Calculating the Upper Bounds for Multi-Document Summarization using Genetic Algorithms

Jonathan Rojas Simón, Yulia Ledeneva, René Arnulfo García-Hernández

Universidad Autónoma del Estado de México,
Unidad Académica Profesional Tianguistenco, Toluca, Estado de México,
Mexico

{ids_jonathan_rojas, renearnulfo}@hotmail.com, yledeneva@yahoo.com

Abstract. Over the last years, several Multi-Document Summarization (MDS) methods have been presented in Document Understanding Conference (DUC), workshops. Since DUC01, several methods have been presented in approximately 268 publications of the state-of-the-art, that have allowed the continuous improvement of MDS, however in most works the upper bounds were unknowns. Recently, some works have been focused to calculate the best sentence combinations of a set of documents and in previous works we have been calculated the significance for single-document summarization task in DUC01 and DUC02 datasets. However, for MDS task has not performed an analysis of significance to rank the best multi-document summarization methods. In this paper, we describe a Genetic Algorithm-based method for calculating the best sentence combinations of DUC01 and DUC02 datasets in MDS through a Meta-document representation. Moreover, we have calculated three heuristics mentioned in several works of state-of-the-art to rank the most recent MDS methods, through the calculus of upper bounds and lower bounds.

Keywords. Topline, multi-document summarization, genetic algorithms, upper bounds, significance.

1 Introduction

Extractive Text Summarization (ETS), is a task contemplated in Natural Language Processing (NLP), that allows to reduce the textual content of a document or a set of them, this reduction is performed through the selection of a set of most representative units (phrases or sentences), of original text obtained from a method or a computational tool, using supervised and unsupervised learning techniques [1, 2, 7, 15, 30].

Nowadays, the ETS task is one of the most worked in NLP. Since 1958, the first advances has been attributed to the works of Luhn [28], and Edmunson [10]. According to [41], these works has been considered as pioneers of Automatic Text Summarization (ATS), and particularly, ETS. Nevertheless, the most recent advances of ATS were presented through Document Understanding Conferences (DUC), workshops. Since 2001 to 2007, these workshops was organized by the National Institute of Standards and Technology (NIST), [9]. The main products of DUC workshops are the DUC datasets and are mainly used for two tasks: Single-Document Summarization (SDS), and Multi-Document Summarization (MDS), [37]. The first one consists in generate a selection of most important sentences from a single-document text, while the second task consist in generate a selection of the most important sentences of textual content of several documents [21].

In the last years, approximately 268 publications have been reported in the state-of-the-art using the DUC datasets [12]. In the most of these publications have been presented several methods for MDS task, using machine learning techniques through supervised and unsupervised methods [13, 15, 35], clustering-based methods for representing a set of clusters different relationships between sentences [14, 39, 55], algebraic reduction through Non-negative Matrix Factorization (NMF), [23, 24], and Latent Semantic Analysis (LSA), methods [18, 24, 51, 52], text representation with the use of graph-based algorithms [12, 33, 34], the use of optimization methods such as Genetic Algorithms (GA), [3, 17],

Memetic Algorithms (MA), [31, 32], Greedy Search (GS), and Dynamic Programming (DP), algorithms [29].

In previous works [26, 27, 41], has been mentioned that one of the main challenges of ETS is to generate automatic extractive summaries that similar to summaries generated by humans (gold-standard summaries). However, for several domains, the gold-standard summaries are made abstracting summaries by substituting some terms and phrases of the original text. For example, in the work of Verma and Lee [49], the gold-standard summaries of DUC01 and DUC02 employ approximately 9% of words not found in the original documents [37]. Consequently, the level of maximum similarity will be less than 100%, and even more, if compared from several gold-standard summaries, the upper bounds will be lower for any ETS method. Therefore, this problem involves the search of the best combinations of sentences of a set of documents that best similarity to gold standard summaries.

For SDS task, some heuristics have been used to compare several commercial tools and state-of-the-art methods with the purpose to comparing the performance of several ETS methods [16, 21, 22]. These heuristics are known as *Baseline-first*, *Baseline-random* [21], and in recent works, the use of *Topline* heuristic has been introduced [43]; in the most recent work [41], these heuristics have been used for calculating the significance of SDS task. However, for MDS has not performed a significant analysis for comparing the best novel state-of-the-art methods, because this task involves a mayor number of possible combinations to represent the best multi-document summary, and therefore for calculating the significant of several state-of-the-art methods requires some variants to the method presented in [41], for finding the best combinations of sentences.

The use of several optimization-based methods in ETS has represented a viable solution to generating extractive summaries of superior performance. These types of techniques include the use of GA, MA and GS methods [17, 29, 31, 32]. Therefore, the use of optimization-based algorithms, represents a viable solution to obtain extractive summaries closest to the human-written summaries. In this paper, a GA is used to obtain the combinations of sentences that best resemble selected by humans using the ROUGE-1.5.5 system and some variants to the method presented in [41], were applied. Furthermore, some meta-document principles were applied to calculating the *Topline* for MDS.

The rest of the paper is organized as follows: Section 2 present some related works that have used techniques based on exhaustive searches to determine the best combinations of extractive summaries and the calculus of significance for SDS. Section 3 describe the general process of GA. Section 4 describes the structure and development of the proposed GA for calculating the *Topline* for MDS using a meta-document representation. Section 5 shows the GA experimental configuration to determine the highest performance sentence combinations for calculating the *Topline* heuristic for DUC01 and DUC02 datasets. Moreover, we present a significant analysis to determine the best novel methods in the state-of-the-art with the use of three heuristics, such as *Baseline-first*, *Baseline-random* and *Topline*. Finally, Section 6 we describe the conclusions and future works.

2 Background and Related Works

Over of the last two decades with the existence of the DUC workshops, many advances have been made in the development of ETS. Several problems have been worked in the ETS, some of them involve the segmentation of sentences [19, 40], and automatic evaluation of summaries [20, 25, 45, 47]. However, to know and determine the best extractive summaries, few studies have been carried out, and some of them use techniques based on exhaustive searches to determine the best combination of sentences that best represent the summaries made by humans [41]. One of the first works was presented by Lin and Hovy [26], in 2003, where they developed a comprehensive search-based method to find the best sentence combinations of a document by taking the first 100±5 and 150±5 words of the DUC01 dataset for SDS task, and evaluating sentence combinations through co-occurrence of bag-of-words of the ROUGE system [25]. Nevertheless, the main drawback that affected the performance of this procedure was exponential increase of the search

space that implies the number of sentences of each document. For example, if we use a document of 100 sentences and furthermore inferred that on average each sentence has a length of 20 words, then to find the best extractive summary of 100 words should take the best 5 sentences of the 100 available (C_5^{100}), generating 75,287,520 possible combinations of sentences to find the best.

Seven years later, Ceylan [6], presented a similar exhaustive search-based method to obtain the best combinations of sentences in ETS. Unlike the work of Lin and Hovy [26], this method employs a probability density function (pdf), to reduce the number of all possible combinations using some metrics of ROUGE system, with the purpose to be applied from different domains (literary, scientific, journalistic and legal).

As we mentioned in [41], the main problem of this method involves the modification of ROUGE-1.5.5 Perl-based script to process several combinations of sentences in a cluster of computers to distribute the processing of the documents. Furthermore, in the news domain it was necessary to divide the original document in several sub-sections to reduce the processing of documents. The reduction of several combinations involves the discrimination of different possible combinations that can be generated.

In 2017, Wang [54], presented a nine-heuristics-based method to reduce the space of search that involves the combination of sentences for SDS and MDS tasks. This method is based to reduce the number of combination of sentences that present a low relation to gold-standard summaries from SDS and MDS. Subsequently, the remaining sentences are introduced through seven weighting methods to measure the similarity of the sentences in relation to gold-standard summaries. However, the use of several heuristics to determine the best combinations of sentences in different domains and different entries allows the increase of the computational cost to find the best sentence combinations. In addition, for SDS only a single gold-standard summary was used and in the case of MDS only 533 documents of 567 of the DUC02 dataset were used, generating more biased results.

Finally, in 2018 we presented a calculus of significance for SDS task [41]. Using three different heuristics (*Baseline-random*, *Baseline-first* and *Topline*) that represent the lower bounds and upper bounds for ETS, it has been calculated the level of significance of several SDS methods. However, this calculus only was performed for SDS. In this paper, we propose the method based on the use of GAs to find the best combinations of sentences that can be generated from the multi-document summaries of DUC01 and DUC02 datasets and rank the MDS methods.

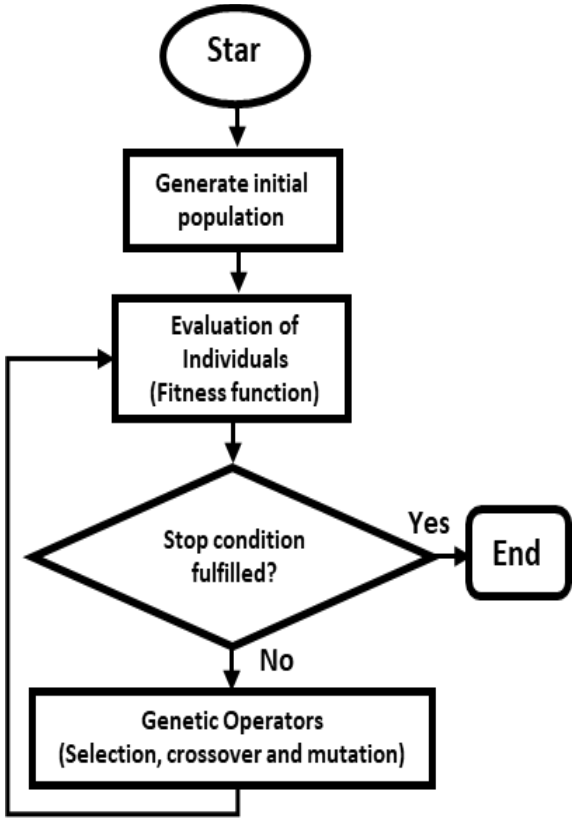

Fig. 1. Basic stages of GA [4-46]

3 Basic Genetic Algorithm

The GAs [22, 38, 42, 54], is a technique of optimization and iterative, parallel, stochastic search inspired by the principles of natural selection proposed by Darwin in 1859 [8]. The GAs was proposed by John Holland in 1975 as a method that pretends to simulate the actions of

nature in a computer to optimize a wide variety of processes [11]. Nowadays, the GA is the most widely used evolutive computing method in optimization problems [44].

A traditional GA is characterized by representing the solution of a problem in individuals, which are represented by variable bit strings and together form a population [4]. A GA begins with a population of N_{pop} individuals who share a set of n characteristics for each generation g, where each $i-th$ individual X_i is randomly generated as shown in Eq. (1):

$$X_i(g) = [X_{i,1}(g), X_{i,2}(g), \dots, X_{i,n}(g)], \quad i = 1,2, \dots, N_{pop}. \tag{1}$$

Each individual $X_r(g)$, is evaluated from a specific adaptation value (fitness function), to determine the quality of individuals and its proximity to the optimal values of GA [11, 38].

From the value obtained as a fitness function, a selection of individuals is performed, where each pair of parents $X_p(g)$ and $X_m(g)$, is chosen to participate in the cross-step forming individuals $Y_i(g)$, which have combined characteristics of $X_p(g)$, and $X_m(g)$. Finally, the new individual $Y_i(g)$, is introduced to the mutation stage, where partial and minimal modifications are made to generate an individual $Z_i(g)$. As mentioned in [31], the mutation of individuals is based on a probability P, as shown in Eq. (2):

$$Z_i(g) = \begin{cases} Mutate(Y_i(g)) & if\ rand < P, \\ Y_i & otherwise, \end{cases} \tag{2}$$

where the function $Mutate(Y_i(g))$, modifies the order of one or more sentences selected as target from a random value $rand$, included in a probability P. Otherwise, the individual $Y_i(g)$, is not modified. Finally, the population is updated according to the new individuals generated from the crossing and mutation stages of individuals. During the new generations, the average fitness function of each generation is improved because each generation produces individuals with better fitness function.

The selection, crossing, and mutation of individuals are iterated until they meet a certain termination criterion, these criteria are based on the number of iterations, the convergence of individuals of a gene, and on a fitness function [22]. In summary, the general process that conducts a GA is guided in Fig. 1, [4-46].

4 Proposed Method

In general, the proposed method is based on the steps and procedures of the basic GA described in Section 3. The proposed GA evaluates several combinations of sentences in an optimized search space, which are candidates in representing the best extractive summary of one or multiple documents. In this section, the proposed GA is presented.

4.1 Solution Representation

In [41], the solution is presented using a coding of individuals considering the order of sentences that can appear in extractive summary. Therefore, each individual X_i is represented in a vector of n positions $[P_1, P_2, \dots, P_n]$, where each position includes a sentence $\{S_1, S_2, \dots, S_n\}$, of the original document D, and the union of all the sentences will represent the content of the original document, as shown in Eq. (3):

$$\bigcup_{i=1}^{n} S_i = D. \tag{3}$$

For each coding to be considered as an extractive summary, the first sentences are considered from a set of words. For example, if we have a document with n=10 sentences and we generate an extractive summary of 100 words with an average of 20 words per sentence, then the position vector can use a sequence equivalent to [4, 1, 5, 6, 3, 2, 8, 7, 10, 9], indicating that the possible solution begins with sentences 4 and 1, ending with sentence 9, although only the first 5 sentences will be considered to comply with first 100 words as a summary.

Nevertheless, for MDS, the search space involves a mayor number of combinations of sentences due to the increase of sentences from a set of documents.

To represent the sentences of multi-documents we used the same genetic codification through the union of n sentences in m documents $\{S_{1,1}, S_{1,2}, \ldots, S_{n,m}\}$, to be considered as a meta-document that contains all the $i{-}th$ sentences of each $j{-}th$ document, where the union of all sentences represent a set of documents SD, as shown in Eq. (4):

$$\bigcup_{j=1}^{m} \bigcup_{i=1}^{n} S_{i,j} = SD. \tag{4}$$

For each coding to be considered as an extractive summary, the first sentences are selected until they comply a certain number of words as constraint. For example, if we have a set of documents SD with m = 5, where each one contains n = 5 sentences and we have an average of 20 words per sentence and as constraint they must be generated extractive summaries with 100 words, then the vector position can use a sequence equivalent to [4, 1, 5, …, 25], indicating that the possible solution begins with the sentences 4 and 1, ending with sentence 25 that corresponds to the last sentence of the last document, although only the first 5 sentences will be considered until to comply with first 100 words as a summary.

4.2 Fitness Function

The fitness function is an important stage for the performance of the GA and is the value by which the quality of the summaries is maximized with the passing of (g + 1), generations. To measure the quality of each summary, F-measure maximization based on the co-occurrence of bag-of-words and bigrams evaluated from ROUGE-1.5.5 system was used [25]. The maximum F-measure score of the individual $X_k(g)$, obtained from $X_i(g)$, population determine the best combination of sentences found in GA. This maximization is shown in Eq. (5):

$$Max\left(F\left(X_k(g)\right)\right) = \frac{\sum_{S \in S_{ref}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in S_{ref}} \sum_{gram_n \in S} Count(gram_n)}, \tag{5}$$

where n determine the size of n-gram for evaluating the sentence combinations of GA summary (ref), F is the F-measure score of ROUGE system and $Count_{match}(gram_n)$, is the number of n-grams that co-occurring between the GA summary and the set of gold-standard summaries. If the individual $X_k(g)$, have the greatest co-occurrence of n-grams from the all generations g of populations $X_i(g)$, then it will have the best combination of sentences when obtaining the largest number of retrieved n-grams.

4.3 Population Initialization

The most common strategy for initializing the population (when g = 0), must be generated with codifications of random real numbers for signature each sentence of the set $SD = \{S_{1,1}, S_{1,2}, \ldots, S_{n.m}\}$, in each position P_i of $[P_1, P_2, \ldots, P_{n \times m}]$. Therefore, the first generation of individuals will be according to Eq. 6:

$$X_c(0) = \left[X_{c,1}(0), X_{c,2}(0), \ldots, X_{c,n}(0)\right], X_{c,s} = a_s, \tag{6}$$

where a_s represents a real integer number $\{1, 2, \ldots, n \times m\}$, that corresponds to the number of selected sentence from the original set of documents SD, c = 1, 2, …, N_{pop}, s = 1,2, …, $n \times m$, n is the number of $n{-}th$ sentence of m document. Therefore, each sentence has the same probability of being included as part of an extractive summary respecting a number W of requested words as condition, as shown in Eq. (7):

$$\sum_{S_{i,j} \in Summary} l_{i,j} \leq W, \tag{7}$$

where $l_{i,j}$ is a length of the sentence $S_{i,j}$ (measured in words), and W is the maximum number of words allowed for generating an extractive summary. For each generation, Eq. (8), was used to generate a dynamic number of individuals depending on the number of sentences of SD:

$$N_{Pop} = NS_{Doc} \times M \times 5, \tag{8}$$

where NS_{Doc} is the number of sentences of each document and M is the number of documents in each set.

In this way, all the set of documents SD can generate different number of individuals because the number of sentences of each set involves different space of search.

4.4 Selection

The selection is the GA stage that allows to take a set of individuals X_c, from a generation *g* to obtain the greatest fitness values with the purpose of obtain best individuals in *g* + 1, generations. One of the operators of selection most known of GA is the elitism operator, which has the feature to choose a set of individuals of best aptitude in the generation *g* to pass to the generation $g + 1$.

According to [31], if we have $Pob(g) = \{X_1(g), X_2(g), \dots, X_{N_{pop}}(g)\}$, as a population of individuals ordered from greater to lesser fitness value, then the set of individuals that will be pass to the next generation will be $E(g+1) = \{X_1(g), X_2(g), \dots, X_e(g)\}$, where $E(g+1) \subseteq Pob(g)$, $e < 100\%$, and e is a parameter that specifies the percentage of individuals to be selected by elitism. However, for the selection of individuals it is required to use at least one selection operator to maintain N_{pop}, individuals for each generation.

To select the remaining individuals from each generation, we propose to generate new offspring from the tournament selection operator by taking several subsets of $N_{Tor,}$ randomly selected individuals to obtain the individual with the best fitness value, as shown in Eq. (9):

$$X_b(g) = Max\left(F(X_1(g)), F(X_2(g)), \dots, F\left(X_{N_{Tor}}(g)\right)\right), \quad (9)$$

where $X_b(g)$ is the individual with the best fitness value and F is the F-measure score of ROUGE-N metric. To integrate the selection stage, we propose to use the elitism operator to choose the best individuals of each generation g, using a percentage of individuals. Finally, the remaining individuals are obtained from the tournament selection operator using samples of 2 and 3 randomly obtained individuals.

4.5 Crossover

For the crossover of individuals, we use the cycle crossover operator (CX). The operator CX has the capacity to generate new offspring from the genetic coding of each pair of parents, considering their hereditary characteristics [11]. For the CX operator to be started, is necessary considering a crossover probability P to determine the subset of individuals who will perform the genetic exchange. Therefore, if b_{rand} is a random number between 0 and P, then the operator must select a starting point for genetic exchange of parents $X_{p1}(g)$, and $X_{p2}(g)$, which represent pairs of parents to cross, this starting point is randomly generated to generate a new individual $Y_i(g)$, as shown in Eq. (10):

$$Y_{i,s} = \begin{cases} X_{p1,s}(g) - & if\ s \le ptC \ \wedge\ 0 < b_{rand} \le P, \\ X_{p2,s}(g) & otherwise, \end{cases} \quad (10)$$

where $X_{p1,s}(g)$, represents the parent gene $X_{p1}(g)$, $X_{p2,s}(g)$, represents the parent gene $X_{p2}(g)$, and ptC is an integer value representing a start point selected randomly in a range of [1, n], where *n* is the size of the individual. To generate a second offspring, the roles of $X_{p1}(g)$ and $X_{p2}(g)$ are exchanged with the first parent being individual $X_{p2}(g)$.

4.6 Mutation

Remembering Eq. (2), of the Section 3, the mutation stage takes a set of individuals Y_i (g), to generate individuals Z_i (g), modifying some features for each generation g. We used the insertion mutation operator to select a pair of genes of the individual Y_(i,t) (g), and Y_(i,r) (g), randomly to insert the gene Y_(i,t) (g), in the gene Y_(i,r) (g), [4], as shown in Eq. (11):

$$Z_{i,s}(g) = \begin{cases} Y_{i,t}(g) = Y_{i,r}(g), Y_{i,t\pm1}(g) = Y_{i,t}(g), \dots, Y_{i,r}(g) = Y_{i,r\pm1}(g); \\ \qquad if\ 0 < rand \le P, \\ Y_{i,s}(g) \qquad otherwise. \end{cases} \quad (11)$$

where r is the variable that relates the gene to be inserted, the variable t represents the target gene to be inserted, which are a subset of numbers $s =$

Table 1. Datasets main characteristics

	DUC01	**DUC02**
Number of collections	30	59
Number of documents	309	567
Number of gold-standard summaries per collection/document	2	1-2
Multi-document gold-standard extractive/abstractive summaries	50, 100, 200, 400 abstracts	10, 50, 100, 200 abstracts 200, 400 extracts

Table 2. GA parameters to calculate *Topline* of DUC01 and DUC02 for MDS

Generations	**Selection**				**Crossover**		**Mutation**	
60	Elitism		Tournament		CX		Insertion	
	e	10%	N_{Tor}	2	P	85%	P	12%

$\{1, 2, \ldots, n\}$, and n identifies the sentence $S_{i,j}$ of each document. Therefore, if the random value *rand* is between the value 0 and P, then the mutation of individuals is performed by insertion operator, otherwise the individual is not modified.

4.7 Replacement of Individuals

For the replacement of individuals, we propose to integrate the set of individuals generated by elitist selection ($E(g+1)$) and the set of individuals $Z_i(g)$, from the mutation stage, to integrate the population of the next generation $X_i(g+1)$, as shown in Eq. (12):

$$X_i(g+1) = E(g+1) + Z_i(g). \quad (12)$$

4.8 Termination Criterion

The termination criterion used to halt GA iterations is determined by several generations established as an execution parameter.

5 Experiments and Results

In this section, we present the experiments performed to generate the best extractive summaries by the proposed GA, using DUC01 and DUC02 datasets. Moreover, the performance of some MDS methods and heuristics was presented through a calculus of significance for determine the best MDS methods in the state-of-the-art.

5.1 Datasets

Remembering some ideas from Section 1, the DUC datasets are the most common used for SDS and MDS task researches. In the state-of-the-art, approximately 89 publications in DUC01 and DUC02 has been reported. Due to this, we used DUC01 and DUC02 datasets to calculate the upper bounds for MDS. DUC01 and DUC02 are products of workshops organized by the National Institute of Standards and technology (NIST), for the development of ETS. The documents of these datasets are based on news articles from some agencies such as The Financial Times, The Wall Street Journal, Associated Press and others [36, 37].

DUC01 dataset consist of 309 English documents grouped into 30 collections, each collection contains an average of 10 documents based on news articles addressing natural disaster issues, biographical information, and others [36].

This dataset is divided for two tasks, the first task consists in generate summaries of single-documents with a length of 100 words and these

Table 3. Results of ROUGE-1 and ROUGE-2 methods and heuristics on DUC01 and DUC02 for summaries of 100 words (evaluated from abstractive gold-standard summaries)

	DUC01		DUC02	
Method	**ROUGE-1**	**ROUGE-2**	**ROUGE-1**	**ROUGE-2**
Topline	**47.256**	**18.994**	**49.570**	**18.998**
R2N2_ILP	36.910	7.870	37.960	8.880
R2N2_GA	35.880	7.640	36.840	8.520
Ur	34.280	6.660	34.160	7.660
Sr	34.060	6.650	34.230	7.810
Ur+Sr	33.980	6.540	35.130	8.020
LexRank	33.220	5.760	35.090	7.510
Baseline-first	**31.716**	**6.962**	**33.385**	**7.042**
Baseline-random	**26.994**	**3.277**	**28.637**	**3.798**

Table 4. Results of ROUGE-1 and ROUGE-2 methods and heuristics on DUC02 for summaries of 200 words (evaluated from extractive gold-standard summaries)

	DUC02	
Method	**ROUGE-1**	**ROUGE-2**
Topline	**75.163**	**66.512**
Baseline-first	**50.726**	**26.979**
Centroid	45.379	19.181
LexRank	47.963	22.949
NMF	44.587	16.280
FGB	48.507	24.103
BSTM	48.812	24.571
FS-NMF	49.300	24.900
WFS-NMF-1	49.900	25.800
WFS-NMF-2	49.100	25.200
Baseline-random	**38.742**	**9.528**

summaries were compared with two gold-standard summaries.

For MDS, consist in generate summaries of multiple newswire/newspaper documents (articles), on a single subject with 50, 100, 200, and 400 words. Moreover, for evaluation step, two abstracts were generated for each collection, generating 60 abstract summaries with the same lengths.

DUC02 dataset consist of 567 news articles in English grouped into 59 collections, each collection contains between 5 and 12 documents dealing with topics of technology, food, politics, finance, among others. Like DUC01, this dataset is mainly used for two tasks, the first is to generate summaries of a document, and each document has one or two gold-standard summaries with a minimum length of 100 words.

For MDS, consist in generate summaries of multiple documents, one and two abstracts were generated as gold-standard summaries for each collection, generating 118 abstracts/extracts with

Table 5. Results of ROUGE-1 and ROUGE-2 methods and heuristics on DUC02 for summaries of 50, 100 and 200 words (evaluated from abstractive gold-standard summaries)

	DUC02					
	50 words abstracts		**100 words abstracts**		**200 words abstracts**	
Method	**ROUGE-1**	**ROUGE-2**	**ROUGE-1**	**ROUGE-2**	**ROUGE-1**	**ROUGE-2**
Topline	**42.967**	**16.084**	**49.570**	**18.998**	**56.120**	**23.682**
ILP	28.100	5.800	34.600	7.200	41.500	10.300
Knapsack	27.900	5.900	34.800	7.300	41.200	10.000
Baseline-first	**26.939**	**5.241**	**33.385**	**7.042**	**41.118**	**10.362**
GS	26.800	5.100	33.500	6.900	40.100	9.500
Baseline-random	**21.599**	**2.298**	**28.637**	**3.798**	**36.074**	**6.308**

Table 6. Ranking of state-of-the-art methods and heuristics on DUC01 and DUC02 for summaries of 100 words

	DUC01		DUC02	
Method	**ROUGE-1**	**ROUGE-2**	**ROUGE-1**	**ROUGE-2**
Topline	**100%**	**100%**	**100%**	**100%**
R2N2_ILP	48.94%	29.22%	44.54%	33.43%
R2N2_GA	43.86%	27.76%	39.19%	31.07%
Ur	35.96%	21.52%	26.38%	25.41%
Sr	34.87%	21.46%	26.72%	26.39%
Ur+Sr	34.48%	20.76%	31.02%	27.78%
LexRank	30.73%	15.80%	30.83%	24.42%
Baseline-first	**23.30%**	**23.45%**	**22.68%**	**21.34%**
Baseline-random	**0%**	**0%**	**0%**	**0%**

lengths of 10, 50, 100 and 400 words [37]. Table 1 shows the general data for each dataset.

5.2 Parameters

To determine the upper bounds (*Topline*), of DUC01 and DUC02, different tests were carried out with some adjustments of parameters with the objective of obtaining the best extractive summaries. Table 2, shows the best tuning parameters applied to GA proposed to calculate the best extractive summaries of multiple documents.

The fitness value of each solution is obtained from the n-gram specification to be evaluated by the ROUGE system. In this paper, the unit of evaluation based on the co-occurrence of bag-of-words and bigrams (ROUGE-1 and ROUGE-2), was used, to compare the performance of the most state-of-the-art methods in relation to set of gold-standard summaries [25].

5.3 Comparison to State-of-the-Art Methods and Heuristics

As we have mentioned on Section 1, the importance of knowing the best multi-document extractive summaries consist in determining the *Topline* from the extractive summaries of several set of documents and calculating the significance of several state-of-the-art methods. In this section, we present a performance comparison of the state-

Table 7. Ranking of state-of-the-art methods and heuristics on DUC02 for summaries in 200 words

	DUC02	
Method	**ROUGE-1**	**ROUGE-2**
Topline	**100%**	**100%**
Baseline-first	**32.90%**	**30.62%**
Centroid	18.22%	16.94%
LexRank	25.32%	23.55%
NMF	16.05%	11.85%
FGB	26.81%	25.58%
BSTM	27.65%	26.40%
FS-NMF	28.99%	26.98%
WFS-NMF-1	30.64%	28.56%
WFS-NMF-2	28.44%	27.50%
Baseline-random	**0%**	**0%**

Table 8. Ranking of state-of-the-art methods and heuristics on DUC02 for summaries in 50, 100 and 200 words

	DUC02					
	50 words abstracts		**100 words abstracts**		**200 words abstracts**	
Method	**ROUGE-1**	**ROUGE-2**	**ROUGE-1**	**ROUGE-2**	**ROUGE-1**	**ROUGE-2**
Topline	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**
ILP	30.42%	25.40%	28.49%	22.38%	27.07%	22.98%
Knapsack	29.49%	26.13%	29.44%	23.04%	25.57%	21.25%
Baseline-first	**24.99%**	**21.35%**	**22.68%**	**21.34%**	**25.16%**	**23.33%**
GS	24.34%	20.33%	23.23%	20.41%	20.08%	18.37%
Baseline-random	**0%**	**0%**	**0%**	**0%**	**0%**	**0%**

of-the-art methods and their advances with respect to performance obtained from *Baseline-first*, *Baseline-random* and *Topline* heuristics. The methods and heuristics involved in this comparison are the following:

- **Baseline-first**: It is an heuristic that allows to use the first sentences of an original text according to a length of words to present as a summary to the user [16]. The performance of this heuristic generates good results in the ETS. However, this heuristic must be overcome by state-of-the-art methods [21]. To perform this heuristic in MDS, the summary is generated from the first sentences of each document until the determined number of words is met.
- **Baseline-random**: It is an heuristic in the state-of-the-art that selects random sentences to present them as an extractive summary to the user [21]. In addition, this heuristic allows us to determine how significant is the performance of ETS methods are in the state-of-the-art [22]. To perform this heuristic in MDS, we generate ten summaries for each set of documents with randomly selected sentences until the number of words is met.
- **Topline**: It is an heuristic that allows to obtain the maximum value that any state-of-the-art method can achieve due to the lack of

concordance between evaluators [43], since it selects sentences considering one or several gold-standard summaries. As mentioned in Section 2, efforts have been made in the state-of-the-art to know the scope of the ETS.

- **Ur, Sr, ILP:** In the work of [5], several machine regression models has been presented, the method *Ur* uses a bag-of-words regression with GS-based selection. The method *Sr* uses a sentence regression method with GS-based selection. Finally, the method Integer Linear Programming (*ILP*), is implement for MDS. These methods wezre considered as baseline methods.
- **R2N2_ILP and R2N2_GA:** In [5]**,** a method for ranking the sentences for MDS is proposed. Through a ranking framework upon recursive neural networks (R2N2), based on a hierarchical regression process the most important sentences of each document are selected.
- **ClusterCMRW and ClusterHITS:** The methods of [55], uses an Cluster-based Conditional Markov Random Walk Model (ClusterCMRW) and the Cluster-based HITS Model (ClusterHITS), to fully leverage the cluster-level information. Through these methods, relationships between sentences in a set of documents are associated.
- **LexRank:** It is a common stochastic graph-based method to generate extractive summaries through a centrality scoring of sentences. A similarity graph is constructed that provides a better view of important sentences from source text using a centroid approach [12].
- **Centroid:** In [39], a multi-document summarizer (MEAD) is presented. This method uses a centroid-based algorithm to score each sentence of each document through a linear combination of weights computed using the following features: Centroid based weight, sentence position and first sentence similarity.
- **GS, Knapsack and ILP algorithms:** In the work of [29] three inference global algorithms are proposed for MDS. Through the GS, Knapsack and ILP algorithms it was performed a study global of performance in MDS. The first is a greedy approximate method, the second a dynamic programming approach based on solutions to the Knapsack problem, and the third is an exact algorithm that uses an Integer Linear Programming formulation problem.
- **NMF:** The method of [52], uses an NMF to measure the relevance of document-terms and sentence-term matrices to ranks the sentences by their weighted scores.
- **FGB:** In [52], the clustering-summarization problem is translates into minimizing the Kullback-Leibler divergence between the given documents and model reconstructed terms for MDS.
- **BSTM:** The BSTM (Bayesian Sentence-based Topic Models), explicitly models the probability distributions of selecting sentences given topics and provides a principled way for the summarization task. BSTM is similar to the FGB summarization since they are all based on sentence-based topic model [53]. The difference is that the document-topic allocation matrix is marginalized out in BSTM.
- **FS-NMF:** The work of [50], considers a selection of theoretical and empirical features on a document-sentence matrix, and selects the sentences associated with the highest weights to form summaries.
- **WFS-NMF-1, WFS-NMF-2:** In [50], the NMF model is extended and provides a framework to select sentences with the highest weights to perform extractive summaries.

ClusterCMRW and ClusterHITS methods do not participate in the following comparisons, because in their evaluation stage was performed with a lower version of ROUGE system (ROUGE-1.4.2) and their results can differ of ROUGE-1.5.5 version.

For comparing and reweigh the performance of the methods previously described with the heuristics of the state-of-the-art, we used the evaluation based on the statistical co-occurrence of bag-of-words and bigrams (ROUGE-1 and ROUGE-2), of the ROUGE system [25], using the function of Eq. (13) to establish the performance of each state-of-the-art method respect to the best

extractive summaries obtained by the proposed GA:

$$ROUGE\text{–}N = \frac{\sum_{S \in Summ_{ref}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in Summ_{ref}} \sum_{gram_n \in S} Count(gram_n)}. \quad (13)$$

Table 3, 4 and 5, shows the average results of ROUGE-1 and ROUGE-2 when calculating the *Topline* for MDS of 30 document sets in DUC01 dataset and 59 document sets in DUC02 dataset using the limit of 50, 100 and 200 words as constraint of GA parameters presented in Table 2. The performance of the state-of-the-art methods are shown in this comparison.

According to the results presented in Tables 3, 4 and 5, *Topline* performance is substantially distant from other state-of-the-art methods, as mentioned by [43]. For DUC01 with 100 words, *Topline* obtained a performance equivalent to 47.256 with ROUGE-1 and 18.994 with ROUGE-2, while the best state-of-the-art method is *R2N2_ILP* obtaining 7.870 with ROUGE-2. For DUC02 with 100 words, *Topline* obtained a performance equivalent to 49.570 with ROUGE-1 and 18.998 with ROUGE-2, in the same way, *R2N2_ILP* is the best state-of-the-art method obtaining 37.960 with ROUGE-1 8.880 with ROUGE- 2 (see Table 3).

For DUC02 with 200 words, *Topline* obtained a performance equivalent to 75.163 with ROUGE-1 and 66.512 with ROUGE-2, while the best state-of-the-art method is *WFS-NMF-1* obtaining 49.900 with ROUGE-1 and 25.800 with ROUGE-2. Moreover, the heuristic *Baseline-first* outperforms all state-of-the-art methods (see Table 4).

For DUC02, *Topline* obtained a performance equivalent to 42.967 with ROUGE-1 and 16.084 with ROUGE-2 for summaries in 50 words. For summaries in 100 words, *Topline* obtained a performance equivalent to 49.570 with ROUGE-1 and 18.998 with ROUGE-2. For summaries in 200 words, *Topline* obtained a performance equivalent to 56.120 with ROUGE-1 and 23.682 with ROUGE-2. The best state-of-the-art methods are the methods *ILP* obtaining 28.100 with ROUGE-1 in 50 words, 41.500 with ROUGE-1 and 10.300 with ROUGE-2 in 200 words. The method based of in the Knapsack problem obtained 5.900 with ROUGE-2 in 50 words, 34.800 with ROUGE-1 and 7.300 with ROUGE-2 for summaries in 100 words. Furthermore, the *Baseline-first* heuristic outperform to the GS-based method in several scores (see Table 5).

A comparison of the level of advance of the most recent state-of-the-art methods is shown in Tables 6, 7 and 8. To determine this performance, we use the Eq. (14) based on the premise that the performance of *Topline* heuristic is 100% and *Baseline-random* is 0%.

$$ROUGE\text{–}N = \frac{(ROUGE\text{–}N_{OM} - ROUGE\text{–}N_{BR}) \times 100}{ROUGE\text{–}N_{TL} - ROUGE\text{–}N_{BR}}, \quad (14)$$

where $ROUGE\text{–}N$ specifies the F-measure score of bag-of-words and bigrams, OM is the performance of other methods, TL is the performance of *Topline* heuristic and BR is the performance of *Baseline-random* heuristic.

The best state-of-the-art method from the Table 6 presents an advance equivalent to 48.94% for ROUGE-1 and 29.22% for ROUGE-2 in DUC01, and DUC02 presents an advance equivalent to 44.54% for ROUGE-1 and 33.43% for ROUGE-2 for summaries of 100 words. Therefore, it follows that for the development of the MDS task there is 51.06% for ROUGE-1 and 70.78% for ROUGE-2 in DUC01, and 55.46% for ROUGE-1 and 66.57% for ROUGE-2 in DUC02 to be explored in summaries of 100 words. In the other hand, it is observed that the performance of *Baseline-first* heuristic is overcome by all state-of-the-art methods (see Table 6).

The best state-of-the-art methods present an advance equivalent to 30.64% for ROUGE-1 and 28.56% for ROUGE-2 (see Table 7). Therefore, it follows that for the development of the MDS task in summaries of 200 words, there is a 69.36% for ROUGE-1 and 71.44% for ROUGE-2 to be explored. In the other hand, the performance of *Baseline-first* heuristic is outperforming to best state-of-the-art method with 32.90% for ROUGE-1 and 30.62% for ROUGE-2.

For summaries of 50, 100 and 200 words, the best state-of-the-art methods were ILP-based method with a percentage equivalent to 30.42% for ROUGE-1 (50 words), 27.07% for ROUGE-1 and 22.98% for ROUGE-2 (200 words), while the

Knapsack problem-based method obtained a percentage equivalent to 26.13% for ROUGE-1 (50 words), 29.44% for ROUGE-1 and 23.04% for ROUGE-2 (100 words), (see Table 8). In general, the best state-of-the art methods presents an average percent of advance equivalent to 28.97% for ROUGE-1 and 24.05% for ROUGE-2. Therefore, it follows that for the development of the MDS task in summaries of 50, 100 and 200 words in DUC02, there is an average 71.03% for ROUGE-1 and 75.95% for ROUGE-2 to be explored. In the other hand, the performance of GS-based method is closer than Baseline-first in several ROUGE metrics.

6 Conclusions and Future Works

In previous works, the upper bounds for SDS and MDS has been calculated on exhaustive search-based methods to obtain the best extractive summaries. However, determine the best extractive summaries through this method was inadequate and expensive due to increase of documents and sentences. In this paper, we propose the use of GAs for calculating the upper bounds (*Topline* heuristic), to reweigh the performance of MDS methods.

Some GA operators were used to obtain the best extractive summaries. In the fit-ness function stage, it was proposed to use ROUGE-N method of ROUGE-1.5.5 system to evaluate the quality of GA combinations. Through ROUGE-N, we obtained several patterns features from gold-standard summaries.

In the state-of-the-art, the maximum possible performance value of MDS in DUC01 and DUC02 were unknown. However, it was possible to approximate the performance of the best extractive summaries with the use of GAs, to know the scope of MDS methods. In the other hand, we propose identifying several patterns of sentence features obtained from the best sentence combinations through supervised and unsupervised machine learning models to improve the performance of MDS methods.

In general, the best state-of-the-art methods (reported in Table 6, 7 and 8), are *R2N2_ILP*, *R2N2_GA*, *WFS-NMF-1*, *ILP* and *Knapsack* in different metrics. However, it was not possible perform a ranking of all state-of-the-art methods because several methods were not implemented in different subsets of documents of DUC01 and DUC02 datasets. In the other hand, the performance of *Baseline-first* is overcome in several subsets of documents (see Table 6 and 8), except for summaries in 200 words (from DUC02). With the new reweight of MDS methods (reported in Table 6, 7 and 8), it was possible to determine the advance percentages of the best state-of-the-art methods. In several subsets of documents (see Table 6, 7 and 8), it is observed that the percentage of significance is much closer to several methods of the state-of-the-art, so it will be very important to analyze the quality of the summaries generated by means of a Turing test, to demonstrate if the level of achieved performance of extractive summaries is confounded with summaries created by humans. Finally, we propose the use of GA-based method for calculating the upper bounds in several languages for determining the ranking of significance for several multilingual ETS methods.

References

1. **Acero, I., Alcojor, M., Díaz, A., Gómez, J.M., & Maña, M. (2001).** Generación automática de resúmenes personalizados. *Procesamiento del Lenguage Natural*, Vol. 27, No. 33, pp. 281–290.
2. **Alfonseca, E. & Rodríguez, P. (2003).** Generating extracts with genetic algorithms. *Advances in Information Retrieval ECIR*, Springer Heidelberg, Vol. 2633, pp. 511–519. DOI: 10.1007/3-540-36618-0_37.
3. **Alguliev, R.M., Aliguliyev, R.M., & Isazade, N.R. (2013).** Multiple documents summarization based on evolutionary optimization algorithm. *Expert Systems with Applications*, Vol. 40, No. 5, pp. 1675–1689. DOI:10.1016/j.eswa.2012.09.014.
4. **Araujo, L. & Cervigón, C. (2009).** Algoritmos Evolutivos: Un Enfoque Práctico, 2nd edn. *RA-MA*, Spain.
5. **Cao, Z., Wei, F., Dong, L., Li, S., & Zhou, M. (2015).** Ranking with recursive neural networks and its application to multi-document summarization. *Proceedings Twenty-Ninth AAAI Conference Artificial Intelligence*, pp. 2153–2159.
6. **Ceylan, H., Mihalcea, R., Öyertem, U., Lloret, E., & Palomar, M. (2010).** Quantifying the Limits and Success of Extractive Summarization Systems

Across Domains. *Human language technologies: The 2010 annual conference of the North American Chapter of the ACL (NACLO 2010)*, pp. 903–911.

7. **Chettri, R. & Chakraborty, U.K. (2017).** Automatic Text Summarization. *International Journal of Computer Applications*, Vol. 161, No. 1, pp. 5–7.
8. **Darwin, C. (1859).** The origin of species by means of natural selection: or, the preservation of favored races in the struggle for life. *AL Burt*, pp. 441-764.
9. **DUC (2002).** *Document Understanding Conferences.* http://www-nlpir.nist.gov/projects/duc.
10. **Edmundson, H.P. (1969).** New methods in automatic extracting. *Journal Association for Computing Machinery (JACM)*, Vol. 16, No. 2, pp. 264–285. DOI:10.1145/321510.321519.
11. **Eiben, A.E. & Smith, J.E. (2015).** *Introduction to Evolutionary Computing.* Vol. 12, No. 1995, Springer-Verlag Berlin Heidelberg.
12. **Erkan, G. & Radev, D.R. (2004).** LexRank: Graph-based lexical centrality as salience in text summarization. *Journal Artificial Intelligence Research*, Vol. 22, pp. 457–479.
13. **Fattah, M.A. (2014).** A hybrid machine learning model for multi-document summarization. *Applied Intelligence*, Springer Heidelberg, Vol 40, No. 4, pp. 592–600. DOI: 10.1007/s10489-013-0490-0.
14. **Ferreira, R., de Souza Cabral, L., Freitas, F. Lins, R.D., de Frana Silva, G., Simske, S.J., & Favaro, L. (2014).** A multi-document summarization system based on statistics and linguistic treatment. *Expert Systems with Applications*, Vol. 41, No. 13, pp. 5780–5787. DOI: 10.1016/j.eswa.2014.03.023.
15. **Gambhir, M., & Gupta, V. (2017).** Recent automatic text summarization techniques. *Artificical Intelligence Review*, Vol. 47, No. 1, pp. 1–66. DOI: 10.1007/s10462-016-9475-9.
16. **García-Hernández, R.A. et al. (2009).** Comparing commercial tools and state-of-the-art methods for generating text summaries. *Artificial Intelligence MICAI´09, Eighth Mexican International Conference on Artificial Intelligence*, pp.92–96. DOI: 10.1109/MICAI.2009.24.
17. **García-Hernández, R.A. & Ledeneva, Y. (2013).** Single Extractive Text Summarization Based on a Genetic Algorithm. ***Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F., Rodríguez, J.S., di Baja, G.S. (eds),*** *Pattern Recognition, MCPR 2013, Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, Vol. 7914, pp. 374–383. DOI: 10.1007/978-3-642-38989-4_38.
18. **Gong, Y. & Liu, X. (2001).** Generic text summarization using relevance measure and latent semantic analysis. *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval - SIGIR '01*, pp. 19–25. DOI: 10.1145/383952.383955.
19. **Kiss, T. & Strunk, J. (2006).** Unsupervised Multilingual Sentence Boundary Detection. *Computational Linguistics*, Vol. 32, No. 4, pp. 485–525. DOI: 10.1162/coli.2006.32.4.485.
20. **Kumar, N., Srinathan, K., & Varma, V. (2012).** Using Graph Based Mapping of Co-occurring Words and Closeness Centrality Score for Summarization Evaluation. *Gelbukh A. (eds) Computational Linguistics and Intelligent Text Processing. CICLing, LNCS*, Springer, Berlin, Heidelberg, Vol. 7182, pp. 353–365. DOI: 10.1007/978-3-642-28601-8_30.
21. **Ledeneva, Y., Gelbukh, A., & García-Hernández, R.A. (2008).** Terms derived from frequent sequences for extractive text summarization. *Lecture Notes in Computer Science (LNCS)*, Springer Heidelberg, Vol. 4919, pp. 593–604. DOI: 10.1007/978-3-540-78135-6_51.
22. **Ledeneva, Y. & García-Hernández, R.A. (2017).** *Generación automática de resúmenes Retos, propuestas y experimentos.* 1st ed. Universidad Autónoma del Estado de México.
23. **Lee, D.D., & Seung, H. S. (2001).** Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems*, Vol. 13, pp. 556–562.
24. **Lee, J.H., Park, S., Ahn, C.M., & Kim, D. (2009).** Automatic generic document summarization based on non-negative matrix factorization. *Information Processing & Management*, Vol. 45, No. 1, pp. 20–34. DOI: 10.1016/j.ipm.2008.06.002.
25. **Lin, C.Y. (2004).** ROUGE: A package for automatic evaluation of summaries. *Proceedings of Workshop on Text Summatization of ACL*, pp. 25–26.
26. **Lin, C.Y., & Hovy, E. (2003).** The potential and limitations of automatic sentence extraction for summarization. *Proceedings HLT-NAACL´03 on Text Summarization Workshop*, ACM, USA, Vol. 5, pp. 73–80. DOI: 10.3115/1119467.1119477.
27. **Lloret, E. & Palomar, M. (2012).** Text summarisation in progress: A literature review. *Artificial Intelligence Review*, Vol 37, No. 1, pp. 1–41. DOI: 10.1007/s10462-011-9216-z.
28. **Luhn, H.P. (1958).** The Automatic Creation of Literature Abstracts. *IBM Journal Research Development*, Vol. 2, No. 2, pp. 159–165. DOI: 10.1147/rd.22.0159.
29. **McDonald, R. (2007).** A Study of Global Inference Algorithms in Multi-Document Summarization.

Amati, G., Carpineto, C., Romano, G. (eds) *Advances in Information Retrieval, ECIR´07, LNCS*, Springer, Berlin, Heidelberg, Vol. 4425, pp. 557-564. DOI:10.1007/978-3-540-71496-5_51.

30. **Meena, Y.K. & Gopalani, D. (2015).** Evolutionary algorithms for extractive automatic text summarization. *Procedia Computer Science*, Vol. 48, pp. 244–249. DOI: 10.1016/j.procs.2015.04.177.
31. **Mendoza, M., Bonilla, S., Noguera, C., Cobos, C., & León, E. (2014).** Extractive single-document summarization based on genetic operators and guided local search. *Expert Systems with Applications*, Vol. 41, No. 9, pp. 4158–4169. DOI: 10.1016/j.eswa.2013.12.042.
32. **Mendoza, M., Cobos, C., Leon, E., Lozano, M., Rodríguez, F.J., & Herrera-Viedma, E. (2014).** A New Memetic Algorithm for Multi-Document Summarization based on CHC Algorithm and Greedy Search. ***Gelbukh, A., Espinoza, F.C., Galicia-Haro, S.N. (eds)***, *Human-Inspired Computing and Its Applications, MICAI´14, LNCS*, Springer, Cham, Vol. 8856, pp. 125-138. DOI: 10.1007/978-3-319-13647-9_14.
33. **Mihalcea, R. & Tarau, P. (2004).** TextRank: Bringing order into texts. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP´04),* Vol. 85, pp. 404–411.
34. **Mihalcea, R. & Tarau, P. (2005).** A Language Independent Algorithm for Single and Multiple Document Summarization. *Proceedings of IJCNLP,* Vol. 5, pp. 19–24.
35. **Ouyang, Y., Li, W., Li, S., & Lu, Q. (2011).** Applying regression models to query-focused multi-document summarization. *Information Processing & Management*, Vol. 47, No. 2, pp. 227–237. DOI: 10.1016/j.ipm.2010.03.005.
36. **Over, P. (2001).** Introduction to DUC-2001: an Intrinsic Evaluation of Generic News Text Summarization Systems. *DUC-2001: an Intrinsic Evaluation of Generic News Text Summarization Systems*, pp. 1-53.
37. **Over, P. & Ligget, W. (2004).** Introduction to DUC´02: an Intrinsic Evaluation of Generic News Text Summarization Systems Document Understanding Conferences, *Proceedings of DUC´04 Document Understanding Workshop,* pp. 1-48.
38. **Ponce, P. (2010).** *Inteligencia artificial con aplicaciones a la ingenieria.* 1st ed. Alfaomega, Mexico.
39. **Radev, D. R., Jing, H., Styś, M., & Tam, D. (2004).** *Centroid-based summarization of multiple documents.* Information Processing & Management*,* Vol. 40, No. 6, pp. 919–938. DOI: 10.1016/j.ipm.2003.10.006.
40. **Read, J., Dridan, R., Oepen, S., & Solberg, L.J. (2012).** Sentence Boundary Detection : A Long Solved Problem?. *Proceedings of COLING´12*, Vol. 12, pp. 985–994.
41. **Rojas, J., Ledeneva, Y., & García-Hernández, R. A. (2018).** Calculating the Significance of Automatic Extractive Text Summarization using a Genetic Algorithm. *Journal of Intelligent and Fuzzy Systems*.
42. **Russell, S. & Norvig, P. (2004).** *Inteligencia artificial: Un enfoque moderno*. 2nd ed. Madrid, Pearson Education S. A., Madrid Spain.
43. **Sidorov, G. (2013).** *Non-linear construction of n-grams in computational linguistics*. 1st edn., Sociedad Mexicana de Inteligencia Artificial, México.
44. **Sivanandam, S.N. & Deepa, S.N. (2008).** *Introduction to Genetic Algorithms.* 1st ed., Springer-Verlag Berlin Heidelberg.
45. **Steinberger, J. & Jezek, K. (2006).** Sentence compression for the LSA-based summarizer. *Proceedings of the 7th International conference on information systems implementation and modelling,* Vol. 180, pp. 141–148.
46. **Suanmali, L., Salim, N., & Binwahlan, M. S. (2011).** Genetic algorithm based sentence extraction for text summarization. *International Journal of Innovative Computing*, Vol. 1, No. 1, pp. 1-22.
47. **Torres-Moreno, J.M., Saggion, H., da Cunha, I., SanJuan, E., & Velázquez-Morales, P. (2010).** Summary Evaluation with and without References. *Polibits*, Vol. 42, pp. 13-20.
48. **Vázquez, E., Ledeneva, Y., & García-Hernández, R.A. (2018).** Sentence Features Relevance for Extractive Text Summarization using Genetic Algorithms. *Journal of Intelligent and Fuzzy Systems* (in press).
49. **Verma, R. & Lee, D. (2017).** Extractive Summarization: Limits, Compression, Generalized Model and Heuristics. *Computación y Sistemas*, Vol. 21, No 4, pp. 787–798. DOI: 10.13053/CyS-21-4-2885.
50. **Wang, D., Li, T., & Ding, C. (2010).** Weighted Feature Subset Non-Negative Matrix Factorization and its Applications to Document Understanding. *IEEE International Conference Data Minning*, Vol.

1550–4786, No. 10, pp. 541–550. DOI: 10.1109/ICDM.2010.47.

51. **Wang, D., Li, T., Zhu, S., & Ding, C. (2008).** Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization. *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* pp. 307–314. DOI: 10.1145/1390334.1390387.
52. **Wang, D., Zhu, S., Li, T., Chi, Y., & Gong, Y. (2011).** Integrating Document Clustering and Multidocument Summarization. *ACM Transactions Knowledge Discovery from Data (TKDD)*, Vol. 5, No. 3, pp. 1–26. DOI: 10.1145/1993077.1993078.
53. **Wang, D., Zhu, S., Li, T., & Gong, Y. (2009).** Multi-Document Summarization using Sentence-based Topic Models. *Proceeding ACLShort '09 Proceedings of the ACL-IJCNLP, Conference Short Papers*, ACM, Suntec, Singapore, pp. 297– 300.
54. **Wang, W.M., Li, Z., Wang, J.W., & Zheng, Z.H. (2017).** How far we can go with extractive text summarization? Heuristic methods to obtain near upper bounds. *Expert Systems with Applications*, Vol. 90, pp. 439–463. DOI: 10.1016/j.eswa.2017.08.040.
55. **Wan, X., & Yang, J. (2008).** Multi-Document Summarization Using Cluster-Based Link Analysis. *Proceedings of the 31st Annual International ACM SIGIR Conference Research and Developmeny Information Retrieval*, pp. 299–306. DOI: 10.1145/1390334.1390386.

Article received on 17/10/2017; accepted on 10/01/2018. Corresponding authors are Jonathan Rojas Simón, Yulia Ledeneva, and René Arnulfo García-Hernández.