a structured representation of behavior as a series of sequential steps over time. It is used to depict workflow, message passing and how element in general cooperate over time to achieve a result. In Figure 4.11, system administrator collect witness reports and collect police reports. System Administrator appoint IE manager to Information Extraction Manager and appoint evaluation manager to Evaluation Manager. System Administrator creates gold standard and information extraction manager extract the keywords. Then information extraction manager update data and send to system administrator.

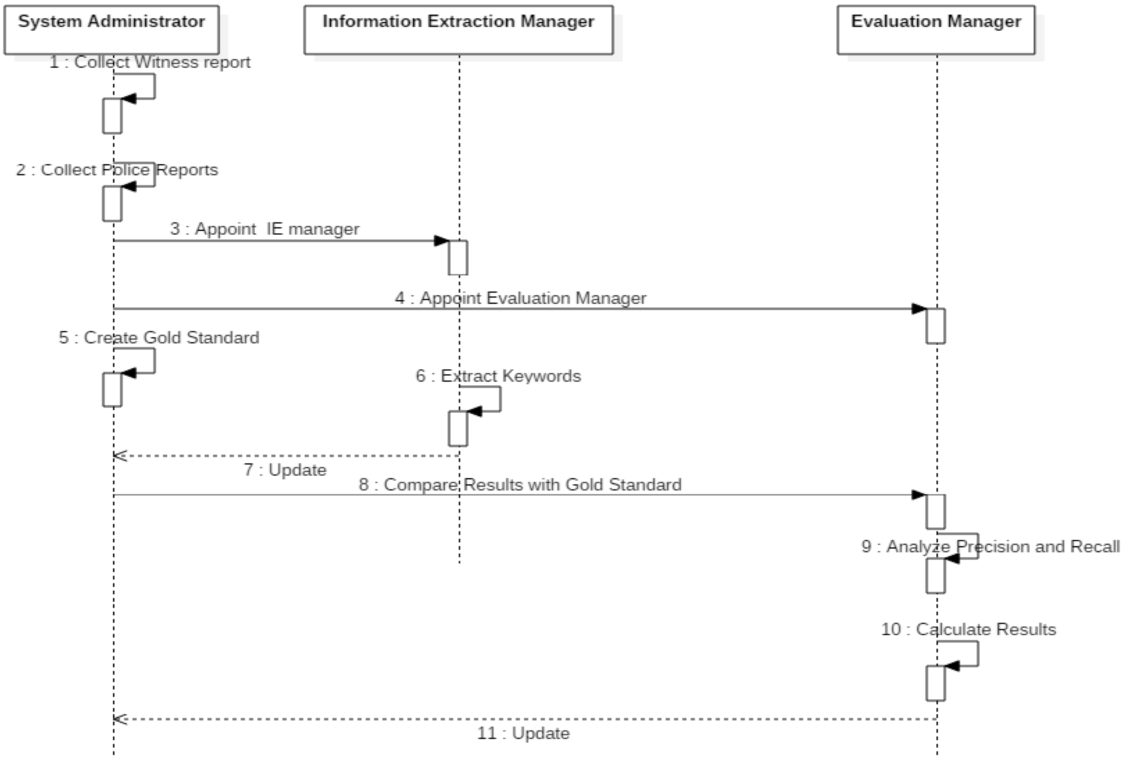

System Administrator
Information Extraction Manager
Evaluation Manager
1 : Collect Witness report
2 : Collect Police Reports
3 : Appoint IE manager
4 : Appoint Evaluation Manager
5 : Create Gold Standard
6 : Extract Keywords
7 : Update
8 : Compare Results with Gold Standard
9 : Analyze Precision and Recall
10 : Calculate Results
11 : Update

Figure 4.11: Sequence Diagram

System administrator compare the results with gold standard and send to evaluation manager. Evaluation manager analyze precision and recall then calculate the result. Evaluation manager send the update data to system administrator.

4.5.4 Activity Diagram

Figure 4.12 shows Activity Diagram. If the sufficient data source is not found then it returns to the first action collect narrative reports. If sufficient data is found then the next actions will take place like create gold standard, filter information, extract keywords, compare results. Fork and join are used.

4.5.5 Component Diagram

A component diagram-Figure 4.13 shows the organization and dependencies among set of components. These diagrams are used to model static view of the system. The components are various files such as executable files, database, class files. Dependency relationship is used to connect them.

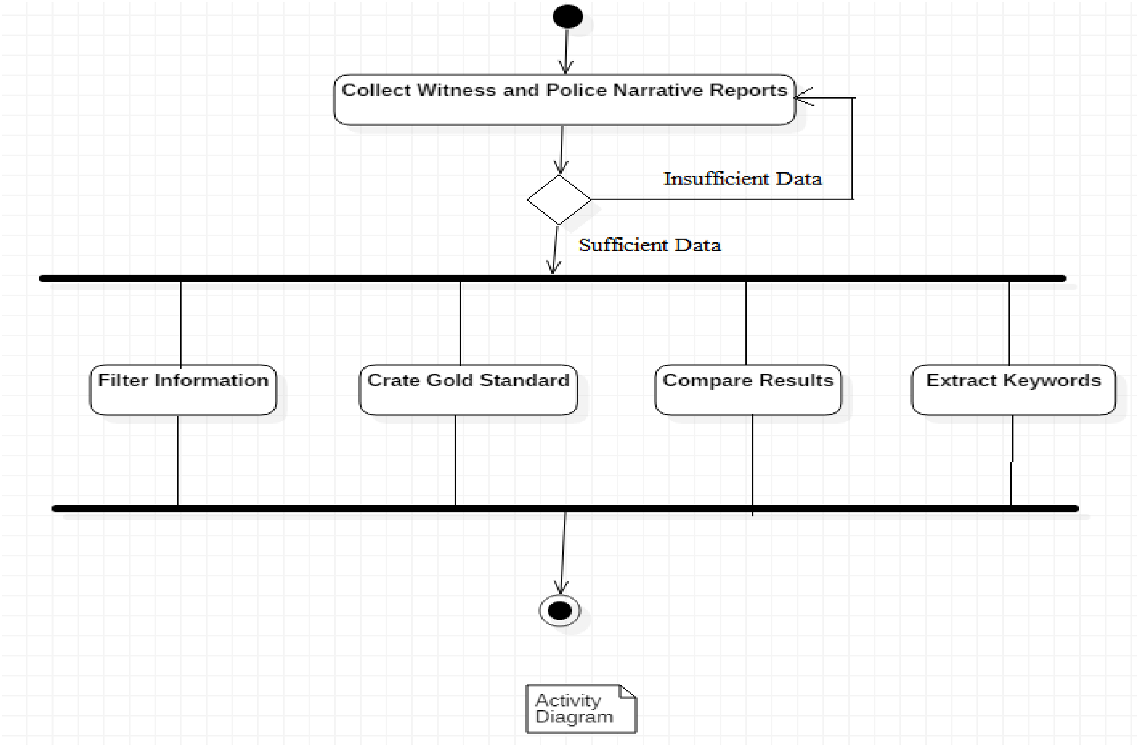


Figure 4.12: Activity Diagram

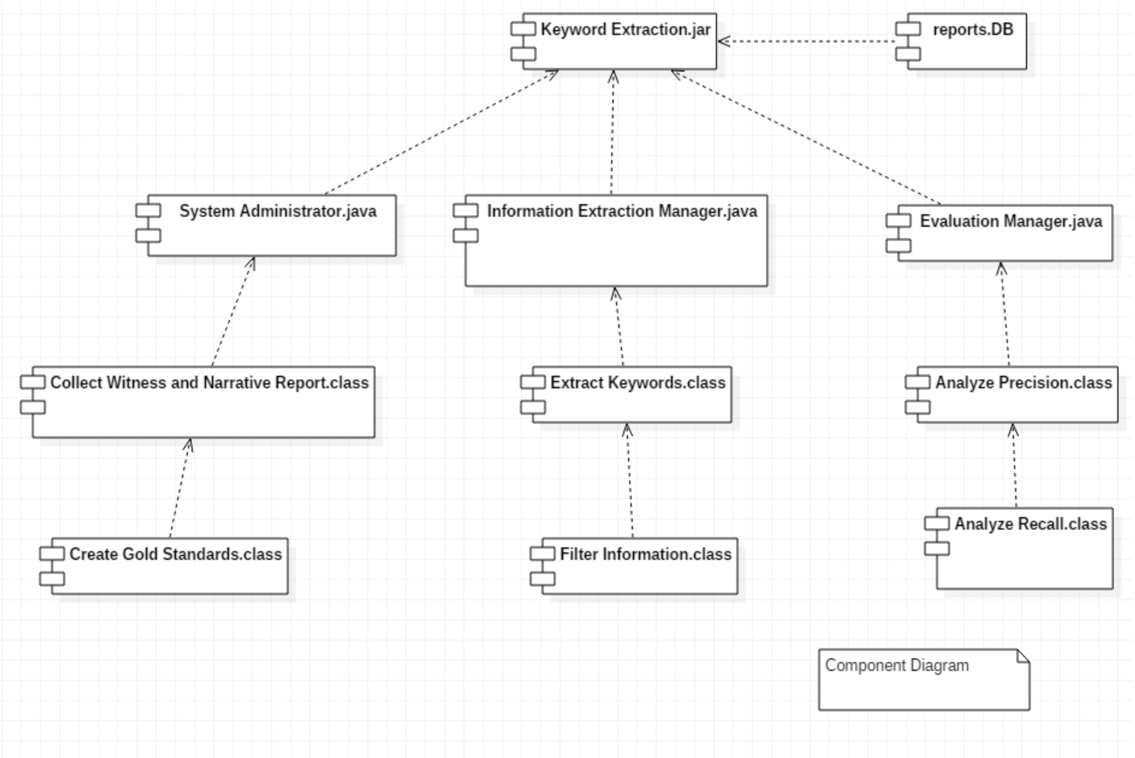


Figure 4.13: Component Diagram

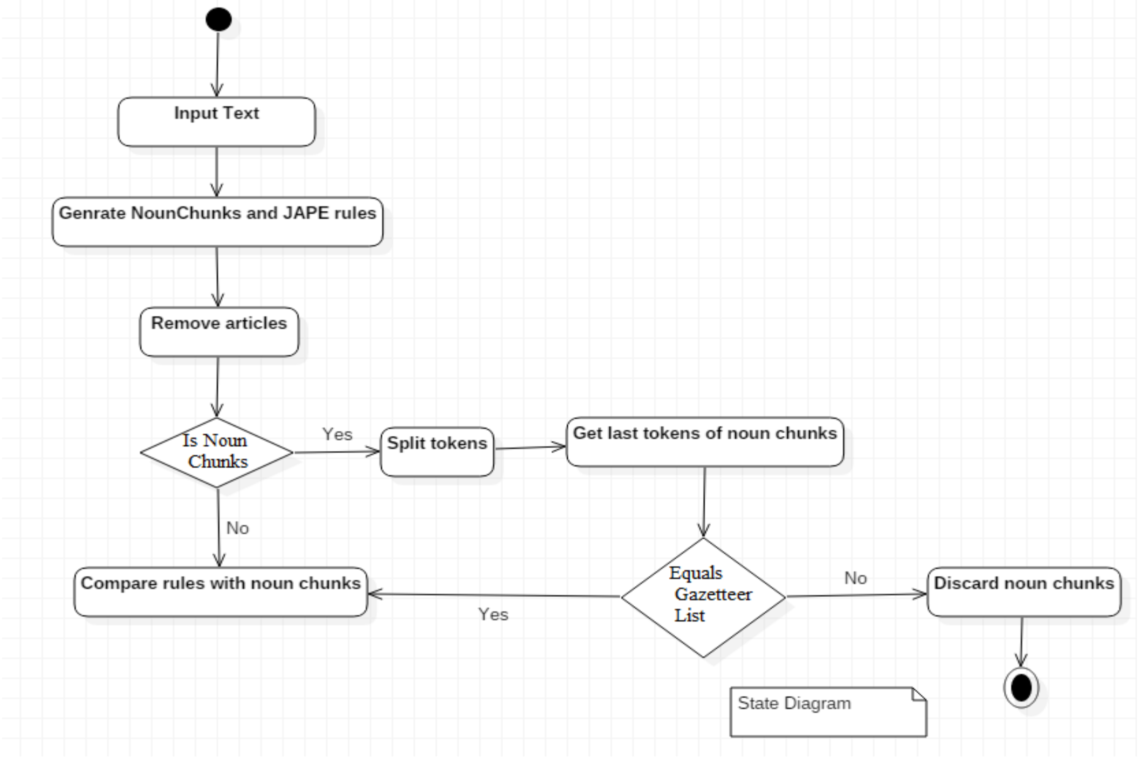


Figure 4.14: State Diagram

4.5.6 State Diagram

Figure 4.15 shows State Diagram. A state diagram shows a state machine, consisting of states, transitions, events, and activities. State diagrams address the dynamic view of our system. It checks condition of the input text is having noun chunks or not.If noun chunks found then there is comparison with the gazetteer list. If it does not matches then it is discarded otherwise compare rules with noun chunks.

4.6 Summary

This chapter describes System Design.

Bibliography

[1] C. H. Ku, A. Iriberri et al., ”Natural Language Processing and e-Government: Crime Information Extraction from Heterogeneous Data Sources,” Ninth International Conference on Digital Government Research, 2008.

[2] S. V. Nath, “Crime pattern detection using data mining,” IEEE/WIC/ACM International Conference, pp. 41- 44, 2006.

[3] D. Freitag, “Machine learning for information extraction in informal domains,” Machine Learning, vol. 39, no. 4, pp. 169- 202, 2000.

[4] F. Wu, R. Hoffmann, et al., “ Information extraction from wikipedia: moving down the long tail”, In KDD, 2008, pages 731-739.

[5] B. Yildiz and S. Miksch. “a method for ontology-driven information extraction”, In ICCSA (3), 2007, pages 660673.

[6] D. W. Embley, “Toward semantic understanding: an approach based on information extraction ontologies”, In ADC, 2004, pages 312.