SENTENCE EXTRACTION BASED AUTOMATIC TEXT SUMMARIZATION USING AN OPTIMIZED FUZZY MODEL

by

Sundus Ayyaz

(2010-NUST-MS PhD-CSE(E)-25)

MS-10 (SE)

Submitted to the Department of Computer Engineering in fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

SOFTWARE ENGINEERING

Thesis Supervisor

Dr Muhammad Younus Javed

College of **E**lectrical & **M**echanical **E**ngineering

National **U**niversity of **S**ciences & **T**echnology

November 2012

بسم الله الرحمن الرحيم

DECLARATION

I hereby declare that I have implemented this thesis completely on the basis of my personal efforts under the guidance of Dr Aasia Khanum and in the supervision of my supervisor Prof Dr Muhammad Younus Javed. All the sources used in this thesis have been cited and the contents of this thesis are not plagiarized. No portion of the work presented in this thesis has been submitted in support of any application for any other degree of qualification to this or any other university or institute of learning.

SUNDUS AYYAZ

THE COMMITTEE

FEATURE BASED AUTOMATIC TEXT SUMMARIZATION USING AN OPTIMIZED FUZZY MODEL

by

Sundus Ayyaz

Approved and Accepted by:

Dr. Muhammad Younus Javed	**(**Supervisor)
Dr. Aasia Khanum	**(**Member)
Dr. Farooq-e-Azam	**(**Member)
Dr. Saad Rahman	**(**Member)

Dr. Muhammad Younus Javed

Dean Faculty of Engineering

ACKNOWLEDGEMENT

First of all I would like to thank ALMIGHTY ALLAH who has blessed me with so much for which I consider myself undeserving.

I am really very grateful to my supervisor **Dr Muhammad Younus Javed** for his supervision, support, guidance and most important his precious time. His encouragement has always been invaluable through out my thesis. I specially like to thank **Dr Aasia Khanum** for her continuous technical and intellectual support. Her precious advices have been of greatest help at all times. I would like to appreciate **Dr Asia Khanam**, **Dr Farooq e Azam and Dr Saad Rahman** for serving on my committee.

I would especially acknowledge my parents who are always been there for my moral and spiritual support and for standing by my side in the difficult times.

To my Loving Parents and Family…

ABSTRACT

The rapid growth of digital data on web has created the problem of information excess. Many users face difficulty to get the required relevant information within time from huge online repository.

Automatic text summarization is used to solve this problem by compressing the text into shorter form containing only the meaningful information so that it is not obligatory for user to go through each and every line in document for understanding the core concept behind it.

This thesis focuses on the design, implementation and analysis of an optimized fuzzy model by using a feature term based automatic text summarization method based on sentence extraction to generate meaningful summary of scientific documents.

Initially, the text document to be summarized is given to the system and the Preprocessing stage removes noise from the input document and produces a clean document. The proposed Model consists of three methods. First is the General Statistical Method (GSM), where feature terms are extracted by paragraph and sentence segmentation which includes further steps of tokenization, stop word removal, case folding and removal of non-essential sentences from document. Based on these identified feature terms; cue words, frequent words and sentence position, weights are assigned and each sentence score is calculated and the high score sentences are extracted. In second method, the Fuzzy Logic Model (FL), the output result from GSM and the identified features are used as an input to Fuzzy inference system (FIS). The FIS, on the basis of fuzzy rule set extracts the most important sentences out of the selected ones to be included in summary. In third method which is the Optimized Fuzzy Model (OFM) the input and output fuzzy parameters as well as the fuzzy rule weights are optimized to get the optimized weight of each feature. Now

each sentence score is calculated based on these weights and the highly scored sentences are selected to be included in final optimized summary document.

The proposed technique is implemented in java using NetBeans IDE 6.9.1 and Jfuzzylogic 2.1a package. In order to evaluate the system, the summaries generated using each of the three methods are tested with the golden standard summary (human-generated summary) and compared with each other as well as with other summarizers such as MS-Word 2007 summarizer and Essential summarizer for the purpose of comprehensive efficiency analysis. The evaluation measurements such as Precision, Recall and F-measure are calculated for each summary generated.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

List of Abbreviations

CoG	Centre of Gravity
DAG	Directed Acyclic Graph
FC	Full Coverage
FL	Fuzzy Logic
FCL	Fuzzy Control Language
GA	Genetic Algorithm
GP	Genetic Programming
GSM	General Statistical Method
HMM	Hidden Markov Model
IDF	Inverse Document Frequency
IR	Information Retrieval
KSRS	K-mixture Semantic Relationship Significance
NLP	Natural Language Processing

OFM	Optimized Fuzzy Model
POS	Part of Speech
SRL	Semantic Role Labeling
SRS	Semantic Relationship Significance
TF	Term Frequency

Chapter 1

INTRODUCTION

Text Mining is a process of extracting important unknown information hidden in the huge volumes of natural language text. Text mining has the analogy with mining of important ores from huge mines [11]. There are many well known applications of text mining e.g. text classification and clustering, information extraction and retrieval, web search, opinion mining, summarization and topic detection [8].

The area of Text Mining presents a vast scope with the concept of text summarization which reduces the text document into its compressed form, called document summary containing only the key information content helpful for the user to understand its relevancy without going through the whole document, hence saving time and effort [2, 13, 16].

Text Summarization is classified into two types of summarization methods, extractive summary and abstractive summary. Extractive summary involves the extraction of key words, phrases or sentences from the given text document [12]. While in Abstractive summary, a meaningful short summary is generated containing the words not explicitly available in the text document, a linguistic method is used to produce such summary. Till now only the extractive summarization method is used to produce document summary as abstractive methods are still fragile [2, 11].

1.1 PROBLEM OVERVEIW

Text Summarization has gained much importance with time as the data size on World Wide Web has increased resulting in an information excess. Most of the users face difficulty in acquiring the relevant information on web within time [16]. Hence Text Summarization helps in reducing the time as well as making the search more compact and producing a meaningful content. [2, 12]. For performing text Summarization on the corpus of scientific articles, it requires pre-processing on each text document. To generate an automatic informative summary of each document, an implementation of GSM and FL methods is done along with the novel approach for sentence extraction called OFM. The blend of these three methods shows good results with time and resource efficiency.

1.2 PROJECT OBJECTIVES

The main idea of this project is to design an extractive text summarization model based on three methods, used to extract important sentences from the text document. Using the general statistical method (GSM), the key features are identified. These features provides a way to assign the weight to each sentence, calculate their total score and finally select the top ranked sentences to produce an initial summary document. Using Fuzzy Model, the initial summary document along with the three features is fed to the fuzzy inference system as crisp input. The output then determines the corresponding importance of each sentence to be included in fuzzy level summary document. One of the main contributions of this project is the design of an optimized fuzzy model. The input and output features are optimized by removing errors and assigning optimized values and weights to features and rule sets to produce a third level summary; an optimized fuzzy model summary document which shows a considerable improvement in information retrieval measurements as compared to the previous two summary documents.

To completely study the effectiveness of the optimized fuzzy model, they had to be implemented so their results can be analyzed. They have been implemented in java using NetBeans IDE 6.9.1 and Jfuzzylogic 2.1a package.

The summaries generated by our approach are also compared with other summarizers, the MS-Word summarizer and the Essential summarizer to show that it gives the most precise results.

1.3 THESIS STATEMENT

The motivation for using a hybrid approach with GSM, FL and OFM is to select the most important and informative text from the given text document and condensing it at the same time with only relevant information.

The thesis is divided into 4 other chapters. Chapter 2 presents a literature review of text summarization and the previous related work describing the approaches used by other researchers for the extractive text summarization of the corpus of different articles. Chapter 3 represents the proposed architecture containing the three methods (GSM, FL and OFM) and their components used to extract the important sentences and generate summary for the scientific articles. Chapter 4 shows the results evaluation. This chapter depicts the precision, recall and f-measure of the summaries produced by the three methods and compares their results with other summarizers. Finally chapter 5 concludes the thesis and presents future directions for further enhancements and research.

1.4 SUMMARY OF SEARCHING ACTIVITIES

This section shows which electronic databases and search engines are used to get the relevant papers on extractive text summarization as well as it shows the key words that are given as input in search engine to get the required papers. An Electronic Database IEEE Explore is searched using the keyword “Text Summarization in Text Mining” through which 257 results are generated and I have analyzed the top 50 focused on articles specifically related to text summarization of only scientific corpus through sentence and keyword extraction rather than generally related to text mining concerned towards multilingual spoken language, video summarization or multi document summarization and text summarization using graph algorithms as our focus is totally on feature terms for including the sentence in a summary. Only 2 papers met the required criteria [1] and [2]. Using terms “Automatic Text Summarization using Sentence Extraction” all of 19 records in IEEE Explore are found by searching From this list 4 papers were selected [3], [4], [5] and [6] based on the criteria defined above.

Google is searched with the keyword ‘Automatic Text Summarization.’ From the generated list of articles SCHOLARLY ARTICLES FOR AUTOMATIC TEXT SUMMARIZATION is clicked, the first heading from the list which gave a broader search finding 1000 records where relevant ones are selected from the first 100 in the ranking which are [7, 8, 9, 10] based on the good impact factor of journals and conference papers and importance of each paper with our area. We have not included any video segmentation and visual summarization corpus, Chinese, Persian and other linguistic and multi-document summarization, also rejected the network or graph based text summarization and summarization in other fields such as biomedical or news. We are more concerned on getting articles towards sentence extraction rather than keyword or paragraph extraction, single document summarization in a domain specific context. With keyword ‘Text Summarization Extractive Techniques’, top 50 results are analyzed. As we are focused towards extractive technique rather than abstractive, we found 7 records which contain the relevant document to study from good journals and conferences, from them we selected and analyzed the best 3, which are [11]. [12] and [13]. By using keywords “text summarization using sentence extraction” – first 70 search results are analyzed omitting the articles with graph-based and learning approaches towards text summarization. Bengali and Chinese Text summarization and other languages are also excluded from our study phase. While focusing on our scientific domain we have rejected the summarization related to speech, multi-document and learning algorithms from our search criteria. This includes [14], [15], [16], [17], [18], [19], [20] and [21].

Chapter 2

LITERATURE REVIEW

2.1 INTRODUCTION

This chapter provides an insight into the literature review of the previous and recent related work on Text Summarization of text documents. It describes the methodologies, techniques and algorithms used by other researchers to carry out their research work on extractive text summarization. Section 2.2 shows the critical analysis of the approaches used in the reference papers and their pros and cons.

2.2 CRITICAL ANALYSIS

Text Summarization is an active research area of Text Mining. More emphasis has been given on extractive summarization; here we will also analyze various research articles written from the viewpoint of extractive summarization. The first tentative research on automatic text summarization embarked in late fifties by H.P. Luhn. Luhn extracted the important sentences from text based on sentence scoring by weighting the word frequency in each sentence [19]. In 1969, [20] Edmunson proposed a novel feature of calculating the sentence weights and extracting important sentences by using Cue words. These Cue words are identified in the given text and are compared with Cue Dictionary corpus to calculate their cue weights but this technique has a limitation of being complex and inefficient. As the years passed, different approaches and techniques were invented for summarization purposes due to an increase in digital text on web which is continuously growing with time and the need to get only desired information from huge repository efficiently and effectively. A few of these approaches which are studied from numerous literatures along with their pros and cons are given below:

2.2.1 Fuzzy logic based approach

Using this approach Ladda Suanmali [1] presented that some sentences are selected based on their features such as title, sentence length, term weight, sentence position, sentence to sentence similarity, proper noun, thematic word, and numerical data and used as an input to the fuzzy inference system for text summarization based on fuzzy logic. A triangular membership function is used for each feature and gives the value between 0 and 1. The input membership function is divided into three membership functions such as for each feature low (L) and very low (VL), Median (M) high (H) and very high (VH). Similarly the output membership function is divided into three membership functions (Unimportant, Average and Important). The important sentences are extracted using if-then rules based on our features criteria. This Fuzzy summarizer when compared with MS word 2007 summarizer and baseline summarizer gives better result in precision and recall but the proposed approach should be extended and combined with other learning methods to give much better results in terms of summary quality [1].

In [3] Kiani and Akbarzadeh proposed a *Hybrid GA and GP* technique for text summarization using a combination of Genetic Algorithm (GA) and Genetic Programming (GP) to optimize membership functions and rules set of a Fuzzy System. The sentence features for an input to fuzzy inference system are title words, sentence position, sentence length, number of thematic and emphasize words. The proposed method is compared with other summarizers such as MS word 2000 summarizer and Copernic with precision and recall. Results show an improvement but require large training data to learn accurately.

Ladda Suanmali in A Fuzzy Genetic semantic based text summarization [6] extended the work in [1] which gives a significant improvement in the quality of text summarization. In the proposed approach a feature fusion technique is applied to find out which features out of available ones are most useful. The approach is a combination of GSM, fuzzy logic, GA and SRL to generate high quality summaries. When compared with other method and benchmark summarizers, this approach does better than other text summarization approaches. Figure 2.1 shows the architecture of the proposed Fuzzy Genetic Semantic Method.

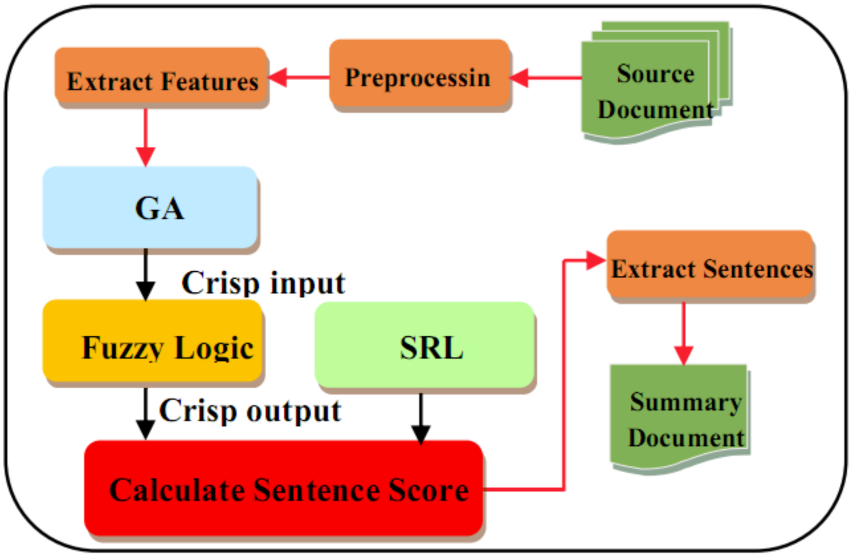


Figure 2.1: Fuzzy Genetic Semantic Method based Text Summarization Architecture [6].

2.2.2 Evolutionary/Learning Approach:

a. **Genetic Algorithm:** Ladda and Naomi in [5] presented a GSM technique in combination with GA for sentence extraction to get an informative summary of the text document. GA is used as an improvement algorithm to optimize the feature weights. The average feature weights acquired by GA cannot assure that the feature weights are best for the test corpus.

In [7], Maher and Abdelmajid proposed a two step extraction, Generation of population of extracts and Classification of these extracts to select the best one from a corpus of 121 NLP articles. An ExtraGen (Extraction using GA) system is designed for the automatic summarization which is compared with Microsoft Auto-summarizer and Copernic summarizer using variable extract's length which shows the best value of the recall and precision but it only works best with specific articles and with single document summarization.

b. **Harmony Search Algorithm:** Ehsan and Leila in [17] proposed a harmony search algorithm for sentence extraction based on three factors; readability, cohesion and topic relation factor. Document to be summarized is represented as a DAG with nodes showing the appearance of sentences. The harmony search algorithm is developed using the three factors and evaluated by applying on the corpus of DUC2002, its precision and recall values are calculated by comparing with previous research on GA and the comparison showed that the harmony search text summarizer gives much better performance. The summarizer results can be improved by addition of more factors or by improving the already implemented factors to get good summarization results.

2.2.3 Linguistic Approach:

A K-mixture probabilistic model is implemented for statistical automatic text summarization [2] to create term weights and Linguistic semantic relationship significance (SRS) of nouns is used to extract meaningful sentences. The summary using this approach is generated starting from preprocessing than determining the term weights and term relationships. For generating a qualitative summary KSRS (K-mixture semantic relationship significance) a combination of statistical and linguistic is used. Two experiments are conducted in [2] to validate the proposed approach. The results showed that using the statistical approach of assigning term weights perform better than TF-IDF approach, while using linguistic approach the results are improved and by combining both the approaches (statistical and linguistic), we get the best results but KSRS only performs best when the summary proportion is less than 40 percent.

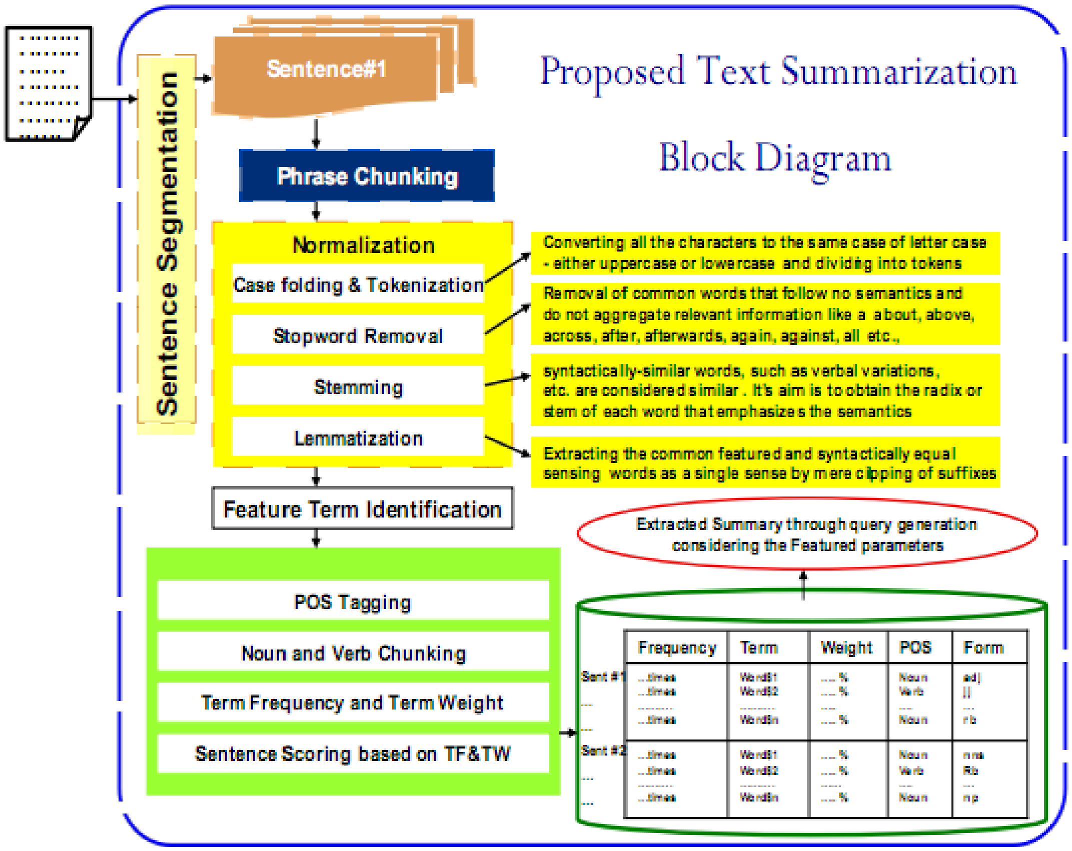
Flores and Chalender in syntactico- semantic analysis for sentence extraction [4] use sentence extraction method based on semantic analysis and a combination of semantic and syntactic analysis on the corpus of scientific articles and newspaper articles. The evaluation of first strategy illustrates that it shows good performance with general language documents such as news articles and enterprise reports. While the second strategy proves that following a hybrid approach always gives a better result but for the improvement in the linguistic quality of summary, other linguistically motivated methods are required other then simple extraction. The evaluation results have clearly presented that the extractive summarization approach are not adequate for long documents so research on abstraction and generalization holds more importance.

2.2.4 Topical Structure approach

Zhan and Loh, uses automatic text summarization on customer reviews obtained from online blogs [8]. The proposed approach analyzes the topical structure of the customer’s online reviews and further identifies, extract and rank the topic for final summary generation. This approach is found to perform better when evaluated with approaches of opinion mining and clustering summarization. Therefore summarization of the customer's reviews gives us a filtered output. But the reviews that are written in different styles and spread across different sources such as Amzon.com and Epinions.com, the integration of these reviews from distributed sources are still a limitation in this field which should be focused.

2.2.5 Stochastic Tagging technique

In Corpus based Automatic Text Summarization [9], Suneetha and Sameen proposed a POS tagging approach with HMM tagger which is evaluated with a brown corpus to extract important sentences as summary.
The proposed method first decomposes the text into sentences and assigns POS to each word in a sentence and stores the result in a table. After this, stop word removal, lemmatization and stemming are performed on the tokens; the words which are left are considered keywords. Next, the feature terms are spotted and the summary is generated based on the feature terms. This technique is very helpful in generating summaries for single document using supervised POS tagging. The Figure 2.2 illustrates its proposed architecture. The model consists of the following stages.


Proposed Text Summarization
Block Diagram
Sentence Segmentation
Sentence#1
Phrase Chunking
Normalization
Case folding & Tokenization
Stopword Removal
Stemming
Lemmatization
Converting all the characters to the same case of letter case - either uppercase or lowercase and dividing into tokens
Removal of common words that follow no semantics and do not aggregate relevant information like a about, above, across, after, afterwards, again, against, all etc.,
syntactically-similar words, such as verbal variations, etc. are considered similar . It's aim is to obtain the radix or stem of each word that emphasizes the semantics
Extracting the common featured and syntactically equal sensing words as a single sense by mere clipping of suffixes
Feature Term Identification
POS Tagging
Noun and Verb Chunking
Term Frequency and Term Weight
Sentence Scoring based on TF&TW
Extracted Summary through query generation considering the Featured parameters

	Frequency	Term	Weight	POS	Form
Sent #1	...times ...times ....times	Word#1 Word#2 .... Word#n	.... % .... % .... %	Noun Verb ... Noun	adj jj ... rb
Sent #2	...times ...times ....times	Word#1 Word#2 .... Word#n	.... % .... % .... %	Noun Verb ... Noun	nns Rb ... np

Figure 2.2: Corpus based Automatic Text Summarization System with HMM Tagger[9].

An automatic text summarization using POS tagging is also proposed by Hashemi [11]. The method involves four stages; the first stage is preprocessing which removes stop words, performs stemming, assign POS tagging and stores the result in table. In second stage, important keywords are extracted from preprocessed text to select the important sentences based on these keywords and many other features. The third stage is to extract the sentence with highest rank and in forth stage the important sentences are filtered to generate the final summary. The following diagram represents the proposed system.

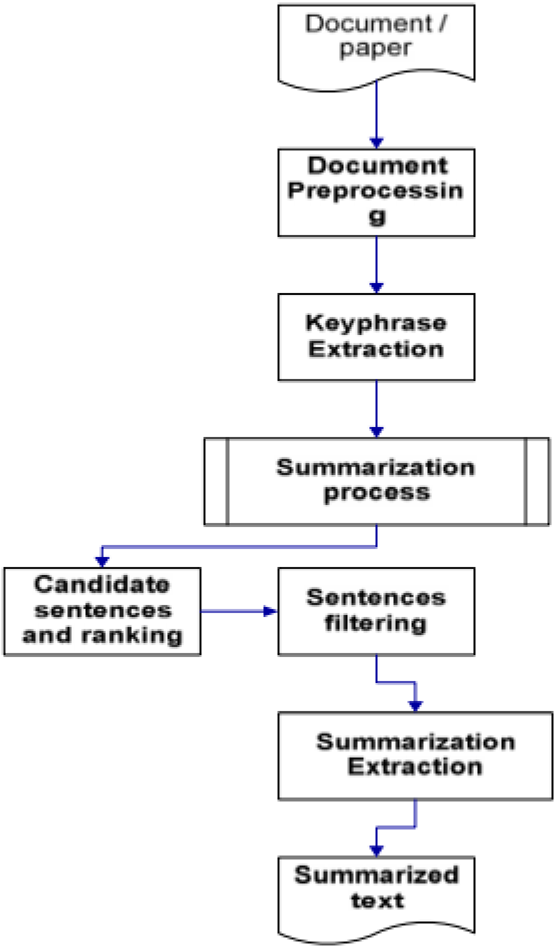


Figure 2.3: Text Summarization Extraction System using Extracted Keywords [11].

A corpus of Computer Science is used with training set size of 90 documents which is tested by 20 documents. Evaluation is done using recall and precision measurements; it shows that the system gives good results with high-quality summary when compared with manual extraction system.

2.2.6 Semantic Approach

A frequent term based text summarization algorithm is proposed by Naresh [12], using this method the summary is generated by selecting the frequent terms in document and their corresponding semantically similar words and storing them together in table, the important sentences are extracted based on these terms. The algorithm is tested on corpus of 183 documents from Computation and Language collection. The system is evaluated using compression and retention ratios. The results shows a better understanding of concept through summary but only the occurrence of frequent term alone cannot give us the best results as we can obtain from manual summarization. The proposed methodology for the given technique is represented as follows:

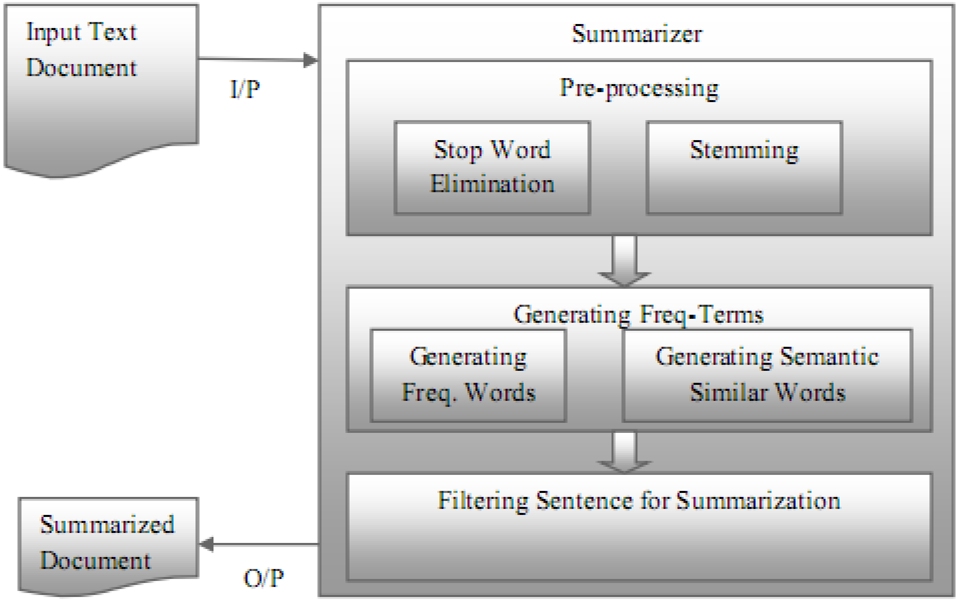


Figure 2.4: Overall methodology of Frequent-Terms and Semantic Similarity Based Summarization [12].

2.2.7 Statistical Approach

Hiroshi and Rihua proposed a method of automatic text summarization based on word importance measures [14]. Using this technique, local and global weights are assigned to nouns in text and the importance of sentences is measured according to the scores of these nouns, high scored sentences are selected for summary but final decision is made after performing coherency test. The proposed method gives satisfied results but not better than other systems when compared to them. The system only performs well for certain applications.

In [16], a statistical text summarization approach for sentence extraction based on sentence scoring is used with the method of assigning weights to every word in a sentence and adding a boost factor if the word appears in font like bold, italic, underlined or any combination of them, in this way the value of word is increased hence increasing the importance of a sentence. The algorithm is tested on 10 documents and evaluated with MS-Word summarizer showing a higher accuracy rate as compared to word summarizer because of the addition of boost factor. The generated summary not always gives a perfect meaning because of the problem of incoherency while extracting sentences.

2.2.8 Information Retrieval Oriented Approach

Daniel and James in [15] used IR approach towards automatic text summarization by proposing a Full-Coverage summarizer for extracting non-redundant sentences. The algorithm works in three steps; after parsing the document into sentences, stop words are removed and word stemming is performed. In second step a subset of sentences are selected that conveys the whole concept of the document. In third step, the final summary is generated based on the FC ranked sentences. For evaluating the FC algorithm a Time magazine collection from SMART and TREC collection from DUC2002 is used. The results show that it is possible to achieve a small loss in precision while using an undersized text of 3-5 percent. More importance is given in examining effective generative algorithms for extracting the sentences that can present the information at a higher level of abstraction.

2.3 SUMMARY

This chapter presented the previous and recent work related with automatic text summarization. The presented concept shows the reason for this thesis work. The different methods and algorithms used by different researchers to implement the automatic text summarization system are discussed here and are critically analyzed to explore the strengths and weaknesses of different techniques.

Chapter 3

PROPOSED METHODOLOGY

3.1 INTRODUCTION

To propose a novel method for automatic text summarization, a comprehensive analysis of recent and previous work of researchers is done and various problems and issues encountered by them are also explored such as:

a) The incoherency in the generated summary while extracting sentences does not give a meaningful summary.

b) The efficiency of the system in terms of execution time and resources.

c) Extraction summarization approach is not always adequate for large documents.

d) The number of feature terms used to calculate the score of each sentence.

e) Only one or two feature terms for sentence score calculation have fewer chances to provide a quality summary of the input text document.

f) Following a hybrid approach for extraction of important sentences always give a better result as compared to an individual technique.

g) Some approaches only perform well when the required summary proportion is less than 40 percent.

To address these issues, the conventional methods and algorithms have to be reorganized to produce a quality summary.

This chapter presents the architecture design and implementation details of the proposed General Statistical Method (GSM), the Fuzzy Logic Method (FL) and the new Optimized Fuzzy Model (OFM) for automatic text summarization. This hybrid approach of three methods helps to produce three different summary documents by selecting important sentences from document using each method and producing the high precision, recall and f-measure summary document using the OFM.

In this chapter, Section 3.2 illustrates the proposed system architecture of our automatic text summarization system along with the details of its different methods and their implementation details. Section 3.3 shows the flow chart of the whole system and section 3.4 represents the summary of the chapter.

3.2 SYSTEM ARCHITECTURE

The proposed architecture of the system is shown in Figure. 3.1. The main objective of the summarization system is to automatically generate summary of documents with high precision, recall and f-measure results. The proposed system consists of three main methods to produce summary; the General Statistical Method, the Fuzzy Logic Method and the Optimized Fuzzy Model. Each method contains different components for converting the text document into its compressed version. The overall system detail is given as follows.

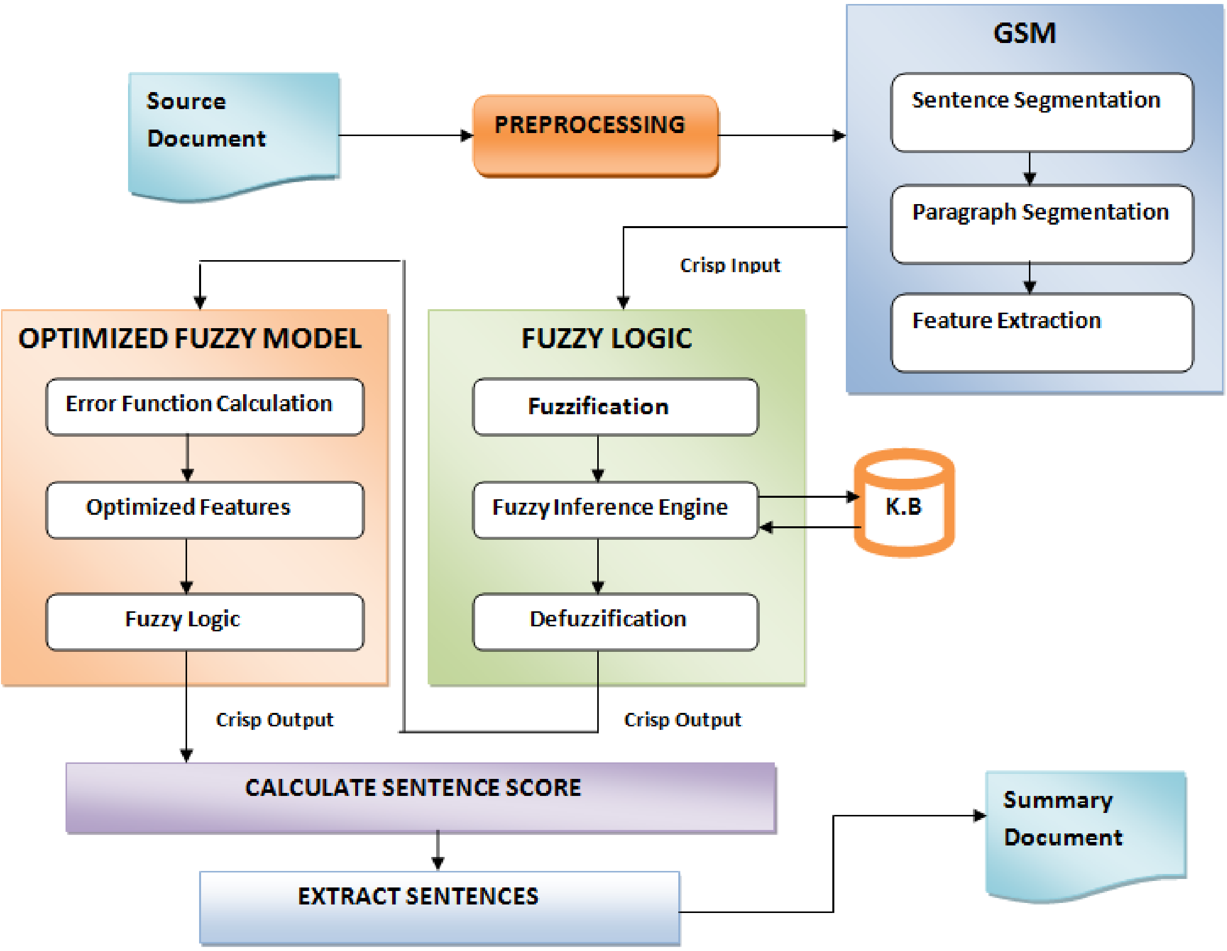


Figure 3.1: Proposed Text Summarization System

3.2.1 Pre-processing

Pre-processing on every input text document is required as the raw articles also contain the noisy text that shouldn't be included in the summary as they are the cause of ambiguity. In the pre-processing step, the text is loaded into the system and a clean document is produced by removing noise from the document. Removing noise includes removal of author names, journal names, page numbers, headers, footers, end-notes, references from the text document to make it ready for summarization process. The pre-processing step can either be performed manually or programmatically. The references part from the document is removed programmatically while other noise text is removed manually by the user.

3.2.2 General Statistical Method (GSM)

The first method used to produce an initial summary is the General Statistical Method. The GSM is used for extracting features and assigning weights to each sentence and than calculating total score of each sentence for extracting the Top Scored sentences to be included in summary. The GSM produces an initial summary document I. The GSM architecture given in Figure 3.2 shows the steps involved.

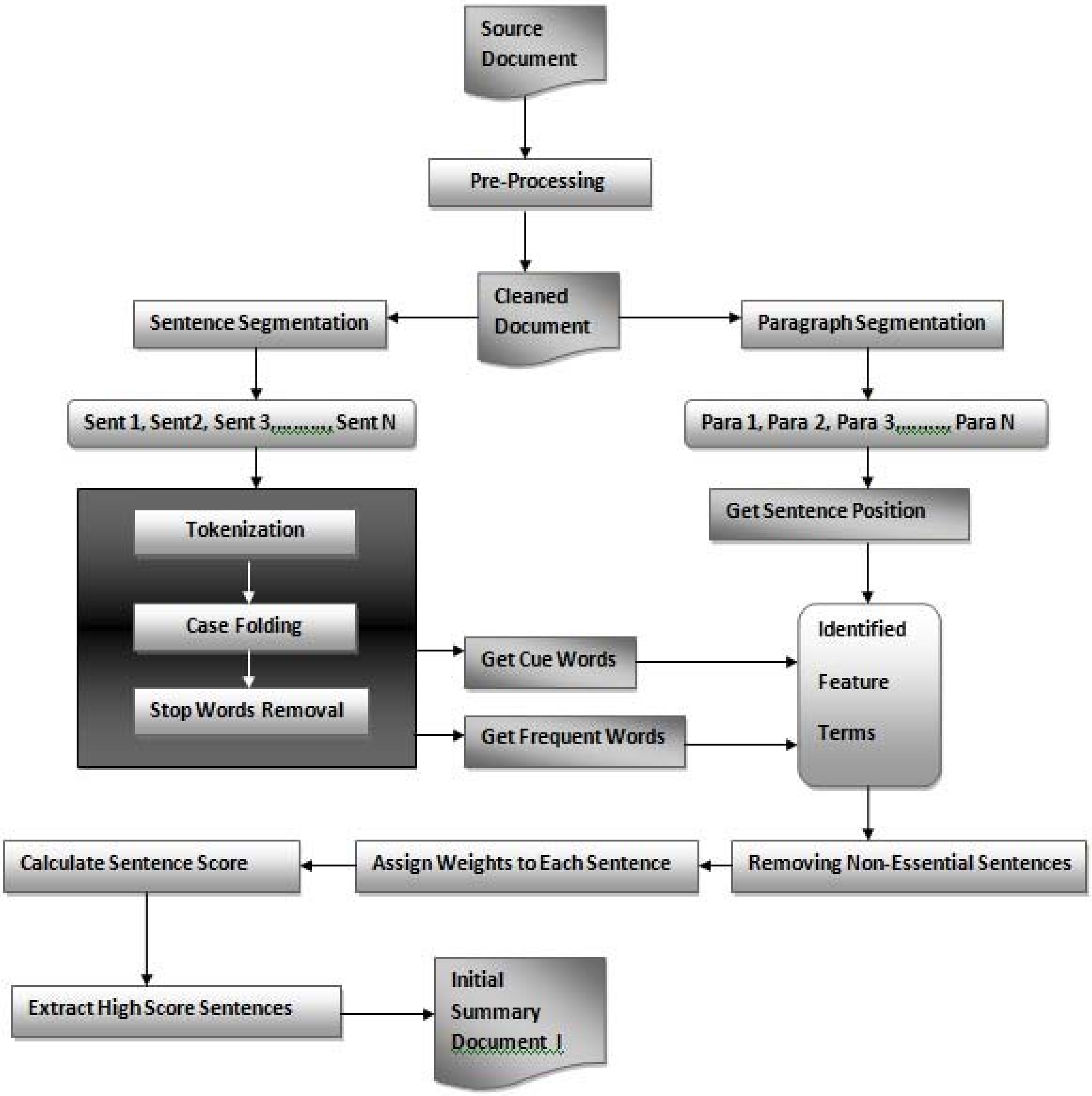


Figure 3.2: Architecture of General Statistical Method for Summary Generation

Paragraph Segmentation

The text document to be summarized is split up into its constituent paragraphs to get the position of each sentence in the paragraph and assign them weights according to their position in paragraph. The sentence position is one of the feature on the basis of which the weight is assigned to sentence. The first sentence in paragraph has more importance then the second so it is assigned greater weight than second, similarly the second sentence carries more importance than third and so on. Therefore, paragraph segmentation helps to get one of the features. The data structure used to store paragraphs is an array list.

Sentence Segmentation

The document text is divided into sentences for the extraction of important features on the basis of which the weights are assigned to each sentence. After Sentence Segmentation, the sentences are divided into its corresponding tokens for identifying the two important features, the cue words and the frequent words. These features then assist us to select the important sentences. For the identification of important sentences, the following steps are performed.

i. Tokenization

Tokenization of sentences is important as they help to extract the two important features, the cue words and the frequent words from the text. In this step, each sentence is split up into its tokens using StringTokenizer class in Java. The array list of String is used to store the tokens of each sentence.

ii. Stop Words Removal

After tokenization, the less significant words from each sentence are removed by loading a stop words list which is stored in a text file using HashSet, the data structure of Java from the package java.util and filtering the remaining ones which are the candidate words. An example of few of the stop words include, ‘a’, ‘an’, ‘is’, ‘by’, ‘the’, ‘of’, ‘are’, ‘am’, ‘and’, ‘only’, ‘just’, ‘there’, ‘your’ etc. The text is now clean and contains only candidate words.

iii. Case Folding

After performing the stop words removal process, all the remaining words called the candidate words are converted to lower case to avoid the replication of words in different cases, this helps to develop the accuracy of the system as well as to get the most frequently occurring words in the document.

iv. Removing Non-Essential Sentences

The document size is further reduced by removing the unnecessary sentences from the text by checking each sentence if it contains or start with certain words that makes them insignificant to be included in summary document. i.e. if the sentence starts with “for example” or if the sentence includes only keywords from document or if it describes different sections of document and discusses about given figures than we consider such sentences not to be included for summary. Hence, condensing the text with valuable information. The words of each sentence are contained in an array list of java which is than checked with the non-essential words and in this way the sentence that start with or contains certain words makes them unimportant and are removed from the document.

3.2.2.3 Extracting Sentence Features

The summarization system uses the following features to extract important sentences from document. We focus on three features for each sentence.

- Cue words
- Frequent words
- Sentence Position

i. Cue Words

Cue words in the sentences are the phrases that give a sign about that particular sentence to be important. From the point of scientific article summarization, some of the cue words are ‘finally’, ‘therefore’, ‘presents’, ‘article’, ‘approach’ etc. if the sentence contain such words then it is assigned a particular weight depending upon the number of cue words it contains. For example, consider the sentence “The purpose of this paper is to explain set of web services approach is…” would be assigned a score of 2 with respect to the cue words included as it contains two cue words, ‘purpose’ and ‘approach’. In this way the weights are assigned and scores are calculated.

ii. Frequent words

The number of occurrences of each word in the text is calculated and the most frequently occurring words (with the specified threshold) from the text are stored in treeMap (data structure in java) with its number of occurrences (integer) from java.util package. Each sentence in the text is checked with this treeMap list and the sentence containing the words which are also in the most frequent words list is assigned a particular weight depending upon the number of most frequent words it contains. Therefore, for calculation of the top N frequent words in the text the data structures, a treeMap and the Collection class is used.

iii. Sentence Position

Sentence position in a paragraph determines how much the sentence is important. The position of the sentence in each paragraph is measured and a weight is assigned to it. We have considered maximum of weight 5. For example, the first sentence in a paragraph is always very important starting with weight of 5 then the second sentence with weight 4, third with weight 3 and zero weight for all the sentences after 5 sentences.
Taken from [1],

Score= weight 5 for 1st sentence, weight 4 for 2nd, weight 3 for 3rd, weight 2 for 4th, weight 1 for 5th and weight 0 for other sentences.

Sentence Score is calculated by adding the weights obtained by these three features and the high score sentences are selected for initial summary.

The results from the GSM system are then passed to the fuzzy system.

3.2.3 Fuzzy Logic Method (FL)

In order to get a more refined summarization result with more precision a fuzzy logic method is used. Text summarization system is further implemented by applying fuzzy logic. The features extracted in GSM are used as the crisp input, which are fuzzified to enter into fuzzy inference system. The FIS extracts the rule set from the knowledge base to get the output result. The output that is generated is defuzzified to get the crisp output for measuring the importance of each sentence eligible to be included in fuzzy summary document.

Figure 3.3 shows the FL architecture.

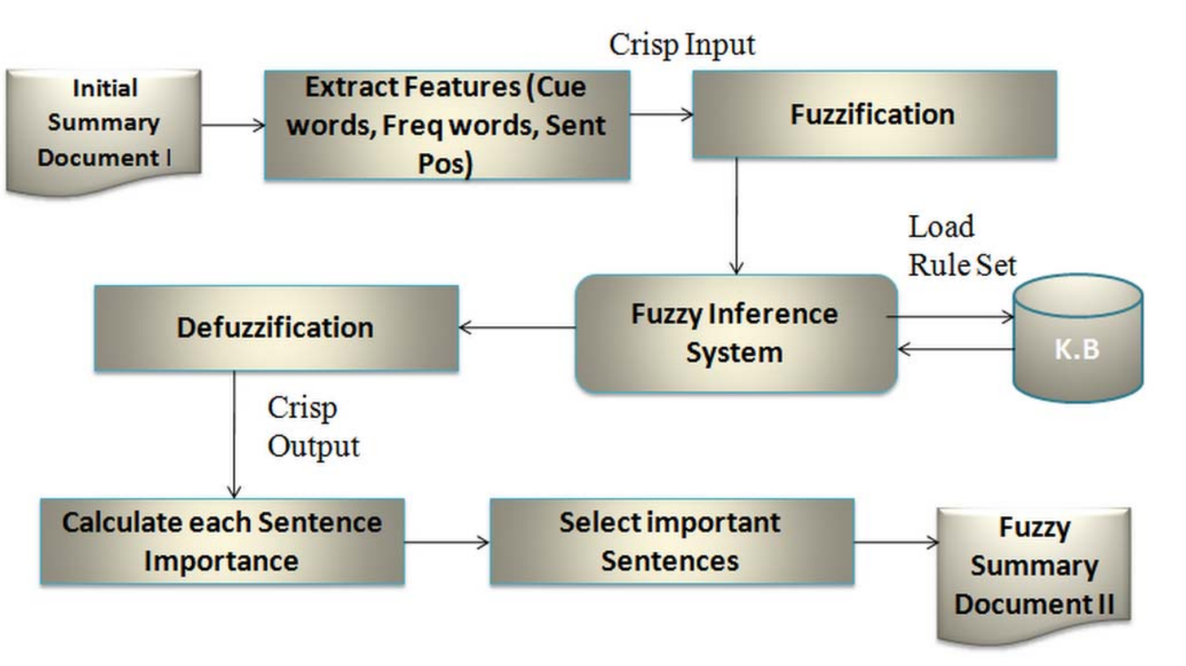

Crisp Input
Initial Summary Document I
Extract Features (Cue words, Freq words, Sent Pos)
Fuzzification
Load Rule Set
Defuzzification
Fuzzy Inference System
K.B
Crisp Output
Calculate each Sentence Importance
Select important Sentences
Fuzzy Summary Document II

Figure 3.3: The Fuzzy Logic Model Architecture for generating Text Document Summary

3.2.3.1 Implementation Details

The FL method is implemented in java by importing the package jfuzzylogic.jar. JFuzzyLogic is an open source java library. The library provides complete FIS and implements Fuzzy Control Language (FCL). The FIS can be implemented in several ways and in this project it is implemented in FCL. FCL is used to define the input variables, the output variables and the rule sets for output variables. In this project, there are three input variables e.g. Cue Words, Frequent

Words and Sentence Position and one output variable, Sentence Importance. The input and output variables are the first to be described in FCL. The output result is generated according to the rules defined. The fuzzy inference engine (fis) applies the fuzzy IF-THEN rules to the input features and gets the output feature sentence importance as either poor or good to be included in Fuzzy Summary. The Fuzzy Control Language (FCL) is used for defining the input and output features and the rule set.

Figure 3.4 shows the structure in which the variables are defined in FCL.

```
// Block definition (
FUNCTION_BLOCK features
// Define input variables
VAR_INPUT
   CueWords : REAL;  FreqWords : REAL; SentencePosition : REAL;
END_VAR
// Define output variable
VAR_OUTPUT
   SentenceImportance : REAL;
END_VAR
```

Figure 3.4: The structure of FCL file for defining the input and output variables.

Next in the FCL, the fuzzy terms for the input variables are defined. Each input feature (Cue words, Frequent words and Sentence Position) is divided into three membership functions which are {vLow, Low, High} and a range is assigned with each membership function. For example, Cue words can be vLow, Low and High depending upon the numerical range given. Figure 3.5 shows the way the variables are fuzzified.

The fuzzy rules are also defined in FCL in Figure 3.5 as follows;

```
// Fuzzify input variable 'CueWords'
FUZZIFY CueWords
    TERM vLow := (0, 1) (1, 0) ;   TERM Low := (0, 0) (1,1) (2,0);  TERM High := (1, 0) (2, 1) (3,1);
END_FUZZIFY
// Fuzzify input variable 'FreqWords'
FUZZIFY FreqWords
    TERM vLow := (0, 1) (2,1) (3,0); TERM Low := (1,0) (3,1) (4,0) ; TERM High := (2,0) (4,1) (10,1) ;
END_FUZZIFY
// Fuzzify input variable 'SentencePosition'
FUZZIFY SentencePosition
    TERM vLow := (0, 1) (1, 1) (2,0) ;  TERM Low := (1,0) (2,1) (3,1) (4,0); TERM High := (3,0) (4,1) (5,1);
END_FUZZIFY
```

Figure 3.5: The structure of FCL file for defining the fuzzy terms for the input variables.

Similarly, the output feature ‘Sentence Importance’ is divided into two membership functions {Poor, Good} with a defined range. In case of the output variable, the inverse is done that is the variable is defuzzified as presented in Figure 3.6.

```
// Defzzzify output variable 'SentenceImportance'
DEFUZZIFY SentenceImportance
    TERM POOR := (0,0) (6,1) (8,0);
    TERM GOOD := (6,0) (8,1) (15,1);
```

Figure 3.6: The Defuzzification of the output variable.

In our project, the method used for Defuzzification is the Centre of Gravity method (CoG). This method can be clearly understood by the concept of heaps of sand as for each fuzzy term, the position of this heap of sand on the output variable is defined and the height of the sand heaps verifies how much this term is true. The CoG is then calculated, to be used as output variable [21]. Figure 3.7 shows the FCL definition for the Defuzzification method.

```
// Use 'Center Of Gravity' defuzzification method

METHOD : COG;

// Default value is 0 (if no rule activates defuzzifier)

DEFAULT := 0;
```

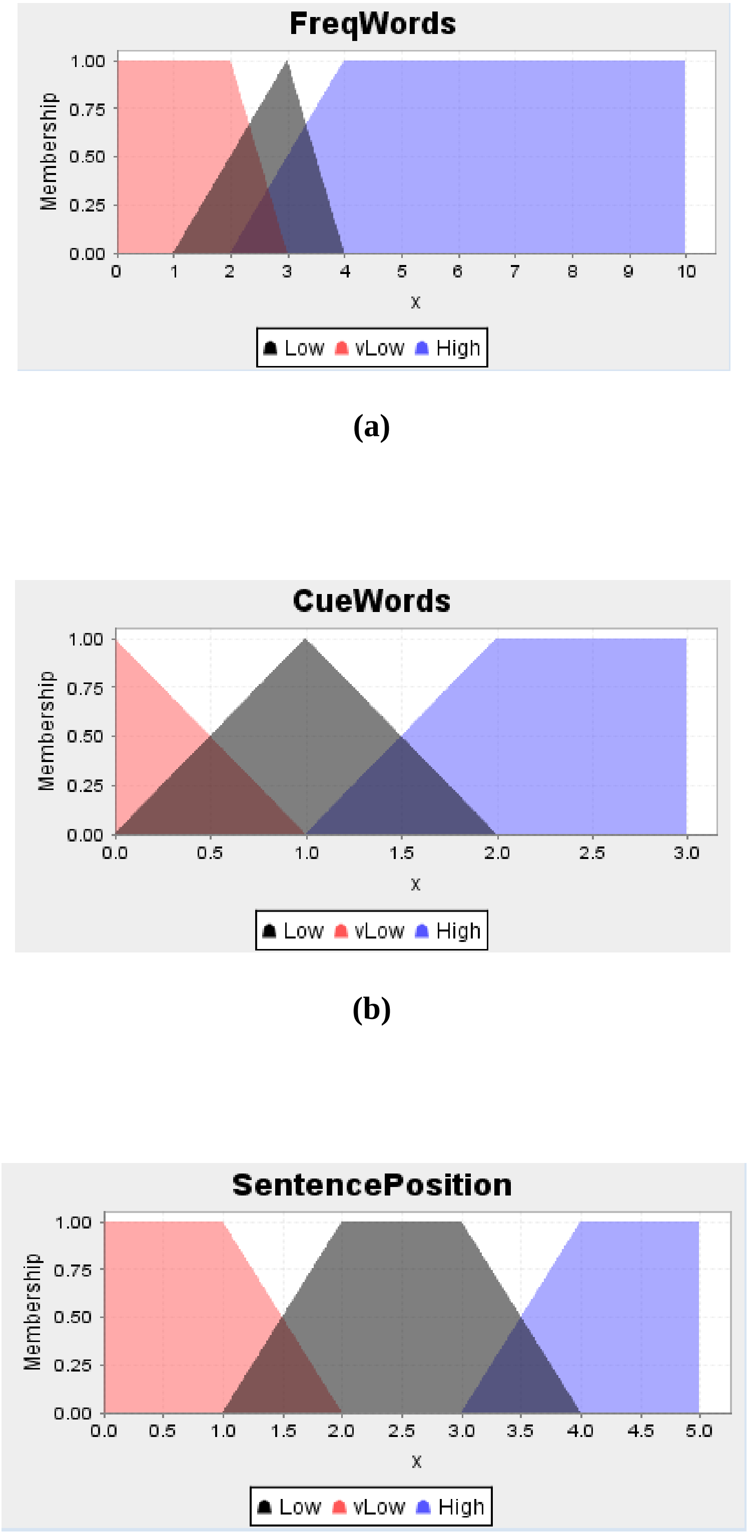
Figure 3.7: FCL definition for the Defuzzification method

After defining the variables and terms, next the Aggregation, Activation and Accumulation methods are defined. The Aggregation method defines how the condition is calculated as the condition is defined with AND or OR. Since fuzzy variables are used as operands, the boolean definitions of AND and OR are not sufficient. The definition for OR was chosen to be the minimum. This implies that AND is calculated with the maximum of the operands. The 'fuzzy rulesets' are described next which are the significant part of the system. In FCL, several rule blocks can be defined but in our project we have considered only one rule block. Each rule will draw a conclusion, based on a condition. This conclusion is a fuzzy variable. How much this conclusion variable is true due to the condition is determined with activation. Multiple rules may come to the same conclusion. For example, multiple rules may draw the conclusion that the result is vLow. Accumulation determines how these same conclusions are combined. The sum of the conclusions was chosen for accumulation. There are 30 rules which are defined in FCL with respect to the three features. The FCL structure for fuzzy rules set is shown in the Figure 3.8.

```
RULEBLOCK No1
   // Use 'min' for 'or' (also implicit use 'max'
   // for 'or' to fulfill DeMorgan's Law)
   OR : MIN;
   // Use 'min' activation method
   ACT : MIN;
   // Use 'max' accumulation method
   ACCU : MAX;
   RULE 1 : IF CueWords IS vLow AND FreqWords IS vLow AND SentencePosition is vLow
          THEN SentenceImportance IS POOR;
    RULE 2 : IF CueWords IS vLow AND FreqWords IS vLow AND SentencePosition is Low
                     THEN SentenceImportance is POOR;
 RULE 3 : IF CueWords IS vLow AND FreqWords IS vLow AND SentencePosition is High
                     THEN SentenceImportance is GOOD;
END_RULEBLOCK
END_FUNCTION_BLOCK
```

Figure 3.8: FCL definition for the Rule Block

Now, for implementing the FL in java, the FCL file is first loaded into the FIS. We've used Trapezoidal membership functions to show the input variables and the output results in graphical demonstration. Figure 3.9 shows the membership functions for (a) Frequent Words, (b) Cue Words, (c) Sentence Position. The graph illustrates the range in which the defined input variables can be very low, low and high. Figure 3.10 is the graphical representation of the output variable 'Sentence Importance' representing the range of getting the importance of sentence as either Poor or Good. Only Good sentences are included in the summary document.



(a)

(b)

(c)

Figure 3.9: Fuzzy Term Definition for Input Variables

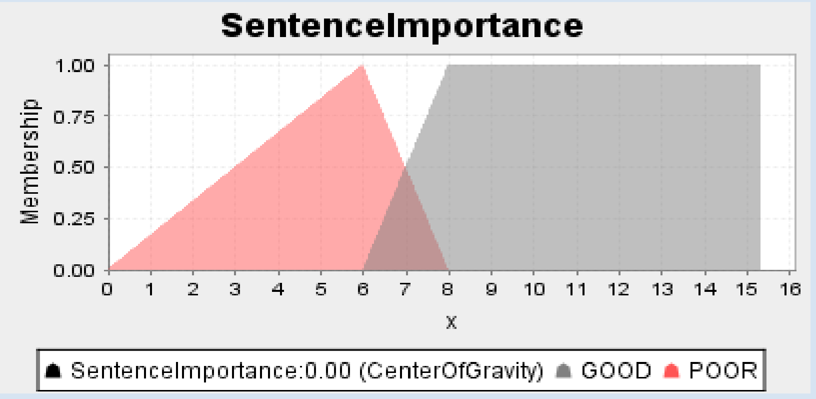
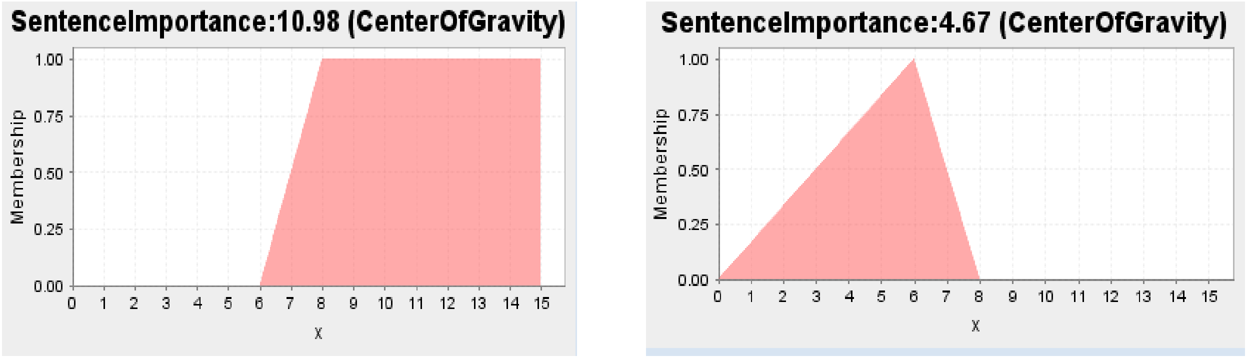


Figure 3.10: Graph of CoG for Sentence Importance.

Only the sentences with CoG of Sentence Importance greater than the threshold are selected to be included in summary. The threshold fixed here is 8.

In Figure 3.11, the CoG for sentence importance shown in (a) is 10.98, this means that the sentences with this value are included in the summary as from Figure 3.10 we can see that its range is Good while in Figure 3.11 (b), the CoG is 4.67, a value much less then the threshold and in the range of Poor sentences. Therefore the sentences with 4.67 CoG and less than 8 CoG are not included in the fuzzy summary document.



(a) Acceptable CoG

(b) Unacceptable CoG

Figure 3.11: CoG calculation for the Output variable ‘Sentence Importance’.

3.2.4 Optimized Fuzzy Model

After acquiring the fuzzy summary from FL method, the summary is further optimized to get a greater IR results in final summary as compared to the previous two summaries generated. Therefore, the system is extended with this novel method called OFM. The initial summary document generated by the GSM summarization technique is fed into the OFM along with the FCL file of FL method containing the input and output variables and the fuzzy rules set. An optimization process is performed on the FCL file and a new optimized FCL file is generated with the updated values of fuzzy terms and the updated weights for the fuzzy rules set, this new optimized values are than used to generate an optimized fuzzy model summary document. Figure 3.12 shows the OFM architecture.

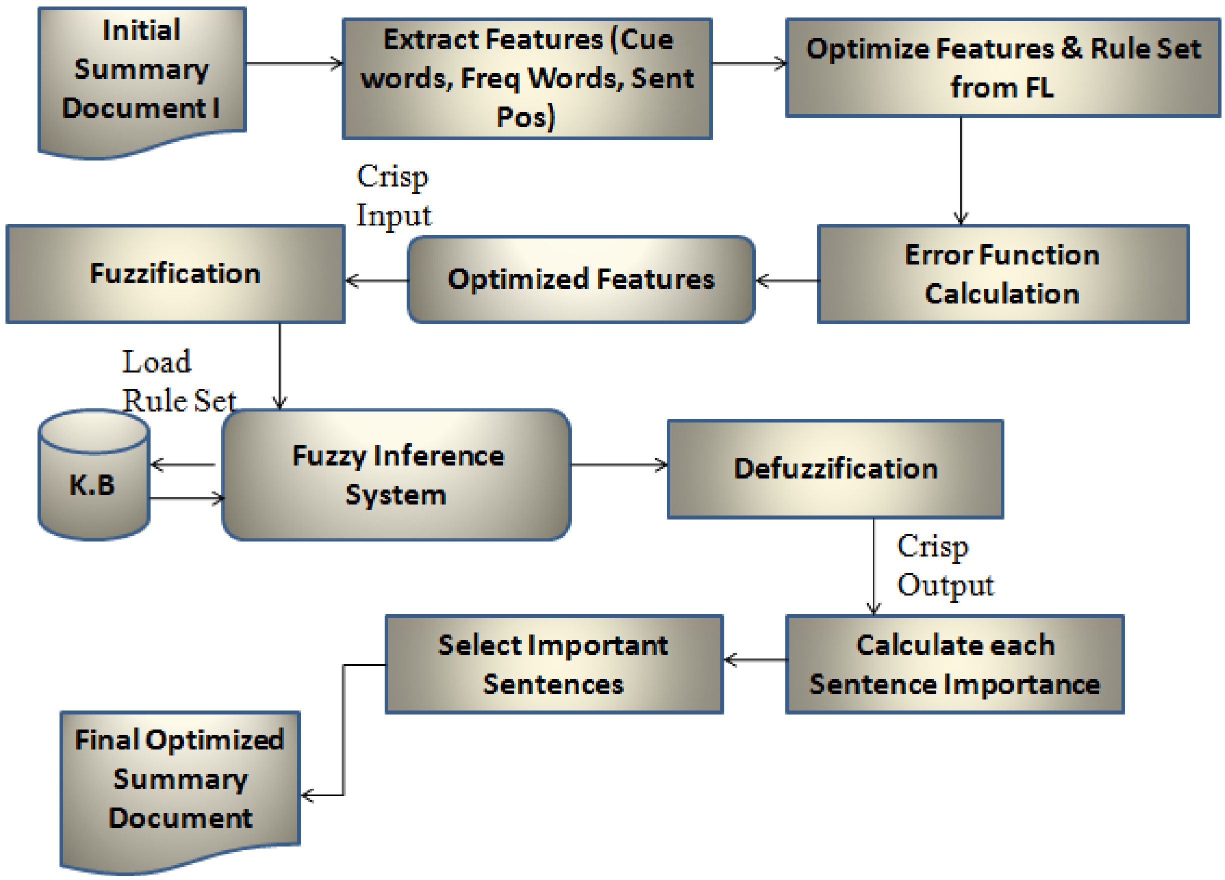

Initial Summary Document I
Extract Features (Cue words, Freq Words, Sent Pos)
Optimize Features & Rule Set from FL
Crisp Input
Fuzzification
Optimized Features
Error Function Calculation
Load Rule Set
K.B
Fuzzy Inference System
Defuzzification
Crisp Output
Select Important Sentences
Calculate each Sentence Importance
Final Optimized Summary Document

Figure 3.12: The Optimized Fuzzy Model Architecture for generating a High Precision Summary

3.2.4.1 Implementation Details

The OFM method is implemented in java by importing the package jfuzzylogic.jar. In this method, the FCL file named “features.fcl” containing the input and output variables along with their fuzzy terms and the fuzzy rule set is loaded into the Fuzzy Inference System. An error function is created to optimize the entered values and evaluate the fuzzy system using the Euclidean Formula given in equation 1.

```
double desiredSentImp = SentImpXL[frequencyIND][posIND] / 10.
error += (SentenceImportance - desiredSentImp) * (SentenceImportance - desiredSentImp) …(1)
```

Here the desired Sentence Importance is calculated by taking a desired sentence position (identified feature) values and the desired frequent words (identified feature) values manually and dividing them by a constant 10. Now, the error is calculated by the formula given in (1), where Sentence Importance is a variable containing the actual defuzzified value of sentence importance. The errors are calculated and new fuzzy term ranges are assigned and the weights are assigned to the fuzzy rules and a new optimized FCL file named "optimized_features.fcl" is generated. This optimized FCL file is now loaded to the FIS.

Figure 3.13 shows the structure of 'optimized_features.fcl' file with optimized fuzzy terms.

```
FUNCTION_BLOCK features

FUZZIFY CueWords

        TERM High :=  (1.0, 0.0) (2.0, 1.0) (3.0, 1.0) ;

        TERM Low :=  (0.0, 0.0) (1.0, 1.0) (2.0, 0.0) ;

        TERM vLow :=  (0.0, 1.0) (1.0, 0.0) ;

END_FUZZIFY

FUZZIFY FreqWords

        TERM High :=  (3.024, 0.0) (4.0, 1.0) (10.0, 1.0) ;

        TERM Low :=  (1.0, 0.0) (3.0, 1.0) (4.0, 0.831) ;

        TERM vLow :=  (0.0, 1.0) (2.0, 1.0) (3.0, 0.0) ;

END_FUZZIFY

FUZZIFY SentencePosition

        TERM High :=  (4.024, 0.0) (4.5120000000000005, 0.488) (5.0, 0.009999999999999998) ;

        TERM Low :=  (2.536, 0.018) (2.6569999999999996, 0.05199999999999998)
(4.7669999999999995,0.09099999999999997) (4.768, 0.09899999999999998) ;

        TERM vLow :=  (3.072, 1.0) (3.816, 0.776) (3.984, 0.8300000000000001) ;

END_FUZZIFY

DEFUZZIFY SentenceImportance

        TERM GOOD :=  (1.3920000000000003, 0.576) (3.3920000000000003, 1.0) (15.0, 1.0) ;

        TERM POOR :=  (0.248, 1.0) (0.8879999999999997, 0.248) (0.8959999999999995, 0.0) ;

        METHOD : COG;

        DEFAULT := 0.0;

        RANGE := (0.0 .. 15.0);

END_DEFUZZIFY
```

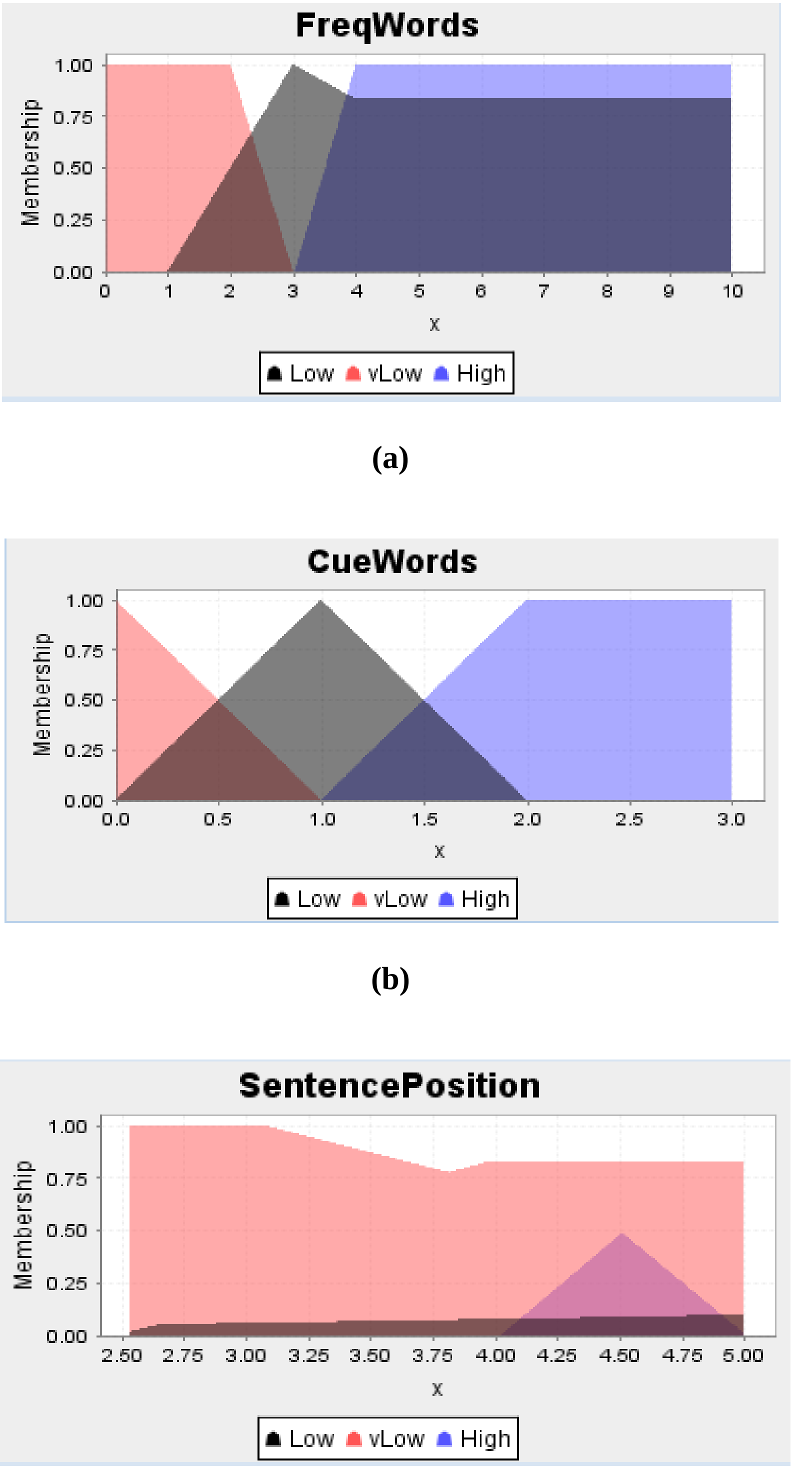
Figure 3.13: The Optimized FCL file with optimized Fuzzy Terms

There are several parametric optimization algorithms and the one used here is called the Optimization Delta Jump. After the optimization process ends, the fuzzy rule sets are optimized and a weight is assigned to few of them as displayed in the Figure 3.14.

```
RULEBLOCK No1

RULE 9 : IF ((CueWords IS High) AND (FreqWords IS vLow)) AND (SentencePosition IS High) THENSentenceImportance IS GOOD WITH 0.12999999999999998;

RULE 17 : IF ((CueWords IS High) AND (FreqWords IS Low)) AND (SentencePosition IS Low) THENSentenceImportance IS GOOD WITH 0.03999999999999998;

RULE 18 : IF ((CueWords IS High) AND (FreqWords IS Low)) AND (SentencePosition IS High) THENSentenceImportance IS GOOD WITH 0.029999999999999978;

RULE 26 : IF ((CueWords IS High) AND (FreqWords IS High)) AND (SentencePosition IS Low) THENSentenceImportance IS GOOD WITH 0.05999999999999997;

RULE 27 : IF ((CueWords IS High) AND (FreqWords IS High)) AND (SentencePosition IS High) THENSentenceImportance IS GOOD WITH 0.28;

END_RULEBLOCK

END_FUNCTION_BLOCK
```

Figure 3.14: The Optimized FCL file with weighted Rule Set.

The output result is generated according to these rules. For graphical demonstration of input and output variables, the OFM is implemented in Java. Figure 3.15 shows the optimized membership functions for (a) Frequent Words, (b) Cue Words, (c) Sentence Position. The graph illustrates the optimized range in which the defined input variables can be very low, low and high. Figure 3.16 is the graphical representation of the output variable 'Sentence Importance' representing the range of getting the importance of sentence as either Poor or Good. Only Good sentences are included in the summary document.



(a)

(b)

(c)

Figure 3.15: Optimized Fuzzy Term Definition for Input Variables

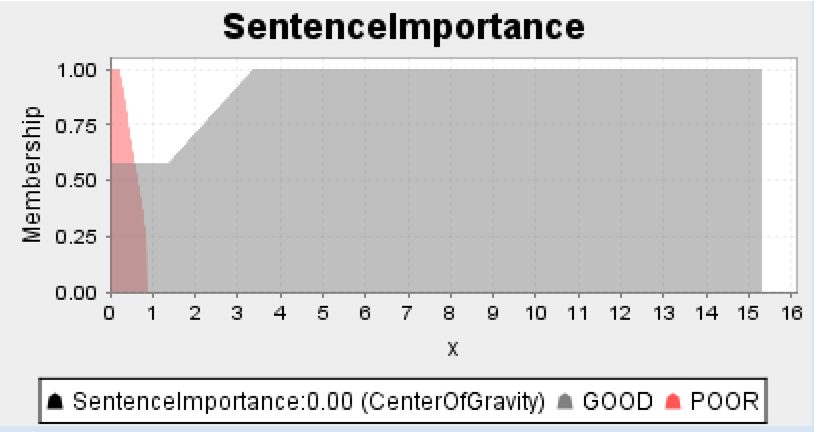
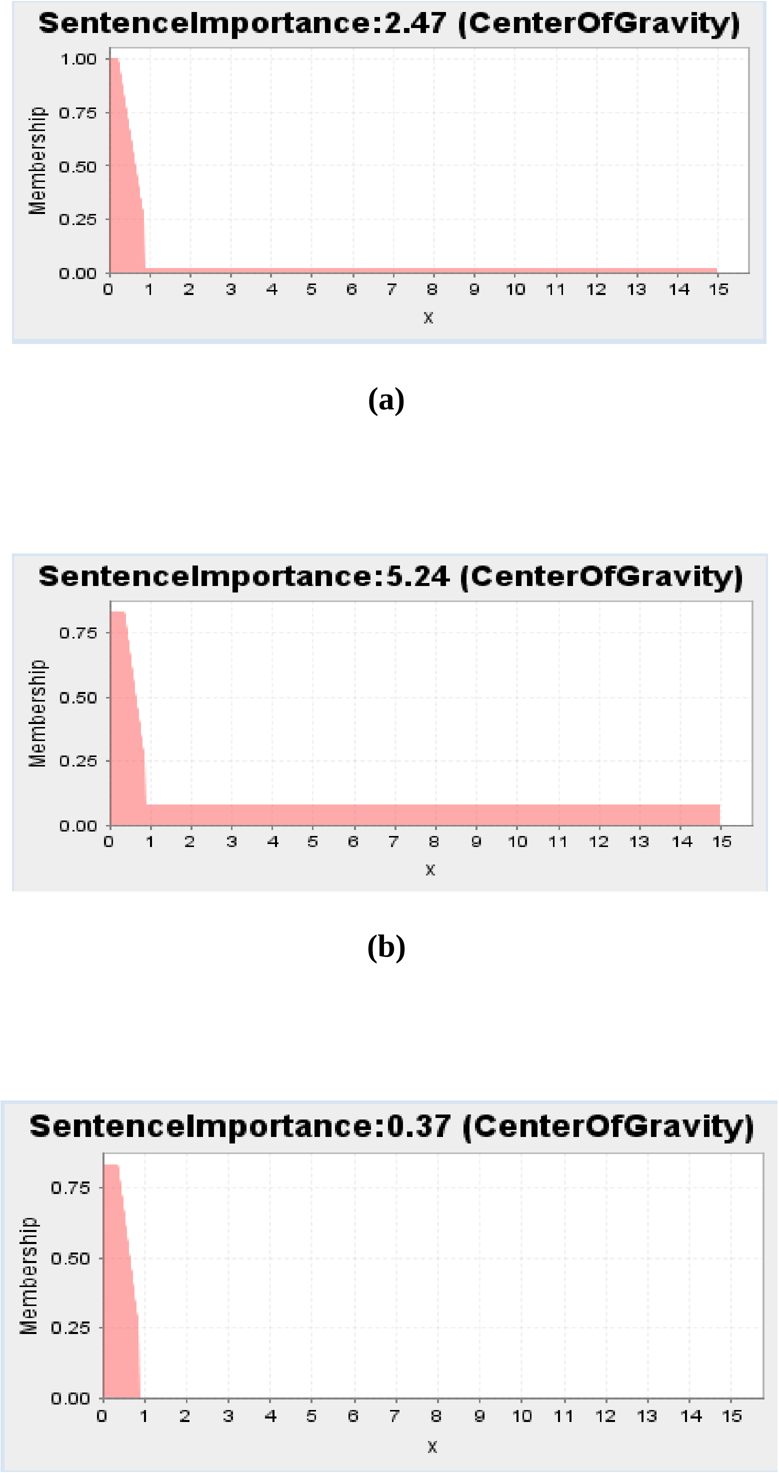


Figure 3.16: Graph of CoG for Optimized Sentence Importance.

Only the sentences with optimized CoG of Sentence Importance greater than the threshold are selected to be included in summary. The threshold fixed here is 1.

In Figure 3.17, the optimized CoG for each sentence importance as shown in (a) is 2.47 and (b) 5.24 means that the sentences with this value are included in the summary as from Figure 3.16 we can see that its range is Good while in Figure 3.17 (c), the CoG is 0.37, a value less then the threshold and in the range of Poor sentences. Therefore the sentences with 1 and less than 1 CoG are not included in the final optimized summary document.



(a)

(b)

(c)

Figure 3.17: Optimized CoG calculation for the Output variable ‘Sentence Importance’.

3.3 FLOW CHART OF PROPOSED SUMMARIZATION SYSTEM

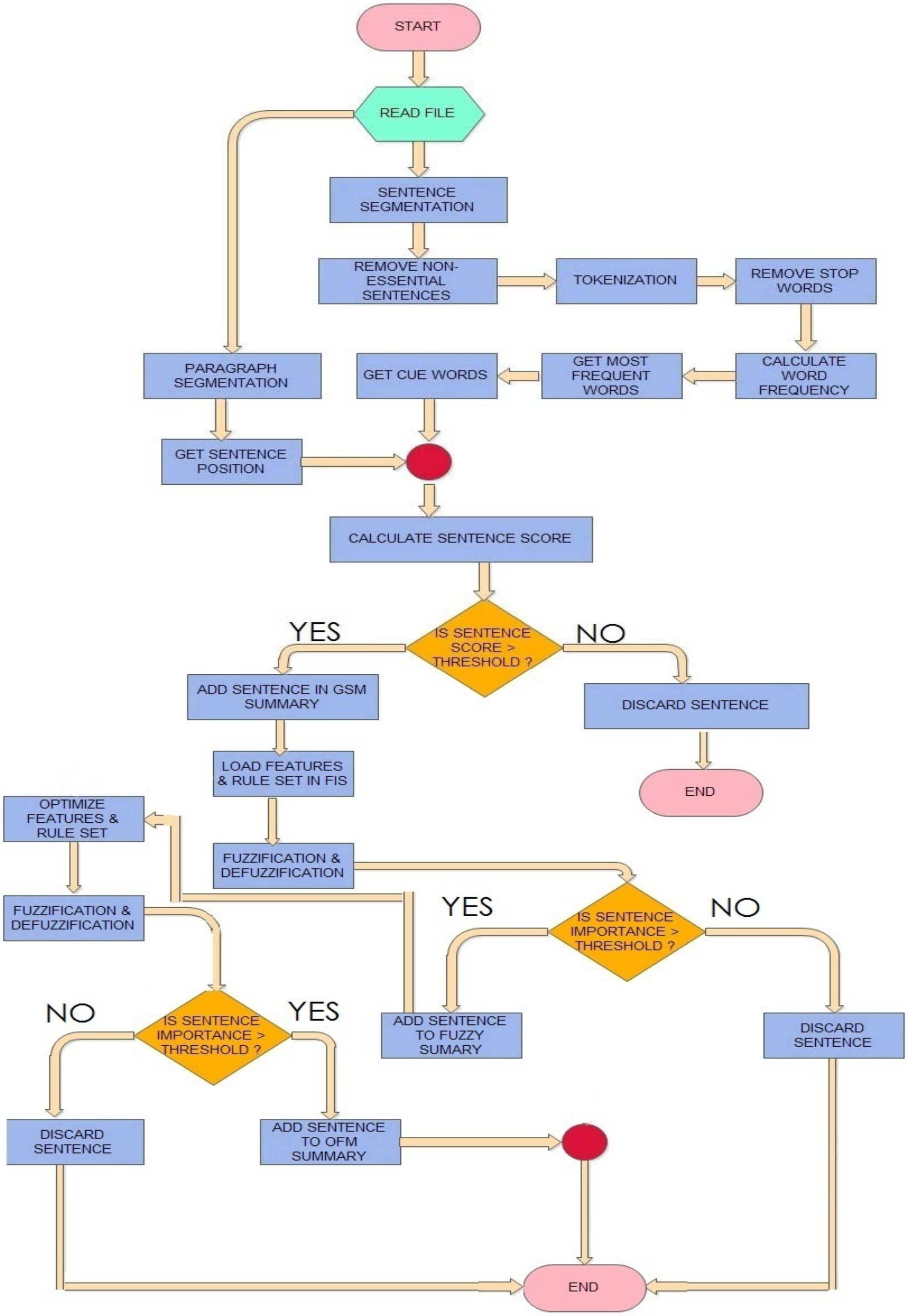

START
READ FILE
SENTENCE SEGMENTATION
REMOVE NON-ESSENTIAL SENTENCES
TOKENIZATION
REMOVE STOP WORDS
CALCULATE WORD FREQUENCY
GET MOST FREQUENT WORDS
GET CUE WORDS
PARAGRAPH SEGMENTATION
GET SENTENCE POSITION
CALCULATE SENTENCE SCORE
IS SENTENCE SCORE > THRESHOLD ?
YES
NO
ADD SENTENCE IN GSM SUMMARY
DISCARD SENTENCE
END
LOAD FEATURES & RULE SET IN FIS
FUZZIFICATION & DEFUZZIFICATION
IS SENTENCE IMPORTANCE > THRESHOLD ?
YES
NO
ADD SENTENCE TO FUZZY SUMARY
DISCARD SENTENCE
OPTIMIZE FEATURES & RULE SET
FUZZIFICATION & DEFUZZIFICATION
NO
IS SENTENCE IMPORTANCE > THRESHOLD ?
YES
DISCARD SENTENCE
ADD SENTENCE TO OFM SUMMARY
END

Figure 3.18: The Flowchart of the Proposed System.

3.4 SUMMARY

In this chapter the architecture and the implementation of our proposed novel automatic text summarization General Statistical Method, Fuzzy Logic and Optimized Fuzzy Model have been illustrated in detail, showing the different components and processes involved in each method and their implementation details. Optimized Fuzzy Model is designed as the enhanced model with good quality output with efficiency and accuracy. OFM is observed to be more efficient and accurate than GSM and FL methods in generating the summaries of documents more closely related to human generated summaries.

Chapter 4

EVALUATION AND RESULTS

4.1 INTRODUCTION

In this chapter, we evaluate the three summaries produced by the three text summarization methods we have implemented in our project, the General Statistical Method, the Fuzzy Logic Method and the Optimized Fuzzy Model. We have used two other very commonly used summarizers, the MS Word 2007 summarizer and the Essential summarizer for comparison and to prove that our proposed summarizer produces the summary of documents giving the best results. The Optimized fuzzy model which is the combination of GSM-FL-OFM is designed to give the best results among all. The OFM addresses various issues encountered by researchers in previous and recent work as described in chapter 3. OFM has considerably increased the effectiveness of IR when compared with other methods from our proposed approach as well as from other automatic summarizers, the Essential and the MS Word 2007 summarizer. To prove that OFM contributes towards generating high quality summary, the evaluation measurement is carried out using precision, recall and f-measure which are usually used for conventional information retrieval tasks.

The chapter is divided into five sections. Section 4.2 gives the description of the scientific articles which is the corpus of computing articles, used to summarize for evaluation purposes. Section 4.3 gives the performance evaluation of our summarizer with three methods and the other two summarizers with the results showing in their corresponding tables. In Section 4.4, the comparison of information retrieval metrics is shown in the form of line charts. Section 4.5 gives the overall comparison of the proposed summarizer and the other summarizers. Section 4.6 presents the overall graph of the summarization system to compute the final results. Finally, Section 5 gives the summary of the chapter.

4.2 SCIENTIFIC ARTICLES DATA SET

An online scientific articles database called Computation and Language (cmp-lg) collection has been used to get the articles in the category of Computer Science and summarize them using the automatic text summarizers. The documents are scientific papers which appeared in Association for Computational Linguistics (ACL) sponsored conferences. Initially, the articles from the online corpus are downloaded in a pdf format which is then converted into an editable word document format using an online pdf to word converter. Preprocessing is performed on each document to remove the unnecessary text from the document such as the author names, headers and footers, page numbers and end notes. We have tested our system with 20 documents (in .docx format). Here each document contains around 100-150 sentences. The summary document is in fact 30 percent of the original document. The tested 20 documents are presented in Table 4.1 with the title of the document, its size in kilobytes and the total number of sentences each document contains.

Document No.	**Document Title**	**No. of Sentences**	**Size in KB**
1	Model Based Software Development: Issues & Challenges.	148	109
2	A Unified CBR Approach for Web Services Discovery, Composition and Recommendation.	116	52
3	Efficient Approach Towards an Agent-Based Dynamic Web Service Discovery Framework with QoS Support.	127	52
4	On the Role of Evolvability for Architectural Design.	124	90
5	Integrations with case-based reasoning.	52	68
6	Discovery.	118	62
7	Data Mining in Web Services Discovery and Monitoring.	40	29
8	A systematic study of software quality models.	96	68
9	The effectiveness of using project management tools and techniques for delivering projects.	106	208
10	Extreme Programming and Embedded Software Development.	324	60
11	Column-oriented Database Systems.	68	40
12	End User Software Engineering	60	40
13	A Trust Based Access Control for Web Services.	175	64
14	Why Specification Workshop Works?	102	42
15	Functional Programming: Why Should You Care?	72	37
16	Introduction to Elemental Design Patterns.	186	45
17	Four Principles of Low-Risk Software Releases.	112	128
18	Want to be Agile? Learn to Fail!	79	38
19	Software In security: Third-Party Software and Security.	96	44
20	How to Build a Strong Virtual Team.	94	41

Table 4.1: The Testing Document Set.

4.3 PERFORMANCE EVALUATION

The results of the summarizers are evaluated using precision, recall and f-measure measurements [10]. Precision (P) and Recall (R) are the standard metrics for retrieval effectiveness in information retrieval. P is the number of correct results divided by the number of all returned results and R is the number of correct results divided by the number of results that should have been returned. In our project, they calculated as follows [11]:

$P = tp / (tp + fp)$ ….. (4.1)

$R = tp / (tp + fn)$ ….. (4.2)

Where tp = true positive, represents the sentences selected by the software system which are also found in the manual summary.

fp = false positive, shows the sentences selected by the software are not found in the manual summary and

fn = false negative, the sentences in manual summary are not found in software system.

F-measure is used in the area of information retrieval for measuring certain tasks related to search, document and query classification.

In our project, we have used f-measure to evaluate the experiment’s accuracy as it considers both the precision P and recall R of the experiment to compute the score.

The F-measure is defined as the weighted average of precision and recall. The traditional f-measure is the harmonic mean of the precision and recall. It is calculated as follows:

$F = 2 * \text{precision} . \text{recall} / \text{precision} + \text{recall}$, …….. (4.3)

Where f-measure gives its best value when the result equals to 1 and its worst value when the result equals 0.

Next, we will evaluate the three methods GSM, FL and OFM implemented in our thesis on the dataset of 20 computing articles using the information retrieval metrics Precision, Recall and F-measure to assess which one gives the better results.

4.3.1 Evaluation of GSM

The Table 4.2 shows the results after evaluating the summaries generated by GSM method. The Table shows the comparison of the sentences extracted automatically using GSM method with the manually extracted sentences from text also called the gold-standard summary and finding out the matched and un-matched ones. The table illustrates that document 2 and document 1 gives the best precision which is above 80 percent, this is due to the reason that system generated sentences are fewer in number when compared with manually generated sentences, meaning that the summary doesn’t contain any irrelevant information that’s why it is giving best results. Whereas in document no 4, 15 and 18, it is observed that the automatically extracted sentences are far greater than the manual ones meaning it contains a lot of irrelevant data, which becomes the reason to lower the precision.

Doc. No.	**Automatic (tp+fp)**	**Manual (tp+fn)**	**Matched (tp)**	**Un-Matched (fn)**	**Precision (%)**	**Recall (%)**	**F-measure**
1	49	45	34	11	69.38	75.56	72.34
2	21	37	18	19	85.71	48.65	62.07
3	28	42	21	21	75.00	50.00	60.00
4	56	41	27	14	48.21	65.85	55.67
5	32	23	19	2	59.37	82.61	69.10
6	35	45	29	16	82.86	64.44	72.50
7	21	18	16	2	76.20	88.89	82.05
8	32	35	28	7	87.50	80.00	83.58
9	28	31	19	12	67.86	61.29	64.41
10	109	112	76	36	69.72	67.86	68.78
11	28	23	21	2	75.00	91.30	82.35
12	26	18	14	4	53.84	77.78	63.63
13	90	67	49	18	54.44	73.13	62.42
14	34	38	31	7	91.17	81.58	86.12
15	48	25	22	3	45.83	88.00	60.27
16	56	36	31	5	55.35	86.12	67.39
17	54	25	16	4	29.62	64.00	40.51
18	51	31	25	3	49.02	80.64	60.97
19	44	28	21	7	47.72	75.00	58.33
20	65	56	43	6	66.15	76.78	71.07
AVERAGE					64.50	73.97	67.17

Table 4.2: The Evaluation Results of Summaries generated by GSM.

4.3.2 Evaluation of FL

Next, we have evaluated the summaries generated by FL method. The Table 4.3 shows the results with automatic, manual, matched and un-matched sentences for getting the precision, recall and f-measure of each document and the overall total results. With the highest precision, the documents are document no. 2, 6, 7, 8, and 14. Here we can see the number of documents giving precision above 80 percent have been increased in comparison to our previously defined GSM technique. This evaluation shows that the summary should be compressed to about 35-40 percent so that it should only contain the important information. Document no. 4 and 17, demonstrates the worst precision which is around 40 percent as it gives greater number of automatically extracted sentences than is required.

4.3.3 Evaluation of OFM

The summaries generated by OFM are actually generated by a hybrid GSM+FL+OFM approach. The evaluation results of OFM summarizer is presented in Table 4.4. By applying this model, the precision of some documents have reached above 90 percent. These documents include the document number 2, 7, 8, 10, 12, 14 and 16 and the least precision it shows is 53 percent only in one document which is document no. 17. This illustrates that the summary is in fact optimized.

Doc. No.	Automatic (tp+fp)	Manual (tp+fn)	Matched (tp)	Un-Matched (fn)	Precision (%)	Recall (%)	F-measure
1	44	45	31	14	70.45	68.89	69.66
2	21	37	18	19	85.71	48.65	62.07
3	26	42	20	22	76.92	47.62	58.82
4	45	41	19	22	42.22	46.34	44.18
5	27	23	19	4	70.37	82.60	76.00
6	25	45	21	24	84.00	46.67	60.00
7	18	18	16	2	88.89	88.89	88.89
8	32	35	28	7	87.50	80.00	83.582
9	21	31	15	16	71.43	48.38	57.69
10	89	112	68	44	76.40	60.71	67.66
11	27	23	20	3	74.07	86.96	80.00
12	16	18	9	9	56.25	50.00	52.94
13	79	67	45	22	56.96	67.16	61.64
14	23	38	20	9	86.95	52.63	65.57
15	34	25	17	8	50.00	68.00	57.62
16	49	36	28	8	57.14	77.78	65.88
17	38	25	16	9	42.10	64.00	50.79
18	33	31	20	11	60.60	64.51	62.50
19	35	28	18	10	51.43	64.28	57.14
20	51	56	40	16	78.43	71.42	74.76
AVERAGE					68.40	64.28	64.87

Table 4.3: The Evaluation Results of Summaries generated by GSM+FL.

Doc. No.	Automatic (tp+fp)	Manual (tp+fn)	Matched (tp)	Un-Matched (fn)	Precision (%)	Recall (%)	F-measure
1	39	45	32	13	82.05	71.11	76.19
2	20	37	18	19	90.00	48.65	63.16
3	24	42	20	22	83.33	47.62	60.60
4	34	41	26	23	76.47	63.41	69.33
5	27	23	19	4	70.37	82.60	76.00
6	33	45	29	16	87.87	64.44	74.36
7	14	18	13	6	92.86	72.22	81.25
8	31	35	28	7	90.32	80.00	84.84
9	22	31	18	15	81.81	58.06	67.92
10	74	112	66	46	89.19	58.93	70.96
11	24	23	21	2	87.5	91.30	89.36
12	15	18	14	4	93.34	77.78	84.84
13	72	67	47	20	65.27	70.15	67.62
14	34	38	31	7	91.17	81.58	86.11
15	30	25	21	4	70.00	84.00	76.36
16	32	36	29	7	90.62	80.56	85.29
17	28	25	15	10	53.57	60.00	56.60
18	30	31	24	7	80.00	77.42	78.68
19	23	28	18	12	78.26	64.28	70.58
20	52	56	42	14	80.77	75.00	77.78
AVERAGE					81.74	70.46	74.89

Table 4.4: The Evaluation Results of Summaries generated by OFM.

4.3.4 Evaluation of MS Word 2007 Summarizer

Next, we have evaluated the summaries of documents generated by the Microsoft word 2007 summarizer by comparing its results with our summarizer system to calculate its retrieval effectiveness. Table 4.5 shows the results for evaluating the summaries generated by MS Word 2007 summarizer. The precision of documents goes down to as worst as 30 percent. Document no. 5, 8 and 13 gives the precision results of 33 percent and 35 percent both as the number of un-matched sentences are double the matched ones. Its best precision only goes to about 60 percent of document no. 7.

4.3.5 Evaluation of Essential Summarizer

Essential Summarizer is user friendly summarization software that efficiently summarizes text document and web pages. We have used this summarizer to compare our summarization system and measure its effectiveness in comparison to this system. Table 4.6 shows the results of summaries evaluation generated by Essential summarizer. Here document number 6 and 7 gives precision above 80 percent that’s means this summarizer is much better than MS Word summarizer 2007 as its worst precision is 45 percent of document no. 4.

Doc. No.	**Automatic (tp+fp)**	**Manual (tp+fn)**	**Matched (tp)**	**Un-Matched (fn)**	**Precision (%)**	**Recall (%)**	**F-measure**
1	50	45	23	22	46.00	51.11	48.42
2	45	37	18	19	40.00	48.65	43.90
3	46	42	20	22	43.47	47.62	45.45
4	42	41	16	25	38.09	39.02	38.55
5	24	23	8	15	33.34	34.78	34.04
6	46	45	27	18	58.69	60.00	59.34
7	20	18	12	6	60.00	66.67	63.17
8	49	35	17	18	34.69	48.57	40.47
9	35	31	14	17	40.00	45.16	42.42
10	114	112	42	70	36.84	37.50	37.16
11	26	23	12	11	46.15	52.17	48.97
12	20	18	8	10	40.00	44.44	42.10
13	69	67	24	43	34.78	35.82	35.29
14	34	38	19	19	55.88	50.00	52.78
15	30	25	16	9	53.33	64.00	58.18
16	35	36	17	19	48.57	47.22	47.88
17	26	25	12	13	46.15	48.00	47.05
18	28	31	16	14	57.14	51.61	54.23
19	30	28	12	16	40.00	42.85	41.38
20	50	56	28	28	56.00	50.00	52.83
AVERAGE					45.45	48.26	46.68

Table 4.5: The Evaluation Results of Summaries generated by MS Word 2007.

Doc. No.	**Automatic (tp+fp)**	**Manual (tp+fn)**	**Matched (tp)**	**Un-Matched (fn)**	**Precision (%)**	**Recall (%)**	**F-measure**
1	43	45	22	23	51.16	48.88	50.00
2	36	37	23	14	63.88	62.16	63.01
3	52	42	38	4	73.07	90.47	80.85
4	49	41	22	19	44.89	53.65	48.89
5	25	23	17	6	68.00	73.91	70.83
6	33	45	28	17	84.84	62.22	71.79
7	11	18	9	9	81.81	50.00	62.07
8	40	35	30	5	75.00	85.71	80.00
9	37	31	26	5	70.27	83.87	76.47
10	80	112	53	59	66.25	47.32	55.20
11	22	23	16	7	72.72	69.56	71.11
12	16	18	11	7	68.75	61.12	64.70
13	72	67	38	29	52.77	56.72	54.67
14	49	38	29	9	59.18	76.31	66.66
15	15	25	11	14	73.33	44.00	55.00
16	32	36	19	17	59.37	52.78	55.88
17	20	25	7	18	35.00	28.00	31.11
18	21	31	13	18	61.90	41.93	50.00
19	36	28	25	3	69.44	89.28	78.12
20	50	56	32	24	64.00	57.14	60.37
AVERAGE					64.78	61.75	62.33

Table 4.6: The Evaluation Results of Summaries generated by Essential Summarizer.

4.4 COMPARISON OF INFORMATION RETREIVAL METRICES

In this section, we have compared the percentage of information retrieval metrics, the precision, recall and the f-measure for the two benchmark summarizers and the three methods of our summarizer on the basis of 20 documents. A line chart is used to understand the effectiveness of all techniques with no trouble.

4.4.1 GSM

The line graph in Figure 4.1 clearly shows that recall measure of the documents is at its highest and the precision is lowest while the f-measure comes in between them. Only one document reached the precision of more than 90 percent. It is also observed that both recall and precision gives results at two opposite extreme that means the number of sentences extracted using GSM summarizer are higher than the human generated sentences for summary. These sentences should be between 30-40% only to include the core information and exclude the unnecessary details.

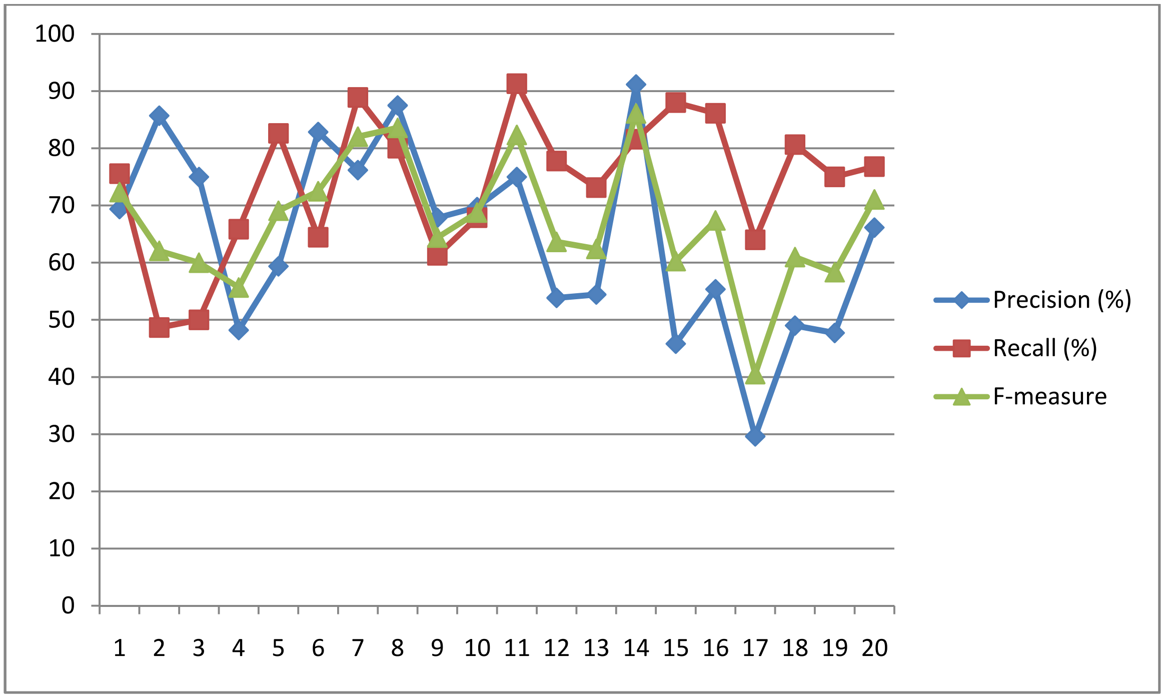


Figure 4.1: Retrieval effectiveness of GSM.

4.4.2 FL

In Figure 4.2, we can see that the three matrices are going almost equal for all documents. The precision is somewhat higher than the other two. The baseline is above 40 percent.. Still giving satisfactory results as the number of sentences extracted by GSM are refined and condensed to give a good proportion of document summary. But we want to find even better results.

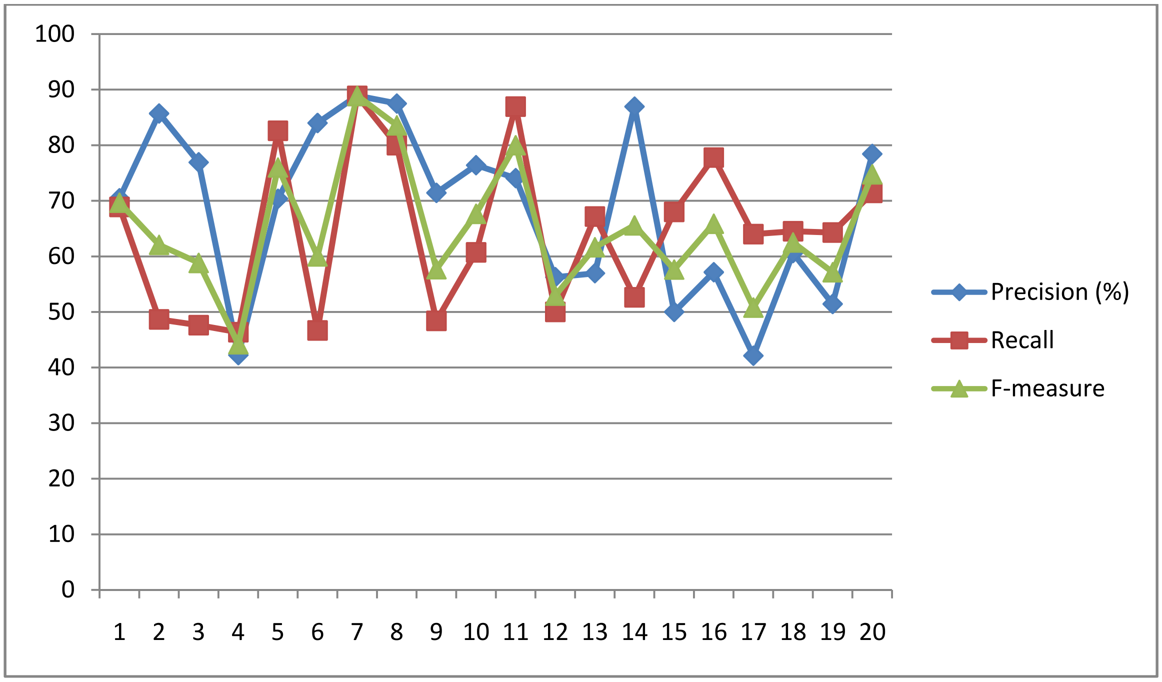


Figure 4.2: Retrieval effectiveness of FL.

4.4.3 OFM

The results we got from GSM are further enhanced using a hybrid approach, GSM-OFM approach. Figure 4.3 presents that all three metrics have values above 50 percent. The highest precision of documents is shown to be between 50 to above 90 percent, the precision gets below 60 percent only for one document and below 70 percent only for 2 documents. Precision is higher than 80 percent for most documents. As we can see only two documents give recall below 50 percents, which makes it a little lesser in recall measure when compared with our GSM method. Overall, this method is among the best one in giving good results.

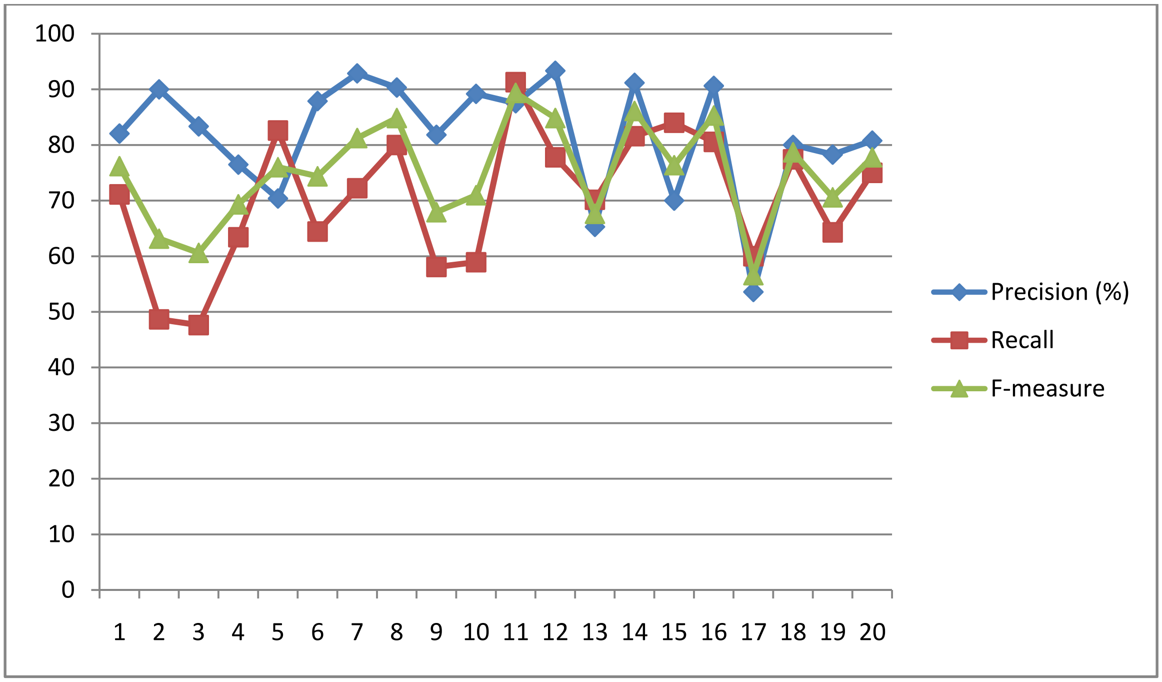


Figure 4.3: Retrieval effectiveness of OFM.

4.4.4 MS WORD 2007 Summarizer

In case of MS word summarizer, the precision, recall and f-measure goes below 40 percent and not above 70 percent as presented in Figure 4.4. The results of all three metrics are almost similar which is between 40 to 50 percent for most of the documents so we can clearly say that the summary produced by this summarizer is not of good quality as its results are very far from human generated summary results.

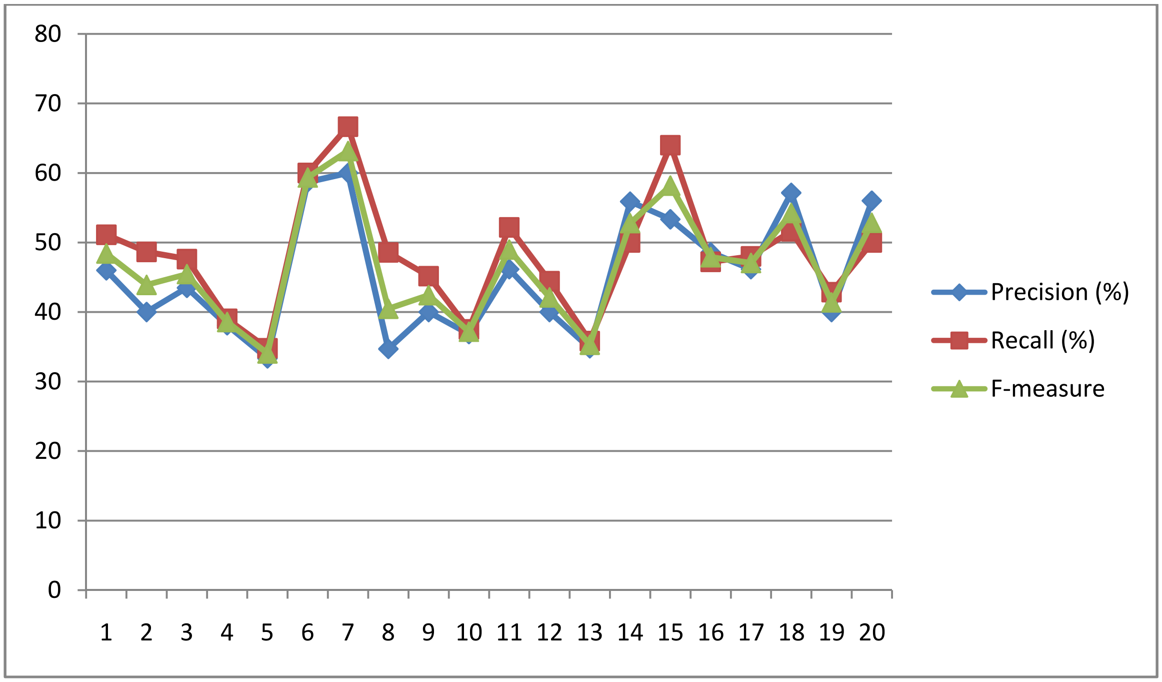


Figure 4.4: Retrieval effectiveness of MS Word.

4.4.5 Essential Summarizer

Figure 4.5 shows that recall measure of documents is as highest as up to 90 percent and for one document it goes as lowest to less than 30 percent, we can see a huge diversity in the range which makes this summarizer a satisfactory one.

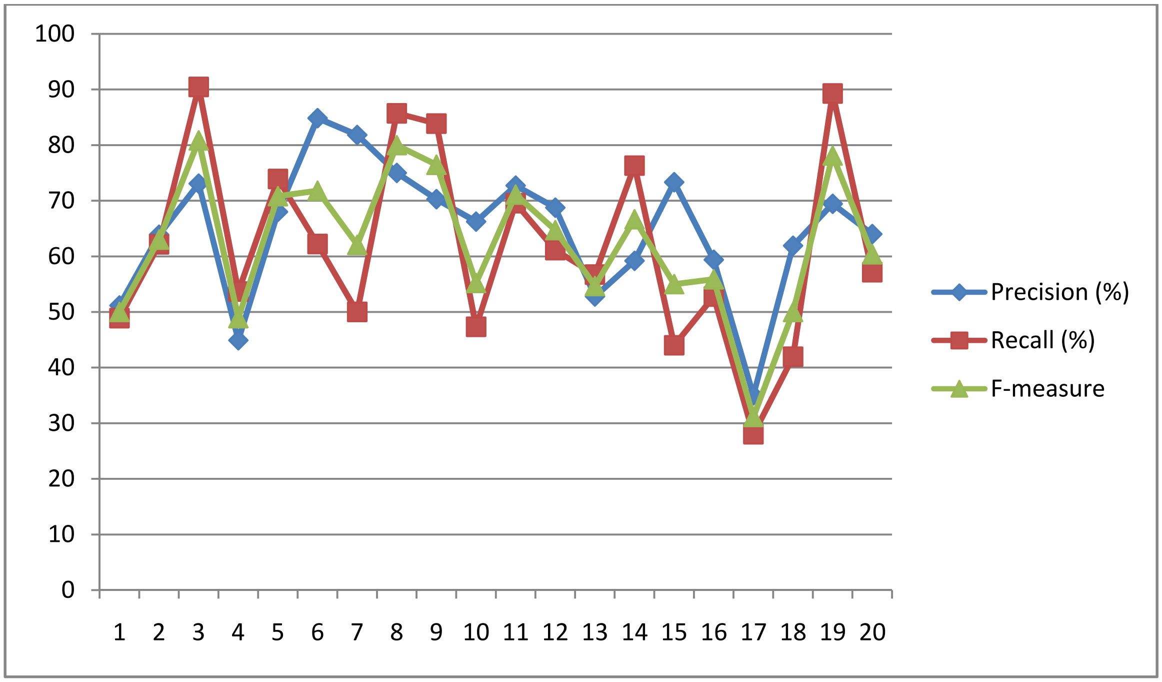


Figure 4.5: Retrieval effectiveness of Essential Summarizer.

Next, to prove the theory we made from the line charts we have to make comparison of the summarizers which is highlighted in next section.

4.5 COMPARISON OF SUMMARIZERS.

Based on the evaluation of 20 documents, we have measured the results of experiment. We have selected two standard summarizers used to compare with three methods of our summarization system, the Microsoft word 2007 and the Essential Summarizer. The results of all summarizers are compared with the gold standard summaries which are the human generated summaries. The compression rate for all summarization systems are kept between 30-40%. Table 4.7 presents the comparison of the average precision, recall and f-measure score between the three proposed methods and the Microsoft Word 2007 summarizer and Essential Summarizer.

SUMMARIZER	**Avg. Precision (%)**	**Avg. Recall (%)**	**Avg. F-measure (%)**
GSM	64.50	73.97	67.18
FL	68.39	64.28	64.87
OFM	81.74	70.45	74.89
MS-WORD	45.45	48.26	46.68
ESSENTIAL	64.78	61.75	62.33

Table 4.7: The comparison of average precision, recall and f-measure score of different summarizers for 20 documents.

According to Table 4.6, the end results of all 20 documents from computer science corpus are presented. On the basis of sentence extraction in text summarization, sentence features are considered very important. To generate a good quality summary document, the significant features for text summarization are identified on the basis of which the important sentences are extracted.

In our project, first the statistical model, the GSM method is examined which is a selection of sentences from the original text without any linguistic resources. As we can see from the Comparison Table, the results of average recall of GSM is the highest with 73.97% among other methods and summarizers, which is considered to be closest to human generated summary and giving better results but the average precision of GSM is quite less than that of essential summarizer and this limitation has overcome through the implementation of OFM. The fuzzy logic method has tried to improve the results of GSM and reached the average precision of 68.39%, higher than the precision of GSM method. As we want to improve the results, we have optimized the FL method in form of OFM and it proves to give the best average precision of 81.74% and f-measure 74.89% with the second highest in average recall of 70.45% as compared with other summarizers so we can say that OFM has improved the quality of summary similar to human generated summaries. The MS-word summarizer has investigated to capture fairly low precision, recall and f-measure results.

The results of the experiment in Figure 4.6 confirm that the optimized fuzzy model has a significant improvement quality of text summarization. The results of this method are the best as compared with the other proposed methods and other two benchmark summarizers. Therefore, it is declared that the results obtained by the OFM method of our summarizer persistently associates very much with human generated summaries and has high precision, recall and f-measure with the evaluation results.

4.6 COMPARISON GRAPH

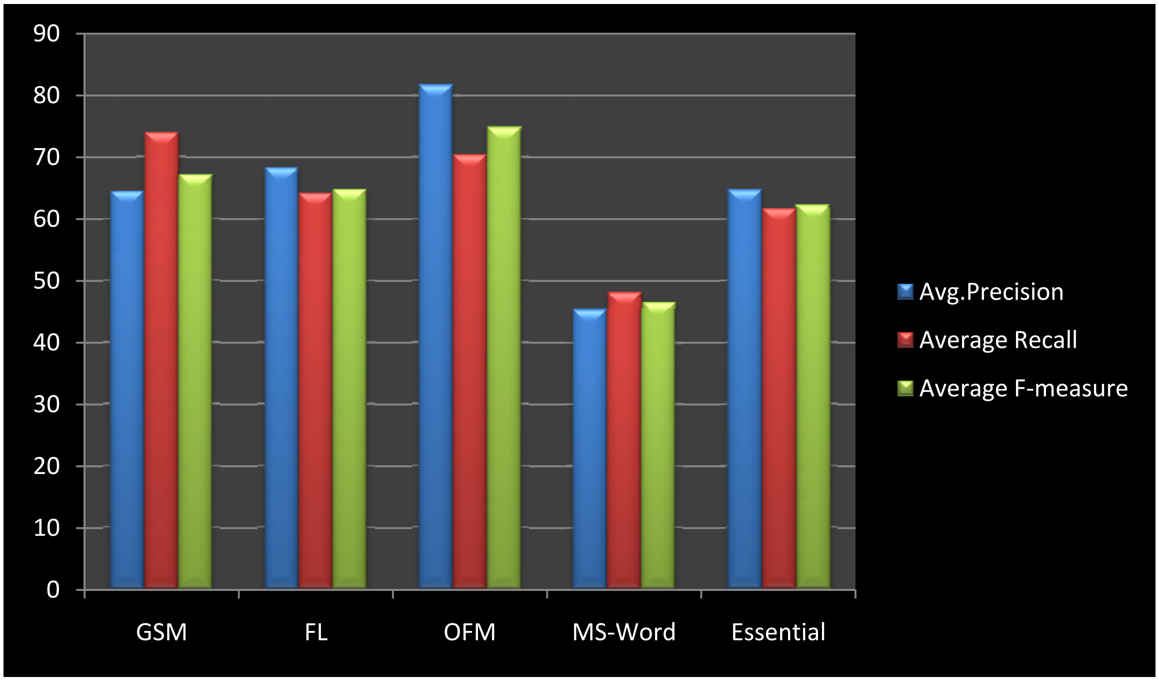


Figure 4.6: The comparison of average precision, recall and f-measure scores of difference summarizer 20 documents (30-40% compression rate).

4.7 COMPARISON WITH LITERATURE

Table 4.7 shows the comparison of our proposed system with three recent papers from literature in terms of average accuracy. L. Suanmali [1] is the paper of 2009 for automatic summary extraction in Fuzzy Logic domain tested on 30 documents from DUC2002 database gives only 49 percent accuracy. Hashemi [11**],** the paper of 2010 using sentence score calculation tested on 10 documents from corpus of General Science, DB, Image Processing and AI gives 70 percent precision and R.C. Balabantary [16] paper of 2012 using statistical approach tested on 10 docs shows 72 percent of accuracy. Although, the parameters are different for the data set used and the total number of documents being summarized but our proposed method has outperformed by giving 9 percent more accuracy from the recent [16] paper with 82 percent of avg. precision.

Method	**Avg. Accuracy (%)**
L. Suanmali [1]	49
Hashemi [11]	70
R.C. Balabantary [16]	72
Proposed Method	81

Table 4.8: The comparison of average accuracy with literature.

4.8 SUMMARY

In order to fully analyze the effectiveness of Optimized Fuzzy Model, it has been evaluated with proposed two other methods and the two different benchmark summarizers using 20 documents. The profound study of these summarizers has clearly shown that the three methods of our summarization system perform better than the MS-Word summarizer. The Essential summarizer shows good precision results when compared with GSM but the OFM has outperformed all the methods and the summarization system in giving the best results.
Our system gives the best results when compared with others because of the specific features used i.e. cue words, frequent words and sentence position. In the third method, the OFM, our focus is mainly on the first two features, the cue words and frequent words and thus it provides the relevant summary document due to the features used and secondly the removal of non-essential sentences from text as we clearly know the sentences containing some words are considered to be having the unnecessary details, which shouldn't be included in summary.

Chapter 5

CONCLUSIONS AND FUTURE WORK

Automatic Text Summarization is the area being explored from more than fifty years and it is gaining much importance with time, many researchers have presented several different individual and hybrid techniques to solve the information overload problem from diverse ways which is increasing as the time is passing. The idea to extract only relevant information from huge text repositories has become the need of every user browsing the web as well as studying the core ideas of any scientific article. Researches have been successful in implementing the system for document text summarization but mostly the research focuses on getting the effective results of information retrieval with high precision, recall and f-measure which means getting a high quality summary of document which is closest to human generated summaries.

5.1 CONCLUSIONS

In our research work, we have used a sentence extraction method to extract the important sentences from document to make a good quality summary based on three sentence features, the cue words, the frequent words and the sentence position. We have proposed a novel hybrid approach using three methods. The system we have implemented functions as follows, first, the important sentences are selected using the GSM method by assigning scores to each sentence based on the identified features. The summary generated by GSM is improved using another method, the fuzzy logic method. This combined approach gives slightly better results than the GSM only. We have optimized the FL method by optimizing the sentence features and fuzzy rule sets and measured the results, which proves to give the best result when compared with summaries generated by GSM and GSM-FL method. The Optimized Fuzzy Model is the proposed novel model for automatic text summarization which is efficient as well as effective and a blend of GSM-FL-OFM. OFM when compared with other benchmark summarizers such as MS-Word and Essential summarizer, gives the best quality results.

To prove the efficiency of our model, it has been implemented in NetBeans IDE using java and jfuzzylogic library. The execution results proves that our summarization system gives best end result to acceptable quality.

5.2 FUTURE WORK

Even though, OFM has been tested and evaluated comprehensively and it provides good results as well as considered a complete approach to get a good quality summary but still the system can be further improved for larger text data including the thesis reports and scientific journals or for generating summaries from multiple documents.

The sentence extraction features can also be increased to get more optimized results that can help to reach the precision measure up to 98 percent with the highest recall and f-measure values.

Although, the extractive summarization system produces a meaningful summary but still the extraction of sentences from document results in cohesion problem. This can only be solved if we can understand the semantic meanings of each sentence and than re-write the sentence in understandable words. In other words, an abstractive summarization method should be considered.

REFERENCES

[1] L. Suanmali, N. Salim, M. S. Binwahlan, "Feature-Based Sentence Extraction Using Fuzzy Inference Rules," Proceedings of the 2009 International Conference on Signal Processing Systems, 2009.

[2] M. Chandra, V. Gupta, S. Kr. Paul, "A Statistical approach for Automatic Text Summarization by Extraction," Proceedings of the 2011 International Conference on Communication Systems and Network Technologies, 2011.

[3] Kiani A. and Akbarzadeh, M.R. 2006. Automatic Text Summarization Using: Hybrid Fuzzy GA-GP. In Proceedings of 2006 IEEE International Conference on Fuzzy Systems, Sheraton Vancouver Wall Center Hotel, Vancouver, BC, Canada. 977-983, 2006.

[4] J. Flores, G. de Chalendar, "Syntactico-Semantic Analysis: a Hybrid Sentence Extraction Strategy for Automatic Summarization," Proceedings of the 2008 Seventh Mexican International Conference on Artificial Intelligence, 2008.

[5] L. Suanmali, N. Salim, M. S. Binwahlan, "Genetic Algorithm based Sentence Extraction for Text Summarization," Ministry of Science, Technology and Innovation under E-Science grant 01-01-06-SF0502, Malaysia, 2006.

[6] L. Suanmali, N. Salim, M. S. Binwahlan, "Fuzzy Genetic Semantic Based Text Summarization," Proceedings of the 2011 IEEE Ninth International COnference on Dependable, Autonomic and Secure Computing, 2011.

[7] Maher Jaoua, Abdelmajid Ben Hamadou, "Automatic Text Summarization of Scientific Articles Based on Classification of Extract's Population."Proceedings of the 4th international conference on Computational linguistics and intelligent text processing Pages 623-634, Springer-Verlag Berlin, Heidelberg, 2003.

[8]Jiaming Zhan , Han Tong Loh, Ying Liu, "Gather customer concerns from online product reviews – A text summarization approach,"Published in Journal, Expert Systems with Applications: An International Journal archive Volume 36 Issue 2, March, 2009 Pages 2107-2115, Pergamon Press, Inc. Tarrytown, NY, USA, 2009.

[9] M.Suneetha, S. Sameen Fatima, "Corpus based Automatic Text Summarizatio System with HMM Tagger," International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-1, Issue-3, July 2011.

[10] Suneetha Mannem, Shaik Mohammed Zaheer Pervez, Dr. S. Sameen Fatima, "A Novel Automatic Text Summarization System with Feature Terms Identification." Proceedings of the India Conference (INDICON), 2011 Annual IEEE, 16-18 Dec. 2011, pp 1-6.

[11] Rafeeq Al-Hashemi, Text Summarization Extraction System (TSES) Using Extracted Keywords, International Arab Journal of e-Technology, Vol. 1, No. 4, June 2010 pp 164-168.

[12] Naresh Kumar Nagwani, Dr. Shrish Verma, "A Frequent Term and Semantic Similarity based Single Document Text Summarization Algorithm" Proceedings of the International Journal of Computer Applications (0975 – 8887) Volume 17– No.2, March 2011.

[13] Vishal Gupta, Gurpreet Singh Lehal, "A Survey of Text Summarization Extractive Techniques," JOURNAL OF EMERGING TECHNOLOGIES IN WEB INTELLIGENCE, VOL. 2, NO. 3, AUGUST 2010.

[14] Hiroshi ISHII, Rihua LIN, Teiji FURUGORI, "A System for Text Summarization Based on Word Importance Measures," 2001.

[15] Daniel Mallett, James Elding, Mario A. Nascimento, "Information-Content Based Sentence Extraction for Text Summarization." Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04) Volume 2-Page 214, 2004.

[16] R.C. Balabantaray, D.K. Sahoo, B. Sahoo, M. Swain, "Text Summarization using Term Weights," International Journal of Computer Applications (0975 – 8887) Volume 38– No.1, January 2012.

[17] Ehsan Shareghi, Leila Sharif Hassanabadi, "Text Summarization with Harmony Search Algorithm-Based Sentence Extraction", Proceedings of the 5th international conference on Soft computing as transdisciplinary science and technology, Pages 226-231, 2008.

[18] Niladri Chatterjee, Shiwali Mohan, "Extraction-Based Single-Document Summarization Using Random Indexing" 19th IEEE International Conference on Tools with Artificial Intelligence, 2007.

[19] H. P. Luhn, “The Automatic Creation of Literature Abstracts,” IBM Journal of Research and Development, vol. 2, pp.159-165. 1958.

[20] Edmundson, H.P. : New methods in automatic extracting. In: Newspaper of ACM tea, 16-2 (1969) 264–85.

[21] Bram Beernink, Arjen Goedegebure, Niels Van Kaam, Remco van der Zon, "Information Supply during Military Missions. Relevant or Not?" Final Report Bsc Project, 2011.

[22] http://www-nlpir.nist.gov/related_projects/tipster_summac/cmp_lg.html