2006 IEEE International Conference on Fuzzy Systems
Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada
July 16-21, 2006

Automatic Text Summarization Using: Hybrid Fuzzy GA-GP

Arman Kiani -B
Kiani@kiaeee.org

M. R. Akbarzadeh –T [+]
Akbarzadeh@ieee.org

Department of Electrical Engineering
Ferdowsi, University of Mashhad, Mashhad, Iran

Abstract— **A novel technique is proposed for summarizing text using a combination of Genetic Algorithms (GA) and Genetic Programming (GP) to optimize rule sets and membership functions of fuzzy systems. The novelty of the proposed algorithm is that fuzzy system is optimized for extractive based text summarizing. In this method GP is used for structural part and GA for the string part (Membership functions). The goal is to develop an optimal intelligent system to extract important sentences in the texts by reducing the redundancy of data. The method is applied in 3 test documents and compared with the standard fuzzy systems as well as two other commercial summarizers: Microsoft word and Copernic Summarizer. Simulations demonstrate several significant improvements with the proposed approach.**

I. INTRODUCTION:

Fuzzy logic and evolutionary computation can be implemented synergistically for solving complex and high order problems. Fuzzy logic provides ability for human like conceptualization and reasoning, while evolutionary algorithms are useful for finding optimal solutions of nonlinear and complex optimization problems. One of these complex problems is text summarization. Text summarization is the technique by which a computer automatically creates a summary. The document of ISO 215 standards in 1986 formally defines a summary as a "brief restatement within the document (usually at the end) of its salient findings and conclusions" that "is intended to complete the orientation of a reader who has studied the preceding text." Automatic text summarization is the technique in which a computer automatically creates such a summary. This process is significantly different from that of human based text summarization since human can capture and relate deep meanings and themes of text documents while automation of such a skill is very difficult to implement. However, research into automatic text summarization has received considerable attention in the past few years due to the exponential growth in the quantity and complexity of information sources on the internet. Specifically, such text summarizer can be used to select the most relevant information from an abundance of text sources that result from a search by a search engine.

Automatic text summarization can be classified into two categories based on their approach:

1- Summarization based on abstraction, and
2- Summarization based on extraction.

Most of the works in this area are based on extraction. In contrast to abstraction method which heavily utilizes computation power for natural language processing (NLP) with the inclusion of grammars and lexicons for parsing and generation, extraction can be simply viewed as the process of selecting important excerpts (sentences, paragraph, etc.) from the original document and concatenating them into a more compact form. In other words, extraction is mainly concerned with judging the importance or the indicative power of each sentence in a given document. All sentences are first rated in terms of their importance, and then a summary is obtained by choosing a number of top scoring sentences.

Various approaches have been applied to the above problems, including statistical learning approaches. Kupiec et al [1] proposed the first known supervised learning algorithm. Their approach estimated the probability that a sentence should be included in a summary based on its feature values. Chuang and Yang [2] studied several algorithms for extracting sentence segments, such as decision trees and naïve Bayes classifiers. These methods perform well for summarizing documents in a specific domain. However, they require a very large amount of training sets to learn accurately. Mani [3] introduced structured features. In this method a rhetorical tree structure is built to represent rhetorical relations between sentence segments of the documents for nonstructural features. Features, here, are those important ideas which are obtained from the text and can be classified as nonstructural features (paragraph location, number of emphasize words, number of title words, etc.) and structural features (rhetorical relations

[+] Author is currently a Visiting Scholar at BISC, UC Berkeley. Author would like to acknowledge the generous support of the Islamic Development Bank under the 2005-2006 IDB Merit Scholarship Program.

0-7803-9489-5/06/$20.00/©2006 IEEE

between units such as causes, antithesis, conditions, contrast, etc.).

Neural networks [4] may present a suitable alternative solution paradigm due to their ability to discover nonlinear mappings well. But textual data bases are typically very large in size, and therefore this method may be very slow and in some cases may not even converge to the desired error. Another well-known method is tf-idf method [12], [13] which is acronym of term frequency–inverse document frequency. The term frequency in the given document gives a measure of the importance of the term within the particular document. The inverse document frequency is a measure of the general importance of the term (it is the log of the number of all documents divided by the number of documents containing the term) as in (1).

$$tf = \frac{n_i}{\sum_k n_k} \qquad (1)$$

With n_i being the number of occurrences of the considered term, and the denominator is the number of occurrences of all terms. Tf-idf is calculated as in (2).

$$tfidf = tf \times Log\left(\frac{|D|}{\left|\left(d_j \supset t_i\right)\right|}\right) \qquad (2)$$

Where: $|D|$ is the total number of document in the corpus and $\left|\left(d_j \supset t_i\right)\right|$ is number of documents where the term t_i appears (that is, $nj \neq 0$). A high weight in tf–idf is reached by a high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents; the weights hence tend to filter out common terms.

Tf–idf is a statistical technique used to evaluate how important a word is to a document. The importance increases proportionally to the number of times a word appears in the document but is offset by how common the word is in all of the documents in the collection or corpus. Tf–idf is often used by search engines to find the most relevant documents to a user's query. There are many different formulas used to calculate tf–idf [14]. The term frequency (TF) is the number of times the word appears in a document divided by the number of total words in the document. If a document contains 100 total words and, for instance, the word "cow" appears 3 times, then the term frequency of the word "cow" in the document is 0.03 (3/100). One way of calculating document frequency (DF) is to determine how many documents contain the word "cow" divided by the total number of documents in the collection. So if "cow" appears in 1,000 documents out of a total of 10,000,000 then the document frequency is 0.0001 (1000/10000000). The final tf-idf score is then calculated by dividing the term frequency by the document frequency. For our example, the tf-idf score for cow in the collection would be 300 (0.03/0.0001). Alternatives to this formula are to take the log of the document frequency.

The technique proposed here applies human expertise in the form of a set of fuzzy rules and a set of nonstructural features. Specifically, the parser is designed for selecting sentences based on their attributes and locations in the article using fuzzy logic inference system. The remarkable ability of fuzzy inference engines in making reasonable decisions in an environment of imprecision and uncertainty makes them particularly suitable [5] for applications involving risks, uncertainty, and ambiguity that require flexibility and tolerance to imprecise values. These features make them attractive for automatic text summarization. This proposed paper is organized as follows. Desired features and extraction of these features is explained in section 2. Sections 2.A, 2.B and 2.C describe the fuzzy logic system and evolutionary computing, respectively. Section 3 exhibits the simulation results and comparisons. And finally in Sections 4 and 5 are discussions and conclusions.

II. THE PROPOSED HYBRID GA-GP FUZZY ALGORITHM:

Initially, a parser is designed that extracts the desired features using Visual C++ 6.0. This program parses the text into its sentences and identifies the following nonstructural features for each sentence as the input of fuzzy inference system:

1-The number of title words in the sentence,
2-Whether it is the first sentence in the paragraph,
3-Whether it is the last sentence in the paragraph,
4-The number of words in the sentence,
5-The number of thematic words in the sentence, and
6-The number of emphasize words.

The main reasons for using the above features are explained in the papers by Brandow [6] and Baxendale [7]. Brandow et al. have shown that summaries consisting of leading sentences outperform most other methods in this field. Baxendale demonstrated that sentences located at the beginning and end of paragraphs are likely to be good summary sentences. Here, features 2 and 3 of the above list are extracted according to [7], assuming that paragraphs begin with an indentation in formal texts.

Feature 1 indicates the number of title words in a sentence relative to the maximum possible. This is determined by counting the number of matches between the content words in a sentence and the words in the title. This feature is expected to be important because the salience of a sentence according to ISO definition may be affected by the number of words in the sentence also appearing in the title.

Feature 4, length of a sentence, is useful for filtering out short sentences such as datelines and author names commonly found in news articles. We also anticipate that short sentences are unlikely to be included in summaries [7].

Numbers of thematic words indicate the words with maximum possible relativity. It is determined as follows: first we remove all prepositions and reduce the remaining words to their morphological roots [4]. The resultant content words in the document are counted for occurrence. The top 10 most frequent content words are considered as thematic words. This feature is important because terms that occur frequently in a document are probably related to its topic [4].

The last feature, emphasize words such as very, most, etc. because the important sentences can be signified with these kinds of words.

The selection of features plays an important role in determining the type of sentences that will be selected as a part of the summary and, therefore, would influence the performance of this fuzzy inference system.

A. FUZZY IMPLEMENTATION:

The features extracted in previous section are used as inputs to the fuzzy inference system. We partition these inputs to several fuzzy sets whose membership functions cover all the universe of discourse.

We use two kind of membership functions Bell membership functions which are specified by three parameters {a,b,c} as in (3). The parameters c and a, determine the center and width of the MF, and then use b to control the slopes at the crossover points.

$$Bell(x,a,b,c) = \frac{1}{1+\left|\frac{x-c}{a}\right|^{2b}} \qquad (3)$$

For instance, membership functions of the thematic word are shown in Figure 1.

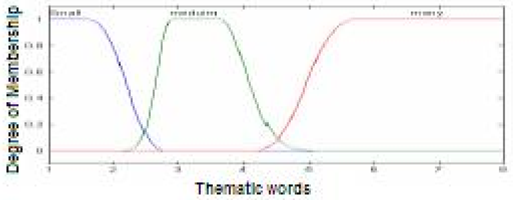


Figure 1:Membership Function of Thematic Words.

The output membership functions are shown in Figure 2, which is divided into 3 membership functions: Output {Unimportant, Average, and Important}. Four criteria for decision making are defined as follows: a) summaries consisting of leading sentences outperform most other methods in this domain b) sentences located at the beginning and end of paragraphs are likely to be good summary sentences. C) Summaries with maximum number of thematic words have important ideas. D) Number of summaries words should be minimized so we should collect sentences with the maximum fresh data (not repeated information) but with shortest length.

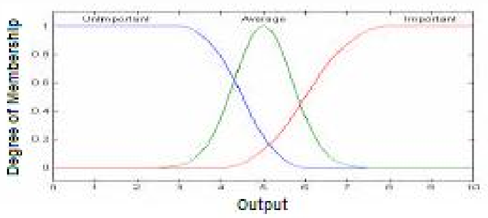


Figure 2: Membership function of Output

The most important part in this procedure is the definition of fuzzy IF_THEN rules. We determine these rules according to our criteria [4], [6] and [7]. We define 729 rules. For example our rules are demonstrated as follow.

IF (sentence-location is first) and (Number-of-title-words is very much) and (Sentence-length is not short) and (Number-of-thematic-words is many) THEN (Sentence is important)

Figure 3: Sample of IF-THEN Rules

Because we aim to implement this summarizer as the web base summarizer, we need to maximize its accuracy while optimize its running time. On the other hand, fuzzy inference system with 729 rules need much time to run and is not necessarily optimal. Therefore, a hybrid of GA and GP is used to find the optimal rule-based system.

B. OPTIMAZATION BY GA:

In previous section, we used a stochastic method for finding membership function. Now our optimization algorithm is implemented.

1) Initial Populations:

First membership function's parameters are specified. Every one of the six inputs has three membership functions and every membership function has 8 characters, so every individual has 48 parameters. The algorithm starts by initializing the two populations randomly. The GA and GP go through the genetic operators separately. In this method we optimize rule sets while membership functions are fixed or vice versa which we describe GP optimization in the next section.

2) Fitness function:

Proper fitness function is very important which it gives higher fitness to better set of membership function parameters and better rule sets.

In this paper we use a fitness function as a function of:

1. Maximization of thematic words in the summary θ_S per original article θ_O :

$$\frac{\theta_S}{\theta_O} = \text{(thematic words ratio)} \qquad (4)$$

2. Maximization of Emphasized words in the summary E_S per the original article E_O.

$$\frac{E_S}{E_O} = \text{(Emphasize words ratio)} \qquad (5)$$

3. Maximization of frequency of the title words in the summary T_S in compare with the original article T_O.

$$\frac{T_S}{T_O} = \text{(Title words ratio)} \qquad (6)$$

4. Minimization of Overlap between words in the summary sentences which summary should give us fresh information about the original article. It is calculated with (7) that is the ratio of the frequency of the same words in the summary $\Sigma(\Psi)$ per all words in the summary K_S.

$$\frac{\Sigma(\Psi)}{K_S} = \text{(Overlap ratio)} \qquad (7)$$

5. Minimization of the length of the summary as a ratio of the words in the summary K_S per words in the original text K_O.

$$\frac{K_S}{K_O} = \text{(length ratio)} \qquad (8)$$

So fitness of membership functions and rule set individuals are computed as follows:

$$f = \alpha \times \frac{\theta_S}{\theta_O} + \beta \times \frac{1}{\frac{K_S}{K_O}} + \gamma \times \frac{E_S}{E_O} + \eta \times \frac{T_S}{T_O} + \delta \times \frac{1}{\frac{\Sigma(\Psi)}{K_S}} \qquad (9)$$

Where α, β, γ, η and δ are weighted factors for balancing among the above parameters[4]. We use this fitness function for both GA and GP to get the optimal fuzzy inference system to make decision and select high ranked sentences.

Figure 5 illustrates the best, average and poorest fitness at each generation; it is shown the gradual improvement of the fitness function. New membership functions after applying GA are obtained. Figure 6 and 7 shows respectively original MF and the optimal MF for one individual membership function which is number of thematic words.

C. OPTIMAZATION BY GP:

Koza introduced the concept of GP [15], [16] in his creative research work. GP extends the chromosome of Genetic Algorithm (GA) into a combination of special programs to construct alternative solutions to special problems in which more emphasis is placed on the optimality of structure and relations among parameters. The individual in the population of GP is represented in the form of tree-like nodes or programs. Two kinds of nodes – functions and terminal - are used in the structure. The function nodes act as a program to fulfill a special task. Arithmetic operators, mathematical functions, Boolean operators, conditional operators are usually selected in the function sets of GP. User defined functions and automatically defined function also can be used in GP to solve special problems. The terminal nodes stand for the basis unit of the problem. Special constants, random numbers and input attribute are usually used in the terminal sets. The choice of function sets and terminal sets vary on the problem to be solved.

The population of GP evolves using the Darwinian principle of survival of the fittest. GP begins with a population of randomly created population using some kinds of tree-grown algorithms. They are possible solution to the desire problem. In every generation, the fitness of each individual is evaluated. For the next generation, the survival probability of an individual is based on its fitness. The population evolves over a number of generations through the application of variation operators, such as reproduction, crossover and mutation [15]. The dynamic tree-like structure of individual ensures the global search capability of GP to find proper structure and parameters of solution.

In this research we use GP to optimize the IF-THEN rules of fuzzy Inference system. As described earlier, the important part in using GP is to code the chromosomes.

1) Initial Populations:

Each individual has a chromosome like Figure 4. The MSB shows which membership function of the number of thematic words is selected in IF-THEN rules set. For exact description of this algorithm we show in the Figure 4.

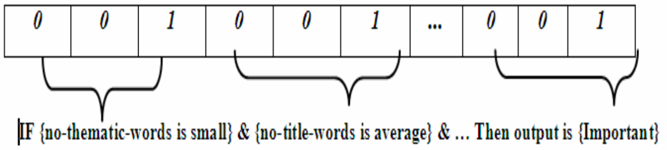

0
0
1
0
0
1
...
0
0
1
IF {no-thematic-words is small} & {no-title-words is average} & ... Then output is {Important}

Figure 4: Genotype Example for GP

2) Fitness function:

Rules obtain their fitness by their bondings with membership functions. Once a set of rules bonds with a set of membership functions, their joint fitness is evaluated as in Equation 9. Individual rule set fitness is then a function of how it has performed by bonding with different sets of membership function as explained in [17].

Figure 6 shows the best, average and poorest fitness at each generation but for GP population, i.e. population of rule sets. Both plots show that GA and GP converge to the optimal solution for our fitness function so now we have an optimal fuzzy inference system. Figure 9 shows our proposed algorithm in detail.

III. SIMULATION AND RESULATS

This section provides the simulation results of fuzzy-GA-GP decision making for text summarization. We used 3 news articles with various topics such as sports, music, world news IT news for training. The entire set consists of 1500 sentences.

Table 1: Comparisons of Precision (P), Recall (R), F1 and f Using Different Methods

Texts	MS Word Summarizer				Copernic Summarizer				Fuzzy Summarizer				Fuzzy GA GP Summarizer				Human summary			
	P	R	F1	f	P	R	F1	f	P	R	F1	f	P	R	F1	f	P	R	F1	f
T.1	0.250	0.300	0.273	0.345	0.561	0.700	0.622	0.562	0.683	0.724	0.700	0.793	0.796	0.631	0.703	0.963	0.943	0.894	0. 917	0.945
T.2	0.250	0.200	0.222	0.425	0.500	0.720	0.590	0.674	0.700	0.730	0.714	0.741	0.801	0.643	0.713	0.971	0.973	0.910	0.940	0.982
T.3	0.375	0.250	0.300	0.398	0.583	0.754	0.657	0.582	0.867	0.821	0.843	0.792	0.896	0.676	0.770	0.951	0.927	0.905	0.915	0.935
Average	0.291	0.250	0.265	0.389	0.548	0.724	0.623	0.606	0.75	0.758	0.752	0.775	0.831	0.650	0.728	0.961	0.947	0.903	0.924	0.954

After training, any news article can be fed into and the output will be shown which sentences are important in the summary.

Furthermore, Microsoft office 2000 and Copernic Summarizer are used as a benchmark to compare the algorithm with the existing commercial text summarizers. Table 1 shows a comparison of these three summarizers when applied to 3 different test documents.

Summarization is evaluated by three parameters precision (P), recall (R), F (Overall Fitness) and f (our objective function) as explained in [7]. These parameters are defined as follows. Let J be the number of title words in the summary, K be the number of selected sentences in the summary, and M be the number of title words in the original document. We refer to precision of the algorithm as the ratio of the number of title words in the summary and the number of selected sentences:

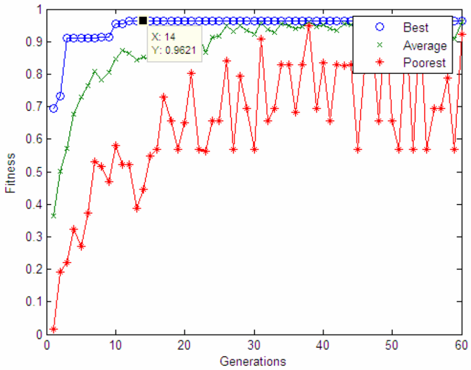


Figure 5: Fitness for GA Evolution.

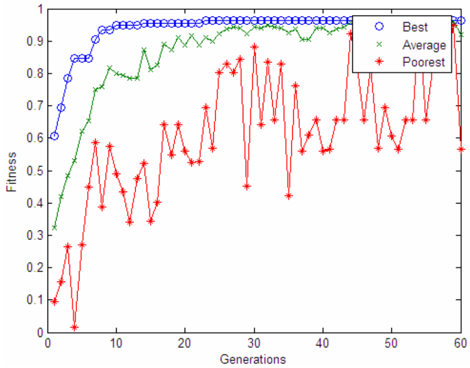


Figure 6: Fitness for GP Evolution. Maximum Fitness 0.9621

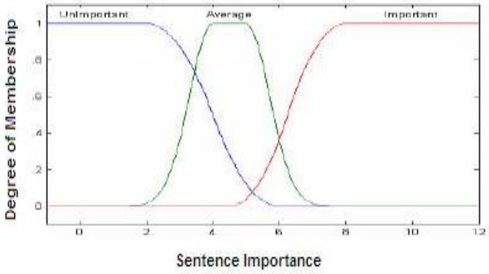


Figure 7: Output Membership Function after Applying GA.

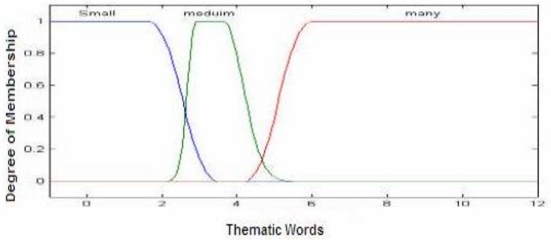


Figure 8: Thematic words Membership Function after Applying GA.

$$P = \frac{J}{K} \qquad (10)$$

Recall is defined as the ratio of the number of title words in the summary and the number of title words in the original document:

$$R = \frac{J}{M} \qquad (11)$$

Finally, F1, a combination of precision and recall can be calculated as follows:

$$F1 = \frac{2 \times P \times R}{P + R} \qquad (12)$$

However, the accuracy of this method ranged with an average accuracy of 94.3% when compared to the desired results obtained from the human reader. Our methods selected two sentences of test articles that were not selected as summary sentences by human reader.

In the Table 1, we compare fuzzy summarization and optimal fuzzy with the other summarizers. We can see that precision and also recall parameters are developed. For more comparison we evaluate the summary of this paper from as in the figure 10 up to part 2.B.

IV. DISCUSSION:

Microsoft word summarizer reaches an average precision 0.291, average recall of 0.25, and F1 of 0.26; while Copernic Summarizer reaches an average precision of 0.54, recall of 0.724, and F1 of 0.62. The fuzzy summarization achieves average precision of 0.75, recall of 0.758 and F1 of 0.752. The proposed evolutionary-optimized fuzzy algorithm achieves precision of 0.831 precision, recall of 0.653 and F1 of 0.728. We can easily understand that fuzzy logic that is optimized with evolutionary algorithms gives the best results because we optimized local and global property of the text and minimized overlap and maximize existence of the important sentences in the summary. We also can see average precision value of our algorithm is significantly improved using the combination of GA-GP with the defined fitness function.

V. CONCLUSION:

In this paper, a novel approach is proposed that extracts sentences based on an evolutionary fuzzy inference engine. The evolutionary algorithm uses GA and GP in concert. The genetic algorithm is used to optimize the membership functions and genetic programming is used to optimize the rule sets. The problem of competing conventions in fuzzy system optimization is thereby reduced by decoupling the two major categories of optimization in fuzzy systems. Fitness function is chosen to consider both local properties and global summary properties by considering various features of a given sentence such as its relative number of used thematic words as well its location in the whole document. Finally, the proposed method is compared with a expert-based fuzzy summarizer, Copernic and Microsoft Word in terms of the new fitness function as well as the more conventional scales of performance such as precision, recall and their combination. Simulations show significant improvement of the classical scales and our proposed scale.

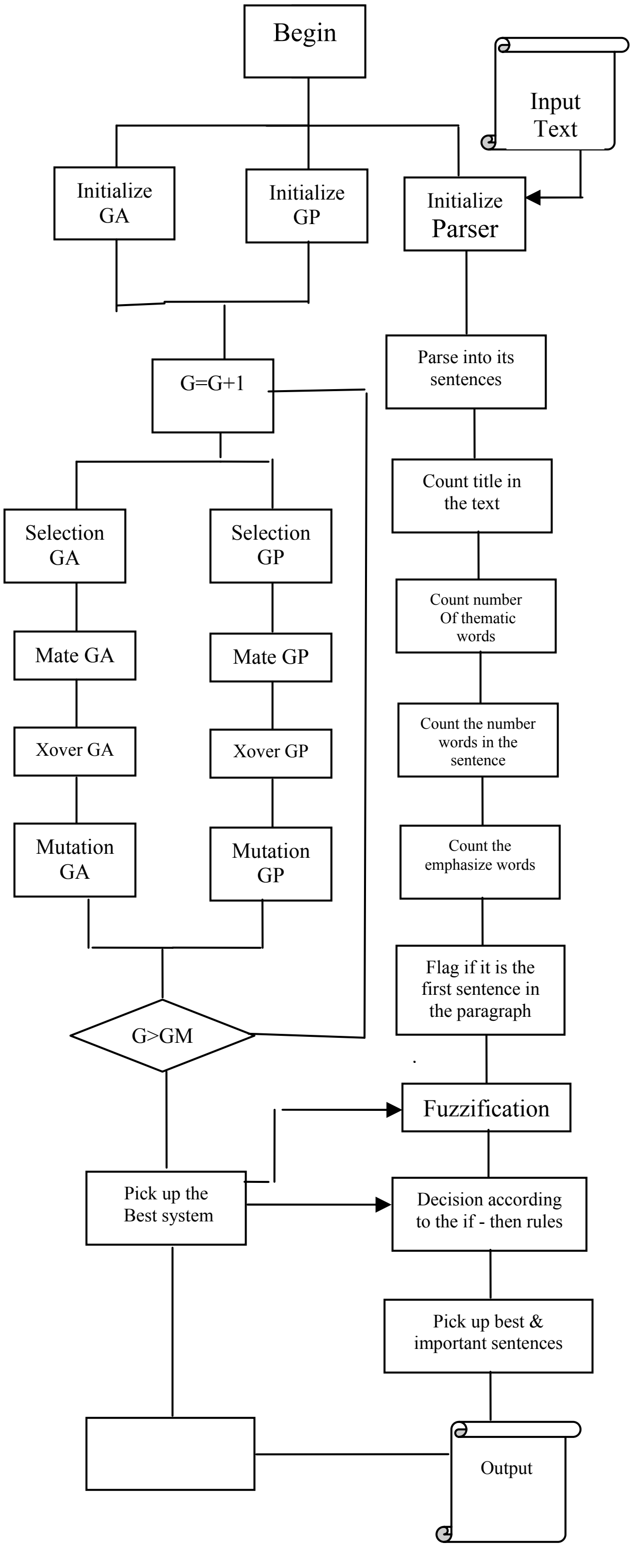


Figure 9: Our Proposed Algorithm Flowchart

VI. REFRENCES:

[1] Khosrow Kaikhah, "Automatic Text Summarization With NNs,", Second IEEE International Conference On Intelligent Systems, June 2004 PP 40-44.

[2] I.Mani, "Advanced in automatic summarization", John Benjamins Publishing company, PP.129-165, 2001

[3] W.T Chuang and J.Yang , "Extracting sentences segments for text summarization: a machine learning approaches"

Proceedings of the 23th Annual international ACM SIGIR conference on Research and Development in Information Retrieval, Athens, Greece, PP.125-159, 2000

[4] Canasai Kruengkrai and Chuleerat Jaruskulchai "Generic Text Summarization Using Local and Global Properties" Proceedings of the IEEE/WIC international Conference on Web Intelligence 2003

[5] R. Brandow, K. Mitlze and L. Rau, "Automatic condensation of electronic Publication by sentence selection", Information Processing and Management, Vol.31, No.5,PP 675-685, 1995.

[6] P.B. Baxendale, "Machine Made Index for Technical Literature: An experiment" IBM Journal of Research and Development, Vol. 2,No.4,PP.354-361,1958.

[7] C.Jaruskulchai and C.kruengkrai. A practical text summarization by paragraph extraction on information retrieval with Asian Languages, 2003.

[8] M.Porter, "An algorithm for suffix stripping", Program, Vol 14 no.3, PP 130-137, 1980.

[9] Canasai Kruengkrai and Chuleerate Jaruskulchai, "Generic Text Summarization Using Local and Global Properties of Sentences" International on web intelligent (WI'3) ,2003.

[10] E.K.Antonsson and H.-J. Sebastian , Fuzzy sets in engineering design. In practical application in fuzzy technologies, The Handbooks of Fuzzy set series Newyork 1999 57-117.

[11] Mizumoto, M." Improvement Methods of Fuzzy Control" In: Proceeding of the 3rd IFSA congress, Seattle, 1989,60-62.

[12] Salton, G. and McGill, M. J. 1983 Introduction to modern information retrieval. McGraw-Hill, ISBN 0070544840.

[13] Salton, G., Fox, E. A. and Wu, H. 1983 Extended Boolean information retrieval. *Commun. ACM* 26, 1022–1036.

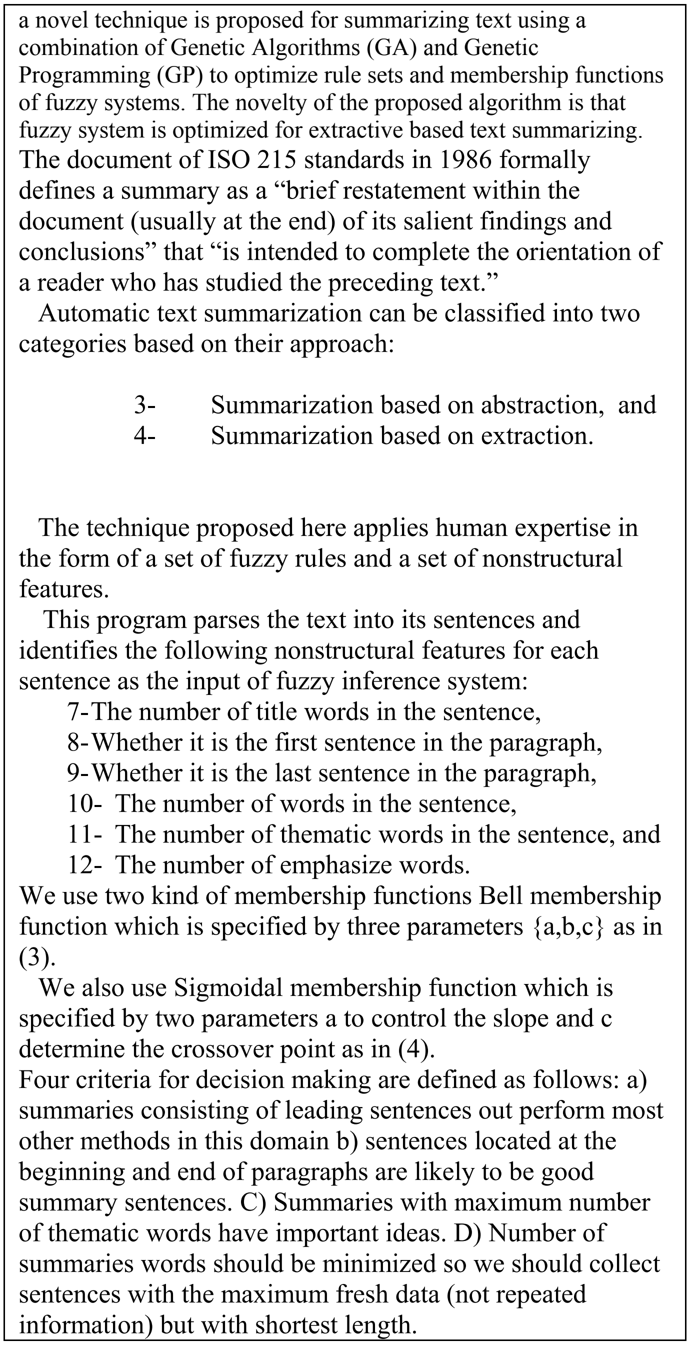
[14]Salton, G. and Buckley, C. 1988 Term-weighting approaches in automatic text retrieval. *Information Processing & Management* 24(5): 513–523.

[15] J.R. Koza, Genetic Programming: on the programming of computers by means of natural selection, Cambridge, MA: MIT Press, 1992.

[16] T. Loveard and V. Ciesielski, "Representing Classification Problems in Genetic Programming" in proceeding congress of Evolutionary Computation, PP.1070-1077, 2001.

[17] M.-R. Akbarzadeh-T., I. Mosavat, and S. Abbasi, "Friendship Modeling for Cooperative Co-Evolutionary Fuzzy Systems: A Hybrid GA-GP Algorithm," Proceedings of the 22nd International Conference of North American Fuzzy Information Processing Society, pp.61-66, Chicago, Illinois, 2003.

[18] S. Abbasi and M.R. Akbarzadeh-T., "Agent-based Cooperative Co-evolution for Fuzzy Systems," Proceedings of the 2004 World Automation Congress and Fifth International Symposium on Intelligent Automation and Control, Seville, Spain, June 28-July 1, 2004.



a novel technique is proposed for summarizing text using a combination of Genetic Algorithms (GA) and Genetic Programming (GP) to optimize rule sets and membership functions of fuzzy systems. The novelty of the proposed algorithm is that fuzzy system is optimized for extractive based text summarizing. The document of ISO 215 standards in 1986 formally defines a summary as a "brief restatement within the document (usually at the end) of its salient findings and conclusions" that "is intended to complete the orientation of a reader who has studied the preceding text."

Automatic text summarization can be classified into two categories based on their approach:

3- Summarization based on abstraction, and
4- Summarization based on extraction.

The technique proposed here applies human expertise in the form of a set of fuzzy rules and a set of nonstructural features.

This program parses the text into its sentences and identifies the following nonstructural features for each sentence as the input of fuzzy inference system:

7- The number of title words in the sentence,
8- Whether it is the first sentence in the paragraph,
9- Whether it is the last sentence in the paragraph,
10- The number of words in the sentence,
11- The number of thematic words in the sentence, and
12- The number of emphasize words.

We use two kind of membership functions Bell membership function which is specified by three parameters {a,b,c} as in (3).

We also use Sigmoidal membership function which is specified by two parameters a to control the slope and c determine the crossover point as in (4).

Four criteria for decision making are defined as follows: a) summaries consisting of leading sentences out perform most other methods in this domain b) sentences located at the beginning and end of paragraphs are likely to be good summary sentences. C) Summaries with maximum number of thematic words have important ideas. D) Number of summaries words should be minimized so we should collect sentences with the maximum fresh data (not repeated information) but with shortest length.

Figure 10: Summary of this paper part 1, 2.A