Automatic Document Summarization System Based on Natural Language Processing and Artificial Intelligent Techniques

M. I. Elalami[#1], A. E. Amin[*2], M. G. Doweidar[#3]

[#1] *Prof. of computer and Information systems, Mansoura University, Egypt*
[*2] *Ass. Prof. of computer and Information systems, Mansoura University, Egypt*
[#3] *Dept. of Computer Science Faculty of Specific Education, Mansoura University, Egypt*

Abstract — *Extract summary optimization is the process of creating a small version from the original text Satisfy user requirements. Extraction approach is one of way of extracting the most important sentences in document, this approach is used to select sentences after calculating the score for each sentence, and based on user defined summary ratio the top n sentences are selected as summary. The selection of the informative sentence is a challenge for extraction based automatic text summarization researchers. This research applied extraction based automatic single document text summarization method using the particle swarm optimization algorithm to find the best feature weight score to differentiate between important and non important feature. The Recall-Oriented Understanding for Gusting Evaluation (F-measure) toolkit was used for measuring performance. DUC 2007 data sets provided by the Document Understanding Conference 2007 were used in the evaluation process. The summary that generated by Particle Swarm Optimization algorithm was compared with other algorithms namely Latent Semantic Analysis, Gong&lui, and Vector Space Model, and used Particle Swarm Optimization algorithm as benchmark. Experimental results showed that the summaries produced by the Particle Swarm Optimization algorithm are better than another algorithm.*

Keywords — *Artificial Intelligent, Natural Language Processing, automatic text summarization techniques, particle swarm optimization.*

I. INTRODUCTION

The needs to extract important information from the texts which emerged as a result of the explosion of data available on the Web has become an urgent necessity. The process of extracting information from texts passed through multiple stages to reach accurate and specific extraction. In automatic text summarization systems that have been surveyed [1], there are some trends are observed, refers to this research area that depends on the ability to find efficient methods for automatic summarization.

Natural language processing (NLP) [2] in this field has contributed to a clear effort, bridging the gap between digital data and human skills. Text summarization is one from the important applications in NLP which is a solution that gives users an overview of all relevant literature data needed, and this helps the user in next decisions making. Signal document and multi-document are two main categories [3] of summarization categorization techniques, each one from these categories has characterized [4].

A. Summarization Of Single Document Methods

In single document summarization (SDS) methods, the used methods were classified according to the epoch. First methods used until before the sixth decade of the last century. In 1950s, SDS method is based on a particular word frequency in an article which provides a useful significance measure [5].This method is based on many steps that greatly influence the summarization process such as, stemmed words and deleted the stop words. Then a list of content words is compiled and stored by decreasing frequency, the index providing a significance measure of the word. Sequentially, the occurrences number of significant words within a sentence are derived by significance factor and the linear distance between them due to the intervention of non-significant words. The text summarization is formed by select a top-ranked sentences which produced from the ranking of all sentences in order of their significance factor.

In 1958s [6], the sentence position was used to find prominent parts of the documents. To verify the validity of this hypothesis, nearly 200 paragraphs

were examined. 85% of the paragraphs found that the topic sentence came as the first one and 7% were found in the last sentence. Thus, the naïve method which proved to be highly accurate was used to determine the topic sentence of the previous two choices.

In 1969 [7], a typical structure for extracting summarization was developed, which contributed to the creation of a manual summary protocol which was applied to 400 documents. This protocol contains four basic features, two features are previously used namely word frequency and sentence position. Whereas the others features were used for the first time are the words cue and skeleton of the document. Each feature has been manually attached to its own weight to give a score for each sentence. During the evaluation, it was found that about 44% of the manual summarization is matched the automatic summarization.

Second, the used methods until this time, which depend on the techniques of machine learning. The summarization was adopted in the early 1990s on statistical techniques [8] in the production of document extracts. Where most systems assumed the independence of features thus used the method of Bayes-naïve [9] that is able to learn from data.

DimSum system [10] is used naive-Bayes classifier but with richer features such as term frequency (tf) and inverse document frequency (itf). TF-IDF algorithm [11] is one of unsupervised algorithms. In this method, the weighting is according to term-frequency and inverse sentence-frequency. Sentence-frequency refers to the number of sentences including a term. In this algorithm, some parts of a sentence may be repeated in the other sentences. The advantage of this method is easy to compute. As the disadvantage of the method, it may be frequented some of the words are not so important, Cannot capture semantics which may cause a deviation in text summarization. While some systems focused on choosing the appropriate features and learning algorithms that support the assumptions of independence, these systems depend on rich features and decision trees [12]. The idea of these systems has studied the importance of a single feature, sentence position. The positioning method works by determining the sentence score where the sentences of greater topic centrality tend to occur in fixed locations (e.g. title, abstracts, etc.).

Other important methods to produce extractive summarization are include hidden Markov models (HMM) [13] and log-linear models (LLM) [14]. The basic motivation for using HMM is to account for local dependencies between sentences by using three features namely the sentence position in the document, terms number in the sentence, and the sentence terms likeliness given the document terms. The LLM follows the approaches of summarization have always assumed feature independence. This model is characterized by a better summary of the naive-Bayes model. While the use of neural networks and third-party features [15] such as common words in search engine queries to improve extractive summarization were among a very recent research in this field.

B. Summarization Of Multi-Document Methods

In the mid-1990s, there was an interest in summarizing the multi-documents in the field of news articles by several web-based news clustering systems [16]. The problem of multi-document summarizing is the multiplicity of information sources overlapping and complementary to each other so the main tasks to summarize the multi-documents is not only to recognize the repetition of sentences but the final summary must be coherent and complete.

At this early stage of the multi-documents summarizing, it was seen as a task that required substantial capabilities of both language interpretation and generation. There are a multi-document summarization techniques based on making use of similarity measures between pairs of sentences [17]. Approaches on how to use these similarities vary such as identify common themes through clustering and then select one sentence to represent each cluster [18], generate a composite sentence from each cluster [19] while some approaches work dynamically by including each candidate passage only if it is considered novel with respect to the previously included passages, via maximal marginal relevance [20] and some recent work extends multi-document summarization to multilingual environments [21].

The remainder of this paper is organized as follows: In section 2 introduce the related work Exhibits a general introduction, motivation, and objectives of the search, in section 3 progresses to the details of the research methodology, Section 4 briefly discusses experimental results, Finally, and Section 5 concludes the paper.

II. Related Work

Automatic summarization has two main different approaches namely extraction and abstraction. Extractive summarization methods (ESM) depend on the extraction of sentences from the original text by identifying important sections of the text and generating them verbatim. Whereas, abstractive summarization methods (ASM) based on interpreting and examine the text using advanced natural language techniques to generate a new shorter text that conveys the most critical information from the original text.

There are three fairly independent tasks for ESM [22]:

i. ***Intermediate Representation:*** To find salient content in any text, an intermediate representation (IR) of the text should be established. The

approaches of IR can be classified to topic representation and indicator representation. Techniques of EMS based on topic representation differ in their complexity terms and representation model which can be divided into:

- ***Frequency-Driven Approaches***: The two most common techniques are used to decide which words are more correlated to the topic namely Word Probability (WP) and Term Frequency-Inverse Document Frequency (TFIDF). The WP [23] is used frequency of words as indicators of importance is word probability by:

$$p(w) = \frac{f(w)}{N}$$

Where; $p(w)$ is the probability of a word (w).

$f(w)$ is the number of occurrences of the word. N is the number of all words in the input.The SumBasic system [24] is used the WP approach to determine sentence importance as:

$$g(S_j) = \frac{\sum_{w_i \in S_j} P(w_i)}{|\{w_i | w_i \in S_j\}|}$$

Where; $g(S_j)$ is the weight of sentence (S_j).

After that, the best scoring sentence that contains the highest probability word is selected, then the weight of each word in the chosen sentence is updated as:

$$p_{new}(w_i) = p_{old}(w_i) * p_{old}(w_i)$$

The selection steps will repeat until the desired length summary is reached. The TFIDF is considered one of the more advanced and very typical methods to give weight to words. Where very common words in the document are identified by weighting technique that giving low weights to words appearing in most documents as follows:

$$q(w) = f_d(w) * log \frac{|D|}{f_D(w)}$$

Where $f_d(w)$ is term frequency of word w in the document d, $f_D(w)$ is the number of documents that contain word w and $|D|$ is the number of documents in the collection D. There are another set of techniques based on TFIDF topic representation like Centroid-based summarization (CBS) [25]. Respectively, the CBS technique goes through several steps, the first clustering the document detection that describes the same topic together. To achieve this goal, creating the TFIDF vector representations of the documents then the TFIDF words scores that below a threshold are removed. After that, a clustering algorithm is run over the TFIDF vectors, consecutively adding documents to clusters and re-computing the centroids according to:

$$c_j = \frac{\sum_{d \in C_j} d}{|C_j|}$$

Where c_j is the centroid of the jth cluster, and C_j is the set of documents that belong to that cluster. The cluster is formed by pseudo-documents (centroids) that consist of the TFIDF scores of words whose are higher than the threshold. In the second step, sentences in each cluster are identified by using centroids that are central to the topic of the entire cluster. That is achieved by defined two metrics known as the cluster-based relative utility (CBRU) [26] and cross-sentence informational subsumption (CSIS) [27]. In order to approximate two metrics, three features (i.e. central value, positional value and first-sentence overlap) are used. Next, the final score of each sentence is computed and the selection of sentences is determined.

- ***Topic Word Approaches:*** One of the common topic representation approaches is the topic words technique (TWT). TWT aims to describe the topic of the input document by identifying words. Using frequency thresholds to locate the descriptive words in the document [28] is one the earliest works that leveraged. A more advanced version is used in documents summarization based on log-likelihood ratio test [29] to identify explanatory words. There are two sentence scoring functions [30] known as a function of the number of topic signatures it contains and the proportion of the topic signatures in the sentence. The first method may assign higher scores to longer sentences because they have more words. The second approach measures the density of the topic words.

- ***Latent Semantic Analysis (Lsa):*** LSA [31] is an unsupervised method for extracting the representation of textual semantics based on the observed words. This method is constructed from several steps beginning with the construction of a matrix (N_word * M_sentence) defined by the term - sentence matrix (T-SM) where N is the number of rows that correspond to the input words and M is the number of columns corresponds to the sentences contained in the document. Each element (a_{ij}) in T-SM represents the weight of the word (i), in the sentence (j), which computed by TFIDF technique. Then T-SM is converted to three matrices by using singular value decomposition (SVD) [32] as:

$$T - SM = U\Sigma V^T$$

Where: matrix U (n*m) represents a term-topic matrix having weights of words. Matrix Σ is a diagonal matrix (m*m) where each row corresponds to the weight of a topic (i). Matrix V^T describes a sentence represent a topic having weight of topic (i) in sentence (j).

- ***Bayesian topic models (BTM):*** BTM is probabilistic models that uncover and represent the topics of documents. The BTM is characterized by

quite powerful and appealing where it describes and represents topics in detail, enabling us to develop summary systems which can determine the similarities and differences between documents to be used in summarization [33]. BTM often utilize a distinct measure for scoring the sentence based on a measure of a difference between two probability distributions P and Q called Kullbak-Leibler (KL) [34] as:

$$D_{KL}(P|Q) = \sum_{w} P(w) log \frac{P(w)}{Q(w)}$$

Where: P(w) and Q(w) are probabilities of w in P and Q.

The Probabilistic topic models were used in several different fields [35], the most important of which was a Latent Dirichlet allocation (LDA) [36] that used an unsupervised technique to extract relevant information from multiple documents. The main idea of LDA is that documents representation was adopted in the form of a random mixture of latent topics. The model has been used to summarize multiple documents extensively in recent times. For example, BayeSum [37] is a summarization proposal based on Bayesian' technique with a focus on querying. While the topic model based on the Bayesian sentence [38] was used to summarization which used both term-document and term-sentence associations. Among the systems that have also performed an important performance, a two-stage hybrid model [39] uses the Bayesian model to summarize multi-document as a prediction problem. The first stage is a hierarchical model that detects the structure of the topic from all sentences. The second stage is a regression model that is trained according to the lexical and structural characteristics of the sentences. Then, the two stages are used to score sentences from new documents to form the summary.

Indicator representation techniques of EMS based on features set and use them to directly rank the sentences. The most widely used indicator representation approaches are:

- ***Graph-based methods***: The graph methods [40] are related to PageRank algorithm [41] where the documents are represented graphically by forming the vertices of the graph while the edges - the line between each vertices and the next - represent the similarity between the two sentences. The most common method of measuring the similarity between the two sentences is the cosine similarity with TFIDF weights for words.
- ***Machine learning techniques***: The approaches of machine learning models [42] are treated with summarization as a classification problem. In the initial attempts, the classification function known as the naive-Bayes classifier was developed to classify the sentences as summary sentences and non-summary sentences based on the features of these sentences. Documents are divided into a training set and their extractive summary, the classification probabilities are learned statistically from the training data using Bayes' rule:

$$P(s \in S|F_1, F_2, \dots, F_k) = \frac{(P(F_1, F_2, \dots, F_k)|s \in S)P(s \in S)}{P(F_1, F_2, \dots, F_k)}$$

Where: s is a sentence from the document collection, $F_1, F_2, \dots, F_k$ are features used in classification and S is the summary to be generated. Assuming the conditional independence between the features:

$$P(s \in S|F_1, F_2, \dots, F_k) = \frac{\prod_{i=1}^{k} P(F_i|s \in S)P(s \in S)}{\prod_{i=1}^{k} P(F_i)}$$

The sentence score is the sentence probability that belongs to the summary. The classifier select is playing the role of the sentence scoring function. There are some frequent features used in summarization [43] such as the position of sentences in the document, sentence length, presence of uppercase words, the similarity of the sentence to the document title, etc. Machine learning approaches have been widely used in summarization, such as decision trees [44], support vector machines [45], Hidden Markov models [46] and Conditional Random Fields [47].

i. ***Sentence Score***: an importance score for each sentence is assigned when IR is generated. In the topic representation, the score of a sentence represents how well the sentence explains some of the most important topics of the text. In the indicator representation, the score is computed by aggregating the evidence from different indicators.

ii. ***Summary Sentences Selection***: There are some approaches using to select the top (n) most important sentences to produce a summary of the text. Selecting the important sentences based on greedy algorithms is one from some approaches that use in EMS. The other some approaches are converting the selected sentences into an optimization problem [48]. The context and type of the document are other factors that should be taken into consideration while selecting the important sentences [49].

III - *RESEARCH METHODOLOGY*:

The proposed system allows the user to produce a summary of the documents based on four methods, three of which are considered the most common methods used in the summary. While as the fourth is a proposed method based on artificial intelligence techniques to optimize the features extracted from the summaries of the previous three methods and their importance to the user. As shown the figure 1, the proposed system is divided into three basic phases: the first phase is the preprocessing of the document, the second phase is the intermediate representation, and the third is extracting the

information sentences from the document. In the first phase, Document pre-processing is the process of incorporating a new document into an information retrieval system. That is achieved through four sub-processing are Sentence Segmentation that aim to segmented separately the document into nth sentences, Tokenization that tokenizing the distinct terms of each sentence, Stop Word Removal that meaning remove the less important significance used words with respect to document (such as 'a', 'and', 'the' …etc.) and Stemming refers to the process of reducing a word to its most basic form. Intermediate Representation (IR) of the text - the second phase - have to summarize and identify important content based on this representation. Sentence informative score is used as IR which calculated by the weight of the words in each sentence. TF.IDF is the common method used in IR which represents the figure 2.

Finally, information sentences extracting, which contribute to the extraction of information by using four different methods are:

I. Latent Semantic Analysis (LSA) – Method A - is an unsupervised approach based on an algebraic statistical technique that extracts the semantic structures of words and sentences. LSA uses the input document context to extract information that represents in common words in several different sentences, keeping in mind that the common words between the sentences indicate that the sentences are semantically related. Singular Value Decomposition (SVD) is used to find out the interrelations between sentences and words. SVD is characterized not only by the capability of modeling relationships among words and sentences but also by the noise reduction that helps to improve accuracy. The summarization algorithms that are based on LSA method usually contain three main steps as shown in figure 3.

II. Gong and Liu [50] – Method B- is algorithm follows the same LSA approach but differs in the way of sentences selection. The $V^T_{m\times m}$ matrix that derived from SVD values is used to selecting the important sentences. The dimension of $V^T_{m\times m}$ matrix is representing the relationship between the sentence and the concept. Where; row order indicates the importance of the concepts, such that the first row represents the most important concept extracted. A higher cell value indicates that the sentence is more related to the concept. In this method, the sentences are chosen according to the order of the importance of the concepts so that the first sentence is chosen from the most important concept. Then the second sentence is chosen from the second most important concept. Thus; until reaching to the predefined number of sentences determined by the user.

III. Vector Space Model (VSM) –Method C - is widely used to represent the documents through the words that they contain in a formal manner by the use of vectors in a multidimensional space. VSM is known as the Bag-of-Words, i.e. the word order is not important, that formed by reducing the document through simplification during preprocessing (lexicon). The concepts behind vector space modeling are that by placing terms, documents and queries in term-document space, where, each term is given a weight which measures its importance in the document. Weight has three calculation types namely Binary, Frequency and Corrective as shown in figure 2. Thus, a matrix of rows (sentences) and columns (terms) is generated. So that, the document is transformed into a set of vectors sentences. A lexicon of words produces a sentence-term matrix (S-TM) where each row contains the weighting of the word in the sentence. Computing the similarities between queries and the terms or documents and allow the results of the computation to be ranked according to the similarity measure between them are used to a great deal in automatic summarization.

IV. Particle Swarm Optimization (PSO) – Method D - is used to available for generating an effective summary that satisfies the optimization properties and improves the performance of text summarization. PSO is used to available for generating an effective summary that satisfies the optimization properties and improves the performance of text summarization. The idea of PSO algorithm is based on discovering the patterns that govern the flights of the birds and their sudden change of direction and the optimal shape of the group. The summarization of previous three methods is used as input to PSO method. The matrix S will be the sentences array of each summarization that is defined as $S = \{s_1, s_2, s_3\}$. There are five physical criteria applied to each sentence as shown in Table 1 then set a weight for each criterion by:

$$Score_{Weight}(S_i) = \sum_{i=1}^{5} W_i \times Score\_f_i(S_i)$$

Where $Score_{Weight}(S_i)$ is the score of sentence S, W_i is the weight of the feature i that produced by PSO, (i) is the number of feature and is a function that calculate the score of the feature (i). Sentences criteria in each summary are placed in the matrix called F that is used as $F_i = \{f_1, f_2, \dots f_M\}$. Because there are five quantitative indexes for each sentence, each F_i is known as $F_i = \{TF(S_i), SL(S_i), SP(S_i), ND(S_i), TW(S_i)\}$.

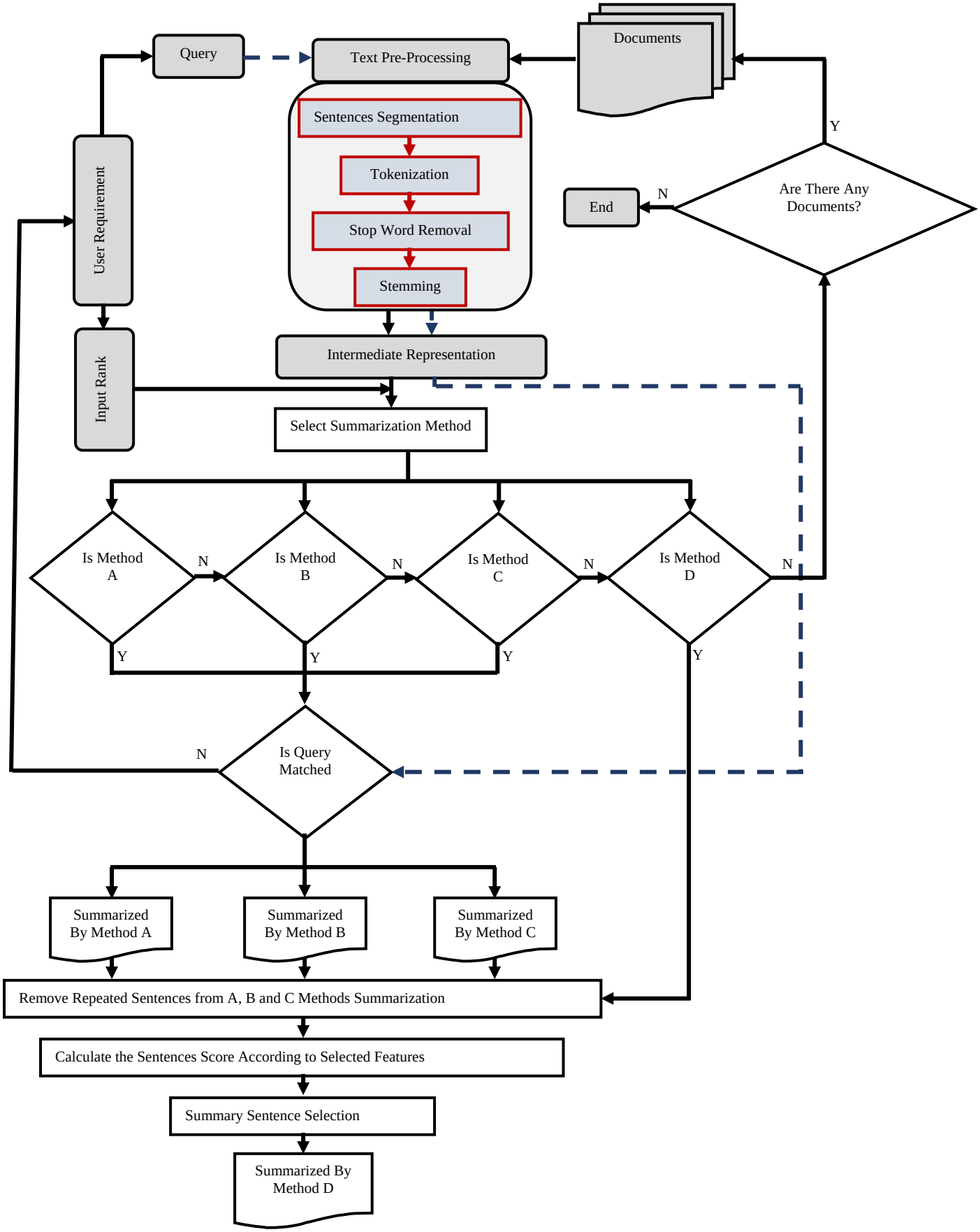

Fig 1: The Research Methodology Flowchart

TF-IDF Algorithm

For Pre-processed Words (Terms):

- *The collection frequency that used for reducing the relevance of the term is:*

$$tf_{t,d} = \sum_{x \in d} f_t(x)$$

$$where\ f_t(x) = \begin{cases} 1 & if\ x = t \\ 0 & Otherwise \end{cases}$$

- *Inverse document frequency of a term:*

$$idf_t = log \frac{N}{df_t} \quad where \quad N = collection\ Size$$

- *Calculate the tf-idf weighting:*
  - *The weight of a term is computed using tf and idf:*

$$w(t,d) = tf_{t,d} \times idf_t$$

$$where\ \ w(t,d) = \begin{cases} High: when\ term\ occurs\ many\ times\ in\ a\ small\ set\ of\ documents. \\ Low: when\ term\ occurs\ fewer\ times\ in\ documents\ or\ when\ t\ occurs\ in\ many\ documents. \\ Very\ low: when\ term\ occurs\ in\ almost\ every\ document. \end{cases}$$

Fig 2: TF-IDF Algorithm

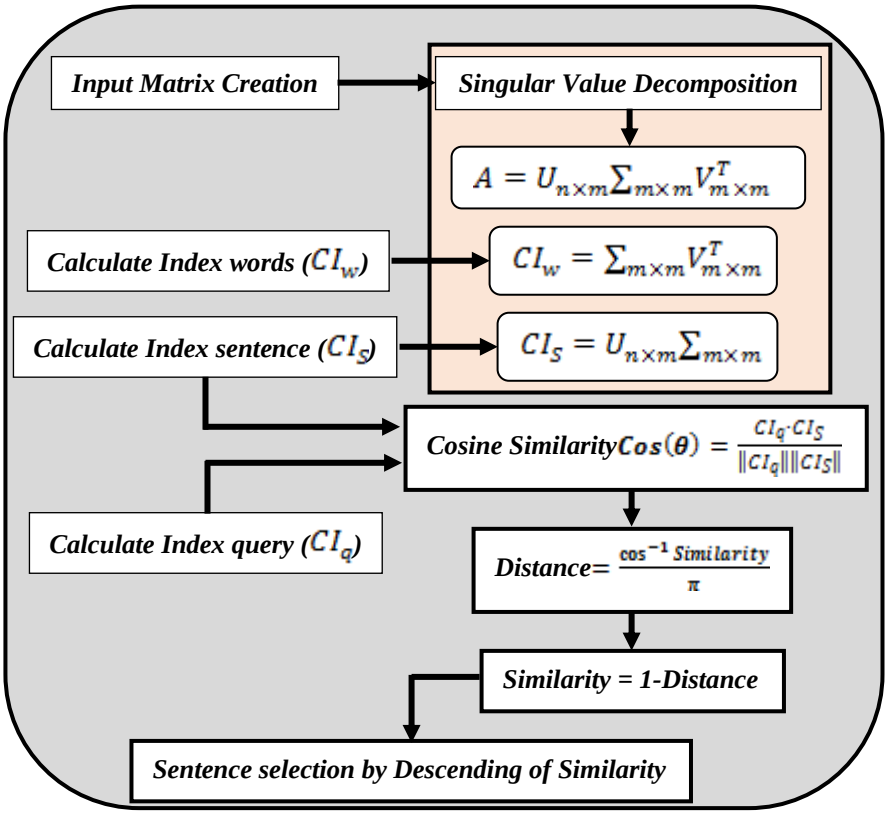

Fig. 3: LSA Diagram.

Table 1: The physical criteria

Criteria	Equation
Title Feature (TF)	$TF(S_i) = \frac{Count\ Word\ (S_i) \cap Count\ Word\ (Title)}{Count\ Length\ (Title)}$
Sentence Length (SL)	$SL(S_i) = \frac{Count\ Length\ (S_i)}{Count\ Length\ (S_j)}$
Sentence Position (SP)	$SP(S_i) = \frac{Count\ Word\ (d) - Current\ Position\ (S_i)}{Count\ Total\ (d)}$
Numerical Data (ND)	$ND(S_i) = \frac{Count\ ND(S_i)}{Count\ Length\ (S_i)}$
Thematic Word (TW)	The most frequency of top ten words are used as thematic.

Additionally, there are sixth criteria is represented in the repeated of the sentence in the previous summarization methods. This criterion is taken as a basic sentence in the PSO summary, then the maximum weight of physical criteria.

*IV-**EXPERIMENTAL RESULTS:***

Experimental results are divided into two main parts: implementation details and system evaluation. Implementation details describes the essential elements and components used in the proposed system in detail. The system is implemented by using Java. The system components will be described in detail through the following screens. The system starts with an opening screen showing the name of the program (Fig. 4.1) followed by a screen explaining the purpose of the program (Fig. 4.2).

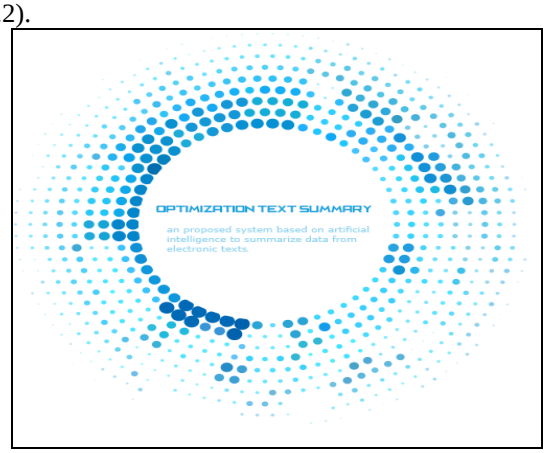

Fig 4.1: The Proposed System Opening Screen.

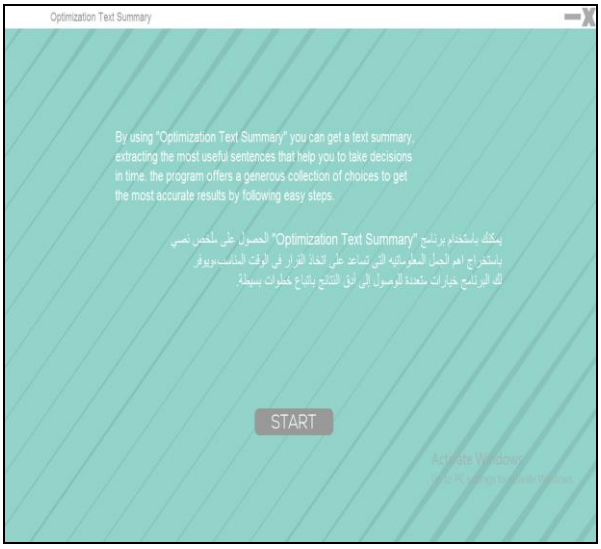

Fig 4.2: The Proposed System Description.

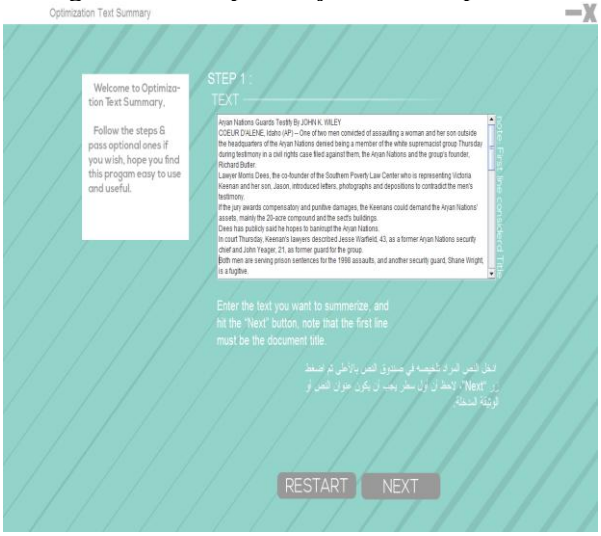

Fig 4.3: Input the Original Document.

Fig. 4.3 shows the entered document to be summarized using the copy and paste command then press NEXT key to select the number of sentences that display in summary. The number of sentences is chosen by entering the numeric value directly or by specifying a percentage of the document as shown in fig. 4.4.

Fig 4.4: Input Sentence Rank.

The document's pre-processing phase starts from the fig 4.5 where the main words of the document are separated without repetition, through which the user selects the words that are important to him (user query).

Fig 4.5: Select the Query from Document Analysis (Tokenization).

The system allows to users after choosing the important words two paths, first are choose the proposed method (Method D) to summarize after applying special features as shown in fig. 4.6.

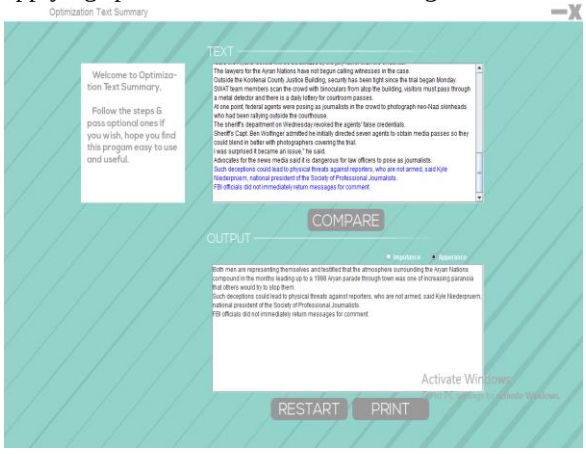

Fig 4.6: Summary by Method D.

The second is extract the summary from the summaries that extracted from the three commonly used methods. The user can determine the percentage of importance of each feature to achieve the proposed method and get optimization text summary as shown in fig. 4.7.

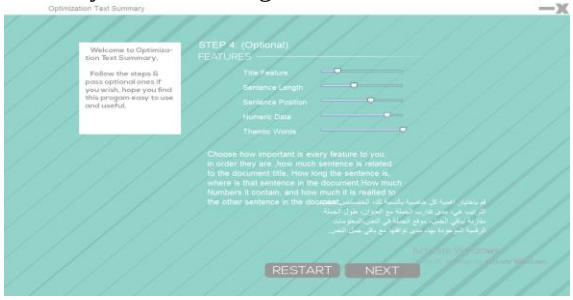

Fig 4.7: Determine the importance percentage for each feature.

At the OUTPUT window of Figures from 4.8 to 4.11, the summary results are shown according to the user's choice of the used summary method from the assigned window**.**

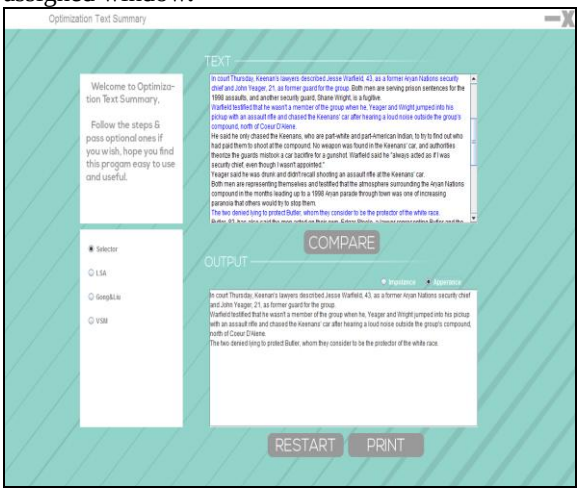

Fig 4.8: Select the Proposed Method.

4.9: Select the Latent Semantic Analysis Method.

Fig 4.10: Select the Gong & Lui Method.

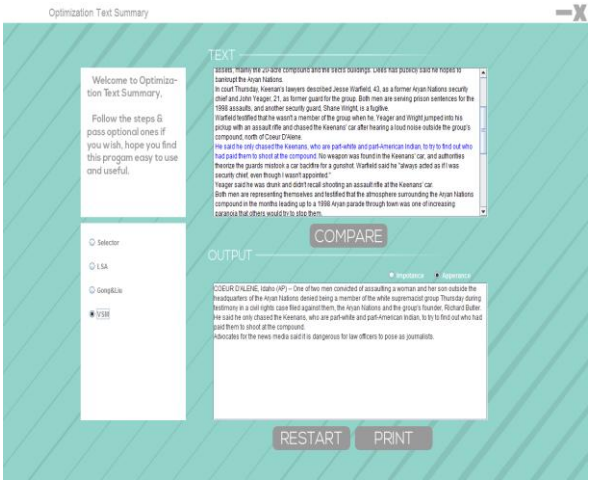

Fig 4.11: Select the VSM Method.

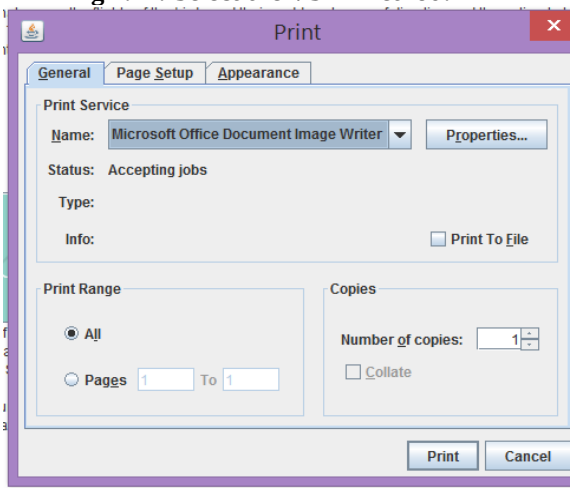

Fig 4.12 Print Summary.

The final summary of the program can be printed or copied in the word processing program installed on the computer as shown in fig. 4.12.

Evaluation of the system:

In extraction document summarization systems that depend on selecting the most important sentences in the source text into summary without change the original sentences. In such setting, the two most frequent and basic measures of information retrieval effectiveness are precision and recall. These are first defined for the simple case where an IR system returns a set of documents for a query. Precision (P) is the fraction of retrieved documents that are relevant:

$$Precision\ (P) = \frac{\#(Relevent\ items\ retrieved)}{\#(Retrieved\ items)} = P(Relevent|Retrieved)$$

Recall (R) is the fraction of relevant documents that are retrieved:

$$Recall\ (R) = \frac{\#(Relevent\ items\ retrieved)}{\#(Relevant\ items)} = P(Retrieved|Relevent)$$

A single measure that trades off precision versus recall is the F measure, which is the weighted harmonic mean of precision and recall:

$$F_{measure} = \frac{2PR}{P+R}$$

The standard summarization benchmark DUC2007 data sets are used for validating the proposed system. File name "APW20000831.0201 NEWS STORY 2000-08-31 23:59, Aryan Nations Guards Testify by JOHN K. WILEY" [51], number of the sentence is 25. Table 4.1 compares the four methods using Average recall, average precision, and average F-measure are calculated for each method.

Table 1: Comparison between Summarization Methods Performance.

Methods	Average Recall	Average Precision	Average F-measure
A	0.72	0.289077	0.383282
B	0.632	0.282686	0.33285
C	0.336	0.086681	0.136849
D	0.56	0.272109	0.309114

Figures 4.13, 4.14 visualize the details of results obtained. Based on the generalization of the obtained results, the performance of the proposed method is considered to be the highest, with a rate of 30.9% after both of methods Latent Semantic Analysis (LSA) and Gong & Lui method, which achieved a rate of 38.3% and 33.0% respectively. While Vector Space Model (VSM) method shows poor performance (13.0%) when compared with all method

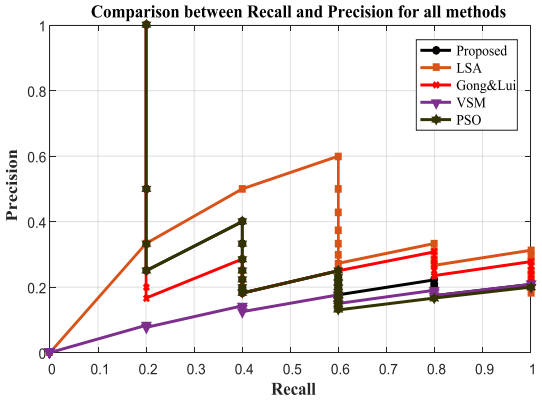

Fig 4.13: Precision and Recall for proposed system

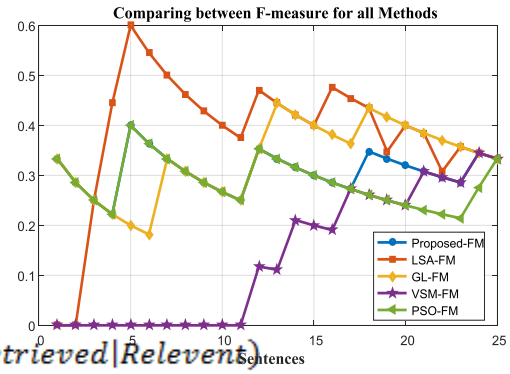

Fig 4.14: Comparison between F-measure for all methods.

V- *CONCLUSION*

In automatic text summarization, there are several techniques which used for selecting important sentences, this paper presents:

- The program receives the document and preprocessing it.
- The user enters the required query, the number of sentences extracted and the keywords.
- Apply PSO algorithm that five effective statistical features were selected (Title Feature, Sentence Length, Sentence Position, Numerical Data and Thematic Word), allows the user to determine the relative importance of each attribute then calculate the weight of each sentence in its input.
- The application results were used for algorithms LSA (Method A), Gung&Lui (Method B), VSM (Method C) were used as input selector (Method D).
- This paper used standard data set called DUC2007, and standard evaluation tools called f-measure.
- The generated summary compared with other algorithms (LSA, Gung&Lui and VSM). The summary that generated by PSO algorithm is better than another algorithms.

VI-REFERENCES

[1] Beel, Joeran, et al. "paper recommender systems: a literature survey." *International Journal on Digital Libraries* 17.4 (2016): 305-338.

[2] Zhu, Xiaojin. "Persistent Homology: An Introduction and a New Text Representation for Natural Language Processing." *IJCAI*. 2013.

[3] Cambria, Erik, et al. "New avenues in opinion mining and sentiment analysis." *IEEE Intelligent Systems* 28.2 (2013): 15-21.

[4] Di Fabbrizio, Giuseppe, Ahmet Aker, and Robert Gaizauskas. "Summarizing online reviews using aspect rating distributions and language modeling." *IEEE Intelligent Systems* 28.3 (2013): 28-37.

[5] Das, Dipanjan, and André FT Martins. "A survey on automatic text summarization." *Literature Survey for the Language and Statistics II course at CMU* 4 (2007): 192-195.

[6] Das, Dipanjan, and André FT Martins. "A survey on automatic text summarization." *Literature Survey for the Language and Statistics II course at CMU* 4 (2007): 192-195.

[7] Nenkova, Ani. "Automatic text summarization of newswire: Lessons learned from the document understanding conference." *AAAI*. Vol. 5. 2005.

[8] Yeh, Jen-Yuan, et al. "Text summarization using a trainable summarizer and latent semantic analysis." *Information processing & management* 41.1 (2005): 75-95.

[9] Agarwal, Shashank, and Hong Yu. "Automatically classifying sentences in full-text biomedical articles into Introduction, Methods, Results and Discussion." *Bioinformatics* 25.23 (2009): 3174-3180.

[10] Nenkova, Ani, and Kathleen McKeown. "Automatic summarization." Foundations and Trends® in Information Retrieval 5.2–3 (2011): 103-233.

[11] Ramos, Juan. "Using tf-idf to determine word relevance in document queries." Proceedings of the first instructional conference on machine learning. Vol. 242. 2003.

[12] Metzler, Donald, and Tapas Kanungo. "Machine learned sentence selection strategies for query-biased summarization." Sigir learning to rank workshop. 2008.

[13] Gupta, Vishal, and Gurpreet Singh Lehal. "A survey of text summarization extractive techniques." Journal of emerging technologies in web intelligence 2.3 (2010): 258-268.

[14] Li, Chen, Xian Qian, and Yang Liu. "Using Supervised Bigram-based ILP for Extractive Summarization." ACL (1). 2013.

[15] Cheng, Jianpeng, and Mirella Lapata. "Neural summarization by extracting sentences and words." arXiv preprint arXiv:1603.07252 (2016).

[16] Torres-Moreno, Juan-Manuel, ed. Automatic text summarization. John Wiley & Sons, 2014.

[17] Ferreira, Rafael, et al. "A multi-document summarization system based on statistics and linguistic treatment." Expert Systems with Applications 41.13 (2014): 5780-5787.

[18] Hellendoorn, Hans, and Dimiter Driankov, eds. Fuzzy model identification: selected approaches. Springer Science & Business Media, 2012.

[19] Krishnamoorthy, Niveda, et al. "Generating Natural-Language Video Descriptions Using Text-Mined Knowledge." AAAI. Vol. 1. 2013.

[20] Aggarwal, Charu C., and ChengXiang Zhai, eds. Mining text data. Springer Science & Business Media, 2012.

[21] Poibeau, Thierry, et al., eds. Multi-source, multilingual information extraction and summarization. Springer Science & Business Media, 2012.

[22] Ganesan, Kavita, ChengXiang Zhai, and Jiawei Han. "Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions." Proceedings of the 23rd international conference on computational linguistics. Association for Computational Linguistics, 2010.

[23] Turney, Peter D., and Patrick Pantel. "From frequency to meaning: Vector space models of semantics." Journal of artificial intelligence research 37 (2010): 141-188.

[24] Celikyilmaz, Asli, and Dilek Hakkani-Tur. "A hybrid hierarchical model for multi-document summarization." Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2010.

[25] Shen, Chao, and Tao Li. "Multi-document summarization via the minimum dominating set." Proceedings of the 23rd International Conference on Computational Linguistics. Association for Computational Linguistics, 2010.

[26] Wang, Dingding, Shenghuo Zhu, and Tao Li. "SumView: A Web-based engine for summarizing product reviews and customer opinions." Expert Systems with Applications 40.1 (2013): 27-33.

[27] Niazi, Muaz, and Amir Hussain. "Agent-based computing from multi-agent systems to agent-based models: a visual survey." Scientometrics 89.2 (2011): 479.

[28] Ramage, Daniel, Susan T. Dumais, and Daniel J. Liebling. "Characterizing microblogs with topic models." ICWSM 10 (2010): 1-1.

[29] Bates, Douglas, et al. "lme4: Linear mixed-effects models using Eigen and S4." R package version 1.7 (2014): 1-23.

[30] Lin, Hui, and Jeff Bilmes. "A class of submodular functions for document summarization." Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. Association for Computational Linguistics, 2011.

[31] Ozsoy, Makbule Gulcin, Ilyas Cicekli, and Ferda Nur Alpaslan. "Text summarization of turkish texts using latent semantic analysis." Proceedings of the 23rd international conference on computational linguistics. Association for Computational Linguistics, 2010.

[32] He, Zhanying, et al. "Document Summarization Based on Data Reconstruction." AAAI. 2012.

[33] Blei, David M. "Probabilistic topic models." Communications of the ACM 55.4 (2012): 77-84.

[34] Saggion, Horacio, et al. "Multilingual summarization evaluation without human models." Proceedings of the 23rd International Conference on Computational Linguistics: Posters. Association for Computational Linguistics, 2010.

[35] Gelman, Andrew, et al. Bayesian data analysis. Vol. 2. Boca Raton, FL: CRC press, 2014.

[36] Hoffman, Matthew, Francis R. Bach, and David M. Blei. "Online learning for latent dirichlet allocation." advances in neural information processing systems. 2010.

[37] Yang, Guangbing, et al. "A novel contextual topic model for multi-document summarization." Expert Systems with Applications 42.3 (2015): 1340-1352.

[38] Delort, Jean-Yves, and Enrique Alfonseca. "DualSum: a topic-model based approach for update summarization." Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, 2012.

[39] Pan, Li, et al. "A two-stage win–win multiattribute negotiation model: optimization and then concession." Computational Intelligence 29.4 (2013): 577-626.

[40] Lin, Hui, and Jeff Bilmes. "Multi-document summarization via budgeted maximization of submodular functions." Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, 2010.

[41] Mihalcea, Rada, and Paul Tarau. "A language independent algorithm for single and multiple document summarization." Proceedings of IJCNLP. Vol. 5. 2005.

[42] Hermann, Karl Moritz, et al. "Teaching machines to read and comprehend." Advances in Neural Information Processing Systems. 2015.

[43] Hu, Minqing, and Bing Liu. "Mining opinion features in customer reviews." AAAI. Vol. 4. No. 4. 2004.

[44] Joachims, Thorsten. Learning to classify text using support vector machines: Methods, theory and algorithms. Kluwer Academic Publishers, 2002.

[45] Hearst, Marti A., et al. "Support vector machines." IEEE Intelligent Systems and their applications 13.4 (1998): 18-28.

[46] Seymore, Kristie, Andrew McCallum, and Roni Rosenfeld. "Learning hidden Markov model structure for information extraction." AAAI-99 workshop on machine learning for information extraction. 1999.

[47] Shen, Dou, et al. "Document Summarization Using Conditional Random Fields." IJCAI. Vol. 7. 2007.

[48] Alguliev, Rasim M., Ramiz M. Aliguliyev, and Chingiz A. Mehdiyev. "Sentence selection for generic document summarization using an adaptive differential evolution algorithm." Swarm and Evolutionary Computation 1.4 (2011): 213-222.

[49] Le, Quoc, and Tomas Mikolov. "Distributed representations of sentences and documents". Proceedings of the 31st International Conference on Machine Learning (ICML-14). 2014.

[50] Wang, Dingding, et al. "Comparative document summarization via discriminative sentence selection." ACM Transactions on Knowledge Discovery from Data (TKDD) 6.3 (2012): 12.

[51] Lin, Hui, and Jeff Bilmes. "A class of submodular functions for document summarization." *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011.