ANSER

CS09-805(P) B.Tech Main Project 2017

Done By

AMRUTHA M U - ETANECS008

ANUPAMA V P - ETANECS014

DEEPANJALI T R - ETANECS024

SREELAL K M - ETANECS054

Guided By

ANISH ABRAHAM

Assistant Professor

Dept. of Computer Science And Engineering
Government Engineering College
Thrissur-680009

ABSTRACT

Some of the busiest people on our planet are also avid readers, mostly news readers. News from sources like newspaper and Internet helps to improve reading habits, knowledge, and awareness. Most importantly, everyone likes to keep up on current events, but keeping up to date on current events is basically a full time job. Thick newspapers can be intimidating. While you want to read the full paper to stay caught up with current events, you may wonder if you have the time. In today's world, time is of the essence and definitely no one has time to read every news story every day. When the source is Internet, RSS readers are great for people who have the time to go through them, but they're not that good for just getting a summary of world news. Besides these, the mindset and opinions of these newsreaders are moulded by the content which is present in the news source. Many of the existing news sources are biased. Therefore, it is important that a new source presents stories which are balanced or unbiased.

Our system has been proposed exactly keeping in mind all of these facts. Our goal is to create a software which presents concise, balanced and always updating news summaries so that you can keep yourself up to date without spending time digging into it. We achieve this by extracting news stories from sources freely available in the Internet and using text summarization algorithms to condense these stories. These algorithms produces coherent summaries of multiple documents in text format to generate an indicative, less redundant summary. The resulting summary is presented to the user in an elegant and user-friendly interface. Users also have the option to go through entire news article if they want to get a detailed description.

ACKNOWLEDGEMENT

First and foremost, we thank the almighty for having bestowed his blessings on us to take up this project and to complete it successfully.

We express our sincere gratitude to our respected HOD of Computer Science Department Prof. Sminesh C N. We also thank our project coordinator Asst. Prof. Ajay James , our guide Asst. Prof. Anish Abraham and all other faculty members of the Computer Science and Engineering Department for their help towards the project. Especially we would like to thank our guide for his moral support and patience while suggesting the outlines of this project and correcting our doubts. We thank him for his overall support. Last but not the least we are also thankful to all our friends who supported us and helped us in doing this project.

Contents

List of Tables

List of Figures

Nomenclature

API	Application programming interface
ER	Entity Relationship
GUI	Graphical User Interface
IEEE	Institute of Electrical and Electronics Engineers
POS	Part-Of-Speech
UML	Unified Modelling Language

Chapter 1

Introduction

1.1 Problem Statement

The aim of this project is to create a system to present concise, balanced and always updating news summaries using summarization techniques. This helps the users to save time and get precise news stories at a glance.

The motive behind our project arises from the absence of a news reading system which can save time and present stories which are unbiased. The existing system involves traditional newspapers and news articles in the Internet provided in depth. Even though summarization techniques exists, problems like inaccurate extraction to essential sentences, low coverage and poor coherence among the sentences are present. We improve this system by using clustering technique for multi-document summarization inorder to produce summaries effectively eliminating the above mentioned problems. News stories are obtained from various freely available sources in the Internet.

1.2 Feasibility Study

1.2.1 Technical Feasibility

We have analysed the technical feasibility of the project and it is feasible based on following factors.

1.2.1.1 Hardware Feasibility

The minimum hardware requirements for developing this software are given below:

- Personal computer with internet connection
- Android smartphone with internet connection

The above hardware requirements are found to be available and feasible for our project work.

1.2.1.2 Software Feasibility

The different languages that are available for developing the application were analyzed. Based on the capabilities and ease of implementation, the following were chosen:

- Android Studio
- Ubuntu/ Windows

Both the hardware and software requirements are feasible from an implementation point of view.

1.2.2 Financial Feasibility

1.2.2.1 Development Cost

For developing an application no particular cost is required. Cost for human eort is the only expense.

1.2.2.2 Installation Cost

No particular installation cost is needed other than the cost of hardwares.

1.2.2.3 Operational Cost

Execution of the application does not actually require any operational cost. The only operational cost required is the cost of power supplies to hardware devices as well as the connectivity charges.

1.2.2.4 Maintenance Cost

No particular maintenance cost is needed for this software.

1.2.3 Operational Feasibility

Execution of our system does not actually require any operational cost. The only operational cost required is the cost of power supplies to the hardware devices.

1.2.4 Schedule Feasibility

We believe that with the available technical expertise and infrastructure, the project can be completed within the scheduled time frame. We shall remain in constant touch with our guide on a weekly basis to ensure that we are proceeding in the right direction and finish the work within the time guidelines.

1.3 Process Model Selection

After having a detailed study on all the five software development models we have come to the conclusion that the Iterative Model is most suitable for our project.

1.3.1 Model Description - Iterative Model

Iterative model creates a high-level design of the application before we actually begin to build the product and define the design solution for the entire product. As there is a working model of the system at a very early stage of development which makes it easier to find functional or design flaws. Finding issues at an early stage of development enables to take corrective measures in a limited budget.

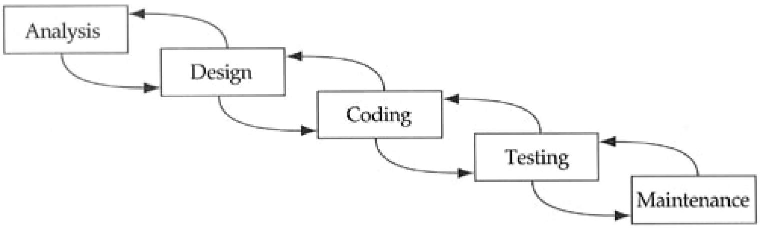

Figure 1.1: Iterative Model

Iterative Model Solves Issues Like:

- Resource wastage
- Costly modifications
- Unclear requirements

Why Iterative Model?

- Supports changing requirements
- Better suited for large and mission-critical projects
- Results are obtained early and periodically
- Easier to manage risk - High risk part is done first
- Testing and debugging during smaller iteration is easy

Chapter 2

Requirement Analysis

Requirements analysis and validation is a process of refinement, modelling and specification of the already discovered user requirements. The systematic use of proven principles, techniques, languages, and tools for the cost effective analysis, documentation, and ongoing evolution of user needs and the specification of the external behaviour of a system to satisfy those user needs forms an integral part of this stage of software development. This chapter describes the method of requirement elicitation employed and the user requirements thus gathered. The requirements are also finalized after validation using a suitable method.

2.1 Method of requirement elicitation

Inorder to analyze and predict stock value effectively, we adopted the stake-holder based technique.

2.2 Approach

The steps involved in the approach for elicitation include:

2.2.1 Identifying the Stakeholders

A stakeholder is any person or organization who can be positively or negatively impacted by, or cause an impact on the actions of a company, government or organization.
In this project, stakeholders are the people who have a say in the project and they are responsible for the acceptance of the project

The stakeholders here are:

- Development and testing team
- Users

2.2.2 Setting the goals

Modeling goals expresses the relationships between a system and its environment. This understanding gives, of the reasons why a system is needed, in its context. This basically describes the current scenario relevant to the domain of the project. The questions given below will help the development team in identifying the problem and coming up with an appropriate solution.

- What problem does this project solve?
- What are the benefits gained by adopting this solution?
- What are the risk factors to be taken into consideration?
- How should the problem be solved?
- What type of user interaction is required?
- Hardware and software availability
- Technical skills required
- Acceptance of the product

2.2.3 Elicitation technique

The techniques we use for gathering requirements from users are given below.

- Questionnaire
- Interview

2.2.3.1 Questionnaire

A set of questions were prepared for taking feedback from different stakeholders. Some of the questions from the questionnaire are given below:

- Have you ever read news on different sites to get the correct information about a particular incident?
- Do you find its difficult to read large news articles?
- Have you ever gone through an entire news article just to search for important information?
- Do you think this is a solvable problem?
- Have you ever wished to have a software that returns a summary of news from various websites?
- Do you think such a software can save time?
- What are your expectations from such software?

2.2.4 Inference

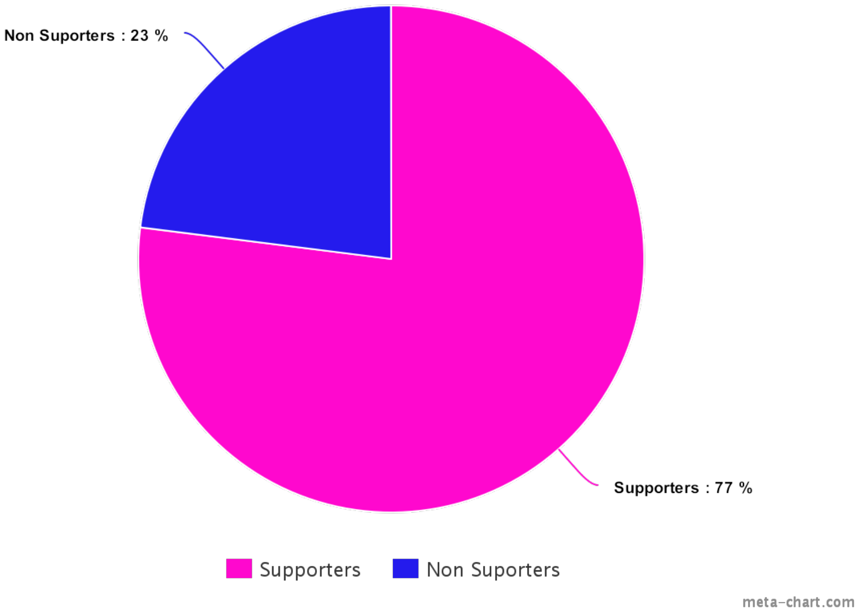

Figure 2.1: Survey Report

2.3 User requirements

On the basis of discussion with users for understanding the difficulties with the existing techniques, an overwhelming 85 percent of respondents claimed that it is difficult to read large news articles and to get the correct information about a particular incident. Approximately two thirds of respondents wished to have a software that returns a summary of news from various websites and reported that such a software can save time. Here are some of our findings:

- Users would appreciate a software that produces a summary of news from various websites.
- People who want to search for particular important information would like to save the time of searching through large news documents through some means.
- Users prefer to go through a condensed version of various news sources rather than reading each one of those.

2.4 Project requirements

On the basis of the requirements demanded by the user the following project requirements are found out:

- Ability to extract data from various news sources through web.
- Ability to find out common news items present in various news sources.
- Ability to combine these sources and to produce a condensed summary for these.
- Ability to present the summary with necessary information through a clean interface in a user friendly manner.

2.5 Requirements validation

The requirement validation ensures that the software being developed will satisfy the needs of its stakeholders. Requirement validation checks the software requirements specification against stakeholders goals and requirements. The typical requirement validation approaches are : Tracing approaches, prototyping, testing, user manual writing, formal validation, reviews and inspections, walkthroughs, checklists etc.

For the requirement validation of our project, we choose the reviews and inspection procedure. In the procedure, a group of people read and analyze the requirements, look for problems, meet and discuss the problems and agree on actions to address these problems. Careful planning and preparation are necessary for completing the validation procedure.

A formal review and inspection pattern was followed to complete the validation. Roles were assigned to each of the member participated in the review and they prepared for the review through studying the requirement specification document.

2.5.1 The Review Process

The review process includes six basic steps:

- Plan review
- Distribute documents
- Prepare for review
- Hold review meeting
- Follow-up actions
- Revise document

2.5.1.1 Plan Review

The review team consisting of the project members and stakeholders were selected. The time and place for the review meeting was decided. The agenda of the meeting was discussed with the team members.

2.5.1.2 Distributed Documents

The project documents on the abstract and the requirement specification were distributed among the team members for analysis and study.

2.5.1.3 Prepare for Review

The stakeholders studied the specified requirements. Their goals and needs were compared with the requirements specified in the document. The deviations, conflicts and omissions in the document were identified. The project members reviewed the specification, planned for possible changes and alternatives.

2.5.1.4 Hold Review Meeting

The review meeting was conducted in the scheduled place and time. The stakeholders came up with their suggestions. The suggestions were kept track by maintaining proper checklists with each reviewer.

2.5.1.5 Follow-up Actions

A final decision was made out of the meeting to implement the suggested features that are acceptable to every stakeholder.

2.5.1.6 Revise Document

The requirement specification document is revised to reflect the changes accepted in the meeting.

2.6 Software Requirements Specification

The software requirements specified in this chapter are applicable for the development of an application. This chapter encompasses those tasks that go into determining the needs or conditions to meet for the application being developed, taking into account the various limitations.

2.6.1 Document Purpose

The main purpose of this document is to provide detailed specification of all the requirements that are considered to be the goals of the project development. Further this document contains the specification regarding the interface requirements, supporting system features and the functional and nonfunctional requirements. The requirements serves as the reference during the system design and the entire system development.

2.6.2 Product Scope

Currently some applications and other softwares are available in the internet which the user can use to get summarized news.Even Though almost all of these systems are very good at avoiding redundant sentences, they have many limitations such as inaccurate extraction to essential sentences, low coverage and poor coherence among the sentences. The summary of this problem is that there is not a real time, software tool that enables the user to get news summaries exactly the way a reader needs. Thus the final goal of this project is to come up with a mechanism to develop a mobile appliation that assists the user to get an accurate,informative and coherent summary of news from multiple news articles using an effective user interface with the help of web scraping and machine learning.We introduce a new concept of timestamp approach for multi document news summarization.Timestamp provides the summary an ordered look,which achieves the coherent looking summary.This software extracts more relevant information from multiple articles.

2.6.3 Intended Audience and Document Overview

This document is intended to those who want to read news in minimal time. The intended stakeholders of the system are common people including students,working people,elders etc. This project enables them to get an accurate,informative,unbiased and coherent summary of news from multiple news articles.Aim of our software is to save time of the users and to provide an effect of reading multiple news papers.

2.6.4 Definition

- Automatic Text Summarization : It is the process of reducing a text document with a computer program in order to create a summary that retains the most important points of the original document. Technologies that can make a coherent summary take into account variables such as length, writing style and syntax. Automatic data summarization is part of machine learning and data mining. The main idea of summarization is to find a representative subset of the data, which contains the information of the entire set. Summarization technologies are used in a large number of sectors in industry today.

- Multi Document Summarization : It is an automatic procedure aimed at extraction of information from multiple texts written about the same topic. Resulting summary report allows individual users, such as professional information consumers, to quickly familiarize themselves with information contained in a large cluster of documents.

- Extractive Summarization : It is basically creating a summary based on strictly what you get in the text. It can be compared to copying down the main points of a text without any modification to those points and rearranging the order of that points and the grammar to make more sense out of the summary.

- API: Application program interface (API) is a set of routines, protocols, and tools for building software applications. An API specifies how software components should interact and APIs are used when programming graphical user interface (GUI) components. A good API makes it easier to develop a program by providing all the building blocks. A programmer then puts the blocks together.

2.6.5 Document Conventions

This document follows IEEE standard format and the conventions followed for fonts, formatting and naming are the standards followed in Computer Science and Engineering Department of Government Engineering College Thrissur.

2.6.6 Overall Description

This deals with general factors considering the proposed platform, the connection between system components, communication methods and the requirements.

2.6.7 Product Perspective

There are several applications for providing summarized news in the market now. But all these applications doesnt provide an accurate and coherent results.The advantage of our system is that it assists the user to get an accurate,informative and coherent summary of news from multiple news articles exactly the way they needs. Thus the final goal of this project is to come up with a mechanism to develop a mobile application that assists the user to get an accurate,informative and coherent summary of news from multiple news articles.

2.6.8 Product Functionality

The features offered by the application are as given below:

- Provide an effective GUI.
- Assists the user to get an accurate,informative and coherent summary of news from multiple news articles exactly the way they needs.
- Helps the user to save their time.

2.6.9 Users and Characteristics

By user classes and characteristics we are broadly defining the users who need frequent use of this product and the various requirements of each particular user classes.

- The main stream of users are the people who are interested in reading newspapers and having very small amount of free time. They can use this application to get a better results.

2.6.10 Operating Environment

The operational environment should satisfy the minimum hardware and software requirements.

- The system requires an Android smartphone with Android 4.0 or later and Internet connection.
- The system is developed by using Android studio.

2.6.11 Design and Implementation Constraints

The issues that will limit the options available to the developers are:

- Non availability of news sources.
- Non availability of a system with the minimum requirements.
- The entire process of the project should be completed by April 2017.

2.6.12 User Documentation

User manuals or online help will be provided for the users to install and use the application and for troubleshooting. The user manual should describe the steps to be followed for installation and the possible errors that can occur during the application run-time. It should also specify the cases where there is a possibility of an error.

2.6.13 Specific Requirements

2.6.13.1 External Interface Requirements

Hardware Interfaces

The only hardware requirement for implementing the system is a personal computer

Software Interfaces

The software interfaces required are:

- Android 4.0 or later.

User Interfaces

The system provides a GUI. It consists of an interface that provides the user with an option to input the compression rate.User will be then provided with effective news summaries based on the compression rate.

2.6.13.2 Functional Requirements

Functional requirements will analyze the services provided to the consumers and company by the application. Function of this application is to create an application that will enable the users to get news summaries.
It has the following phases:

Collecting news from different well known websites:

It can be done using free APIs available on the internet.

Process words:

This phase includes classification of the words into different categories. This is done by assigning ranks or score to each of the words.

Process sentences:

Here we identifies key sentences,Which are to be included in the resulting summary. A timestamp is also incorporated with each sentences to keep ordering or coherency.

Applying the solution algorithm:

Here the algorithm is implemented to generate the news summary.

2.6.13.3 Behaviour Requirements

Use Case View

A use case diagram at its simplest is a 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. A use case diagram can identify the different types of users of a system and the different use cases. A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements.

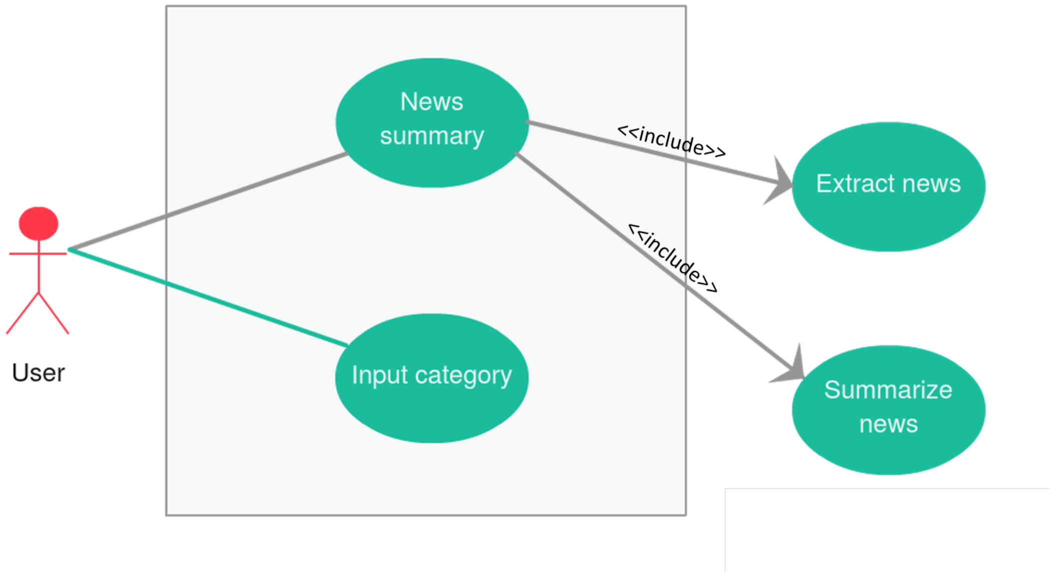

Figure 2.2: Use case view

Other Non-functional Requirements

Non-functional requirements are the constraints or limitations under which the system should provide its services to users. Following are the non-functional requirements of our project:

2.6.14 Performance Requirements

The main performance requirements that the product should satisfy are:

- Speed : The system must be able to generate an effective summary of news in a very short amount of time, such that the user can save their time.
- Resource usage:- The application should not use much system resources and degrade the system performance.

2.6.15 Software Quality Attributes

The most important quality requirements that the system should satisfy are

2.6.15.1 Scalability

The system should have scalability property.The application should be able to accommodate further addition of features.

2.6.15.2 Maintainability

The system must be maintainable.Sometimes there might be bugs present in the application.It must be easy to correct when it is reported.

2.6.15.3 Adaptability

The application must have the ability to adapt to the modifications that the server application gets with the error reporting with little or no modification.

2.6.15.4 Reliability

The product must be reliable. No change should go unnoticed. Similarly it must avoid faulty predictions.

2.6.15.5 Testability

The product must be properly tested under various circumstances in order to assure its reliability.

Chapter 3

Design

This chapter will present the design of the proposed system. The design of a system requires extra attention, especially because these systems may deal with complex algorithms of information extraction, natural language processing combined with heavy text processing (which may consume lots of computational resources) while still suffering from structural problems. For instance, communication problems of any nature between the system and the Internet, would severely affect the subsystem in charge of feeding the local database with opinions. Therefore, non-functional requirements play an important role to assist the system to accomplish its task.

The purpose of the software design document is to provide a detailed design of all the entities participating in the software.The project is divided into a number of modules and their integration provides the nal prediction mechanism. By designing the individual modules we can predict the feasibility and implementation cost of the application. The most important purpose of this document is to check whether the design satisfies the requirements specified in the SRS document.

3.1 Overall Design

3.1.1 System Design

The design of the system is described by the following sections:

- Architectural Design
- Decomposition Description
- Design Rationale

3.1.2 Architectural Design

The architectural design is a high-level overview of how the responsibilities of the system were partitioned and then assigned to sub-systems.The working of the system can be explained by combining the functionalities of the sub modules. The sub-modules of the system are:

- Pre-processing module: The pre-processing involves collection of relevant documents.The documents collected are tagged using a POS tagger, namely a tree tagger.
- Hierarchical clustering module: This is done using semantic cosine similarity. Cosine similarity is a technique to find out the similarity between pairs of sentences in a document. Using the cosine similarity values, the sentences in the document are clustered.
- Topic and subtopic identification module: A fuzzy algorithm is used for identifying the topic and subtopic of each cluster generated by hierarchical clustering.
- Sentence ranking module: The sentences inside each cluster have to be scored for their relevance to identify the most important sentences in the document.For this each sentence is scored based on different metrics such as length, location, etc.
- Lexical chaining module: The required number of sentences is selected from each cluster according to their score. The sentences selected are subjected to hierarchical lexical chaining.The topics and subtopics are first ordered according to the sequence in which they occur, to maintain coherency.

3.1.3 Decomposition Description

This section is used to show the decomposition of sub-problems described above. The working of the sub modules are explained here by using UML dataflow diagrams.
Pre-processing: The documents to be summarized are presented to the POS Tagger.
Automatic Text Summarizer: The following steps are involved in this module:

1. The tagged documents are input to the Automatic Text Summarizer
2. Hierarchical Clustering is applied at single document level, The Hierarchical Clustering for the multiple documents is carried out parallel and the output is a set of clusters from all the documents. Semantic Cosine Similarity is used for Hierarchical Clustering
3. These clusters are subjected to Topic Word and Subtopic word identification using Fuzzy C Means Clustering. A sentence is categorized into a particular topic/subtopic based on the presence of certain words. The sentences are re-clustered based on the topic/subtopic and hierarchical clustering index
4. The sentences in the newly formed clusters are ranked on 5 different dimensions and a rank is assigned for each sentence in the cluster

5. The sentences with a high rank are picked from each cluster according to the percentage of summarization specified by the user

6. The sentences which have been picked are lexically chained according to the order of topic words to which they have been categorized and according to their line number in the original document.

The overview of the system is given by the following dataflow diagram:

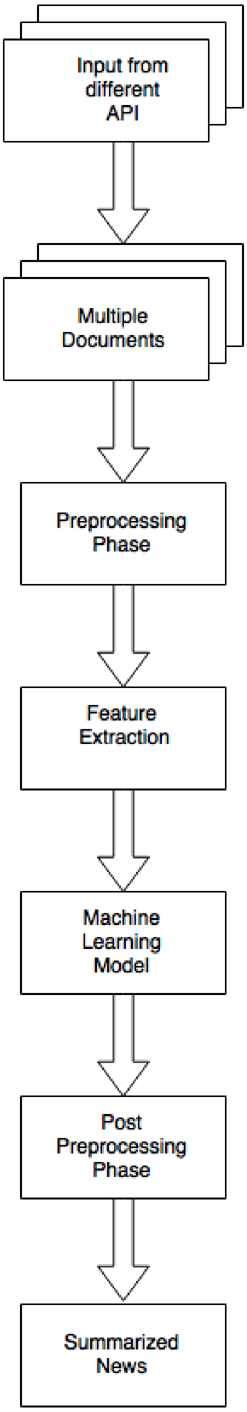

Figure 3.1: Basic Data Flow Diagram

3.1.4 Flow Diagram

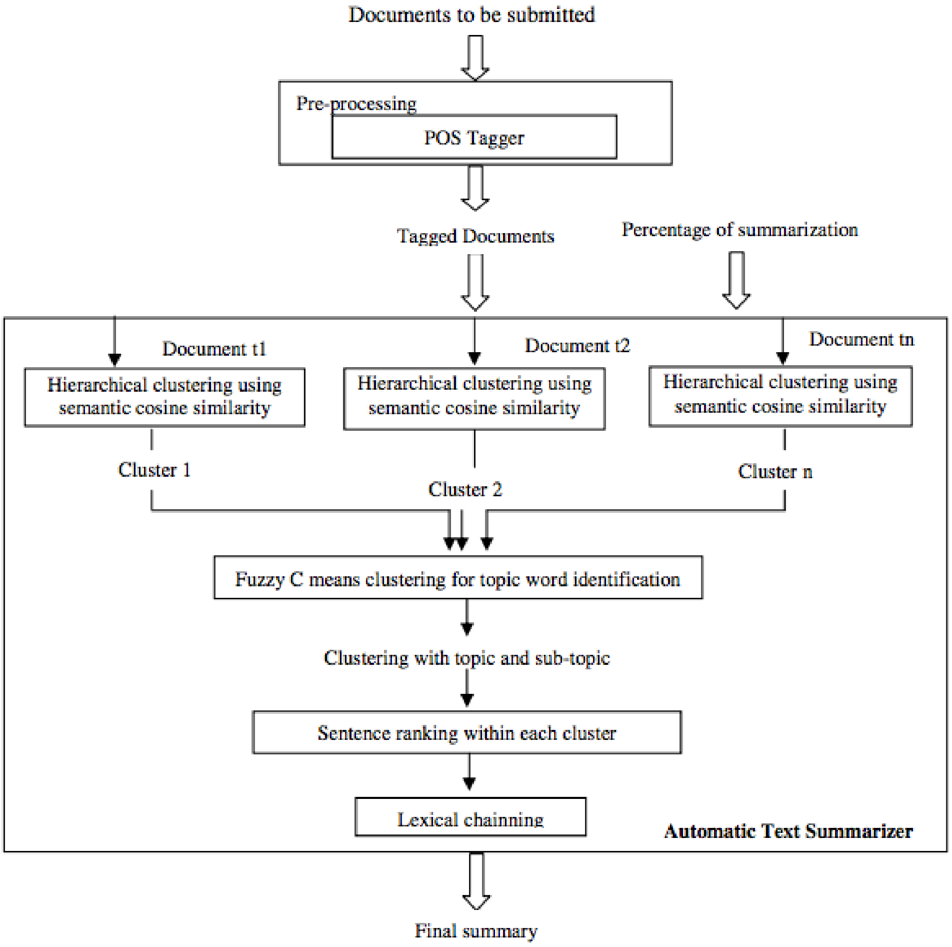
Documents to be submitted
Pre-processing
POS Tagger
Tagged Documents
Percentage of summarization
Document t1
Document t2
Document tn
Hierarchical clustering using semantic cosine similarity
Hierarchical clustering using semantic cosine similarity
Hierarchical clustering using semantic cosine similarity
Cluster 1
Cluster 2
Cluster n
Fuzzy C means clustering for topic word identification
Clustering with topic and sub-topic
Sentence ranking within each cluster
Lexical chainning
Automatic Text Summarizer
Final summary

Figure 3.2: Summary Generation

3.1.5 Design Rationale

The architecture explained in the above sections provides a detailed describes the overall working of the section. The main problem can be effectively subdivided as given in the architectural design. There were many decisions taken during the design phase. The summarization was decided to be done using Machine learning using clustering and lexical chaining. This was to enhance the accuracy and coherence of the summary. Moreover it was decided to use sentence ranking and cosine similarity measures for sentence selection.The sub-modules can be coded and tested independently. But the working of each of them are correlated to each other by the sequence of execution of the modules. This method is cost effective and also the most minimal solution that can be adopted for this project.

3.2 Design Validation

Design verification is used to ensure that the product as designed is the same as the product intended. In order to meet the customer expectations and avoid costly design modifications, appropriate selection of design verification method is essential. We have analysed the following project activities for accomplishing this goal:

- Concept through detailed design
- Specification development
- Detailed design through to pre-production

3.3 Case Tools

Computer Aided Software Engineering (CASE) is the application of a set of tools and methods to a software system with the desired end result of high-quality, defect free, and maintainable software products. In this section, we describe two tools that have been used so far, i.e. upto the design phase - Draw.io and SharelaTeX.

3.3.1 ShareLaTeX

ShareLaTeX is an online LaTeX editor which allows for real time collaboration and online compiling of projects to PDF format. Sharelatex was used to prepare all the documents related to the project. Documents were prepared using previously designed templates for articles and reports.

3.3.2 Draw.io

Draw.io is a free online diagram software for making flow charts, process diagrams, org charts, UML, ER and network diagrams. This website was used for drawing all the diagrams related to the project. The diagrams were designed according to the project requirements.