Genetic Clustering Algorithm for Extractive Text Summarization

Sebastian Suarez Benjumea, ssuarezbe@unal.edu.co, Elizabeth Leon Guzman, eleonguz@unal.edu.co,
National University of Colombia, Bogota D.C, Colombia

***Abstract*—Automatic text summarization has become a relevant topic due to the information overload. This automatization aims to help humans and machines to deal with the vast amount of text data (structured and un-structured) offered on the web and deep web. In this paper a novel approach for automatic extractive text summarization called SENCLUS is presented. Using a genetic clustering algorithm, SENCLUS clusters the sentences as close representation of the text topics using a fitness function based on redundancy and coverage, and applies a scoring function to select the most relevant sentences of each topic to be part of the extractive summary. The approach was validated using the DUC2002 data set and ROUGE summary quality measures. The results shows that the approach is representative against the state of the art methods for extractive automatic text summarization.**

I. Introduction

Nowadays, the volume of text data is a lot bigger than 10 years ago. With the establishment of the web 2.0, Twitter, Facebook, online forums, social networks, blogs, self-newspaper (made by individuals and not big media companies) and others, the task of extracting value of such data maze becomes more important. This immense amount of digital data presents an obstacle for people who expect better tools that help them to deal with the information overload.

The objective of the text summarization is, "*obtain a reductive transformation of the base text to summarize via condensation, applying generalization and/or particularization of what it is important in the base text*" [12]. But this functional definition is incomplete because it does not take into account the particular interest of the user, which affects the usefulness of the summary. A better definition could be given combining the previous definition with the one given in [25]: "*The text summarization aims to produce a brief but accurate representation of the most important information present in the base text to satisfy a set of user/users information requirements*". Additionally, this definition has to deal with the fact that humans are not sure about what information should be in the summaries [22], as they are not able to foresee readerships interests and expectations. Then, automatize text summarization as well as the ways to validate it automatically become a difficult problem that requires new approaches to be solved.

The Automatic Text Summarization (ATS) is simply an automatic implementation of the text summarization applied to large volumes of documents (source text) to help humans and machines to cope with the vast amount of (structured and un-structured) data present on the web and deep-web[1]. Depending on the summary form it could be an extract or an abstract. The extract summary is composed by exact words or phrases which are present in the source text. The abstract summary is composed by words, phrases or expression that are not necessarily present in the source text; this type of summary is strongly related with the text understanding .

Different techniques have been used to solve the extractive ATS problem. The techniques spectrum include statistical based, graph based, machine learning based, and bio-inspired[17], [20], [25], [29]. However, few of them take into account the possibility of exploring clustering techniques. In this paper a novel single document extractive summarization approach based on sentences clustering for topics detection called SENCLUS is presented. SENCLUS uses a genetic clustering technique with a fitness function based on coverage and redundancy to automatically detect the text topics generating good extractive text summaries which cover the most important text topics with little algorithm configuration parameters. The algorithm showed good results across different experiments compared with the best algorithms reported for a single document extractive summarization.

This paper is organized as follows: background in Section 2, clustering summarization approach in Section 3, genetic clustering algorithm details in Section 4, application and experiments in Section 5, and conclusions and further research in Section 6.

II. Background

A. Document Summarization

The extractive summaries are those which are composed by exact words or phrases which are present in the source text. Then the problem of obtain extractive summaries from the base-text is reduced to find the smallest set of sentences that represent the whole text accurately [11]. In practice, the extractive summaries are constrained by size; for example, an extractive summary must not be

[1]The Deep Web (also called the Deep-net, the Invisible Web, the Undernet or the hidden Web) is World Wide Web content that is not part of the Surface Web, which is indexed by standard search engines. Wikipedia

978-1-4799-7560-0/15 $31.00 © 2015 IEEE
DOI 10.1109/SSCI.2015.139

longer than the 10% of the whole text where the length of the summary is calculated by the number of words. This implies that for real problems extractive summaries are really the best possible approximation of the base-text which fulfills the defined summary-constraints.

Different approaches have been proposed to solve this problem. In most cases the vector space representation is used [17] and different optimization techniques were used [17], [20], [25], [29].

B. Bio-inspired approaches

Genetic strategies has been used to solve the summarization problem. Works presented in [8], [13], [16] use Genetic approaches defining a set features $f = \{f_1, \ldots, f_n\}$ to extract the best sentences of a document optimizing the features weights w_i .

The work [8] uses eight sentence features: sentence length, similarity to the title, occurrence of non-essential information, sentence-to-centroid cohesion and others. The genetic algorithm is designed to find the best weight w_i for each feature f_i that maximizes the fitness function $f(x)$, which was defined as the average classification precision. The only differences with [16] are the use of 31 features and the support for multilingual problems. A similar approach is applied in [13] using a genetic algorithm to optimize a function with weight w_i for six features. In this work, the GA (Genetic algorithm) is used to optimize the weights while the GP (Genetic Programming) is used to optimize the set of fuzzy rules which leads to decide if a sentence should or should not belong to the summary.

In [3] the summarization problem is modeled as a *p-median* problem. The authors used a fitness function that balance the relevance, content coverage and diversity in the summary in order to find the best combination of sentences. The optimization method used in the genetic algorithm is Differential Evolution (DE) algorithm, which is a population based stochastic search technique.

In [24] the extractive summarization problem is solved using a Fuzzy Evolutionary Optimization Modeling (FEOM) which is applied to solve the sentence clustering problem, where each cluster center is sentence of the summary.

The MCMR function is used in[1], [2], [4]. MCMR is based on the idea that a summary sentence should have a high text coverage and low redundancy against the others summary sentences so the summary sentences are the ones that maximize this function. The approach described in [2] uses PSO, showing very good results that are supported also by the results obtained in [1], [4] where DE (Differential Evolution) is used instead.

In [23] a summarization method based on harmonic search is used to extract the most relevant sentences of the source text. The authors take into account three (3) factors in the objective function: (i) Topic Relation Factor: Measures the similarity between the sentences and the text title. (ii) Cohesion Factor: Similarity between the summary sentences. (iii) Legibility Factor: Similarity of one summary sentence with the next. The used harmonic vector is of length n (total number of sentences in the document), and a binary model where 1 means that the sentence belongs to the summary and 0 otherwise.

In [10] a Genetic Algorithm is used to find the optimal values of weight w_i for each feature f_i, where $i = 1, \ldots, 10$ using a training data set. After the training stage, the test stage is run. In this stage with a linear combination of $w_i f_i$, a new instance (sentence) is assigned to a real value. The top n sentences are selected to conform the final summary. In the GA a chromosome is represented as as the combination of all w_i, and a total of 100 generations selecting the 10 best individuals for the crossover process is performed to obtain the optimal individual.

In [6] an automatic summarization model which integrates fuzzy logic and swarm intelligence is proposed. The swarm model is used to calculate the values or weights w_i for the features f_i, where $i = 1, \ldots, 5$. Then the weights are used as inputs for the fuzzy inference system in order to assign a final value to the sentences, which is used to rank the sentences and select the top n sentences.

Finally, a recent algorithm called MA-MultiSumm[19] has shown great results compared with the state of the art algorithms using a evolutionary algorithm to select the best sentences applying a binary optimization.

C. Genetic Clustering

ECSAGO[15] is self adaptive genetic clustering algorithm that uses a niching technique. As in nature, niches in the clustering context correspond to different subspaces of the environment (clusters) that can support different types of life (data samples). This algorithm is able to adapt the genetic operators rates automatically at the same time it is evolving the clusters prototypes.

Each individual of the population represents a candidate cluster (center and scale). While the center of the cluster is evolved using the EA, it scale or size is updated using an iterative hill-climbing procedure. To preserve individuals in the niches already detected, a restriction in the mating is imposed: only individuals that belong to the same niche produce offspring.

One disadvantage of the Genetic Algorithms is the genetic operator tuning. This task consists in selecting the right group of genetic operator and assigning them a probability value to decide when to apply each one. To manage the genetic operators tuning parameters of the GA, ECSAGO uses Hybrid Adaptive Evolutionary Algorithm (HAEA) which is a parameter adaptation technique of Evolutionary Algorithms. At the same time that the individual is evolved, the rates of its genetic operators are updated, and a different operator can be applied in each iteration of the Evolutionary Clustering.

In HAEA, each individual is evolved independently of the other individuals in the population. In each generation, one genetic operator (crossover, mutation, etc.) is selected for each individual according to operator rates that are encoded into the individual.

III. SENCLUS Approach for Summarization

It is a fact that a writing is a representation of ideas that the writer intends to transmit. These ideas are also known as text topics. For very small documents the number of topics tends to one, but for longer writings this number is larger than one. Besides, on every writing there is a main idea or a set of main ideas around which the text is written. Therefore, there should be a set of relevant topics that dominate the full text. Each cluster corresponds to a topic and the size of each cluster (number of sentences) is the topic relevance. To select the best summary sentences SENCLUS ranks each sentence using a score value based on cluster relevance (number of sentences in the cluster) and the similarity between the sentence and the clusters centers to which it corresponds. Also, the advantages of using a clustering technique over a supervised technique is that it requires less human intervention. SENCLUS requires no specific number of topics and it is capable to detect the number of topics automatically without human intervention, which is an important advantage over other algorithms. This is possible due to the topics detection model in which SENCLUS is based.

The rest of this section is divided into two parts. The first explains the topics detection approach and the second explains the approach as an optimization problem.

A. Topics detection problem

A text is a written representation of one or more ideas that are intended to be expressed by the writer. Each one of this ideas could be represented by one or more sentences.

To summarize a text you need to detect the ideas or topics, and then select the sentences subset which is an optimal representation constrained by size .

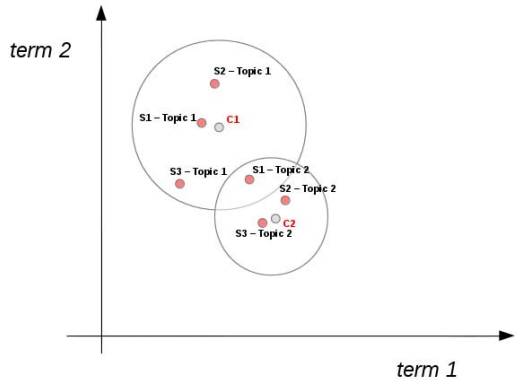
Lets define I as the set of ideas which the writer want to represent in the text. A property of I is that it do not change after the text was written and any misunderstanding of the text intention , also I, happens due to a bad writing or bad reading.

Until now it has been established that a text is an approximated representation of one or more ideas which the writer intends to communicate. Also, that the intended set of ideas, $idea_t \in I$, are susceptible to the writer's translating error and to the reader's understanding error.

When a text is given to summarize it, the text sentences are the written representation of the text intention which is the set of ideas emboided in the text.

The way of representing text numerically has been studied by many researchers who have worked with the problem of semantics, and they concluded that the meaning of words are closely connected to the statistics of word usage. The historical use of numerical vectors to represent text has showed how powerful and useful is this [28], [21]. In this case, the vector space represents the text as a $m \times n$ matrix in which the vertical axis represent the sentences and the horizontal axis represent the terms found in the text. Each sentence vector contain a numerical value with the term frequency–inverse document frequency ($tf - idf$).

Figure 1. Text representation at sentence level in the Vector Space 2-D



Contrary to term-frequency, a big $tf - idf$ value is an indicator of term relevance. Using the vector space representation, modeling the sentences as documents and terms as dimensions, is possible to cluster the sentences into k groups and use clusters centroids as representation of each $idea_t \in I$. This centroids are only numerical vectors which do not represent text sentences and therefore are *meta-sentences* or theorical sentences.

As can be seen in Figure 1, there is a set of theoretical centers for each cluster $\{c_1, c_2\}$. These theoretical centers are the *meta-sentences* represented in the vector space.

Based on that idea, the sentences are clustered and their clusters centers are used as a good approximation of the text *meta-sentences* or topics.

B. Proposed Model

Lets define $S = \{s_1, s_2, \ldots, s_m\}$ where $s_x \in S$, as the set of sentences extracted from the analyzed text and $T = \{t_1, t_2, \ldots, t_n\}$ as the text terms which implies that each sentence vector s_x has a size n or $\mid s_x \mid = n$.

To measure if two sentences $s_i, s_j \in S$ talk about a similar topic, the cosine similarity between two vectors defined in (1) is used.

$$sim(s_i, s_j) = \frac{\sum_{x=1}^{n} S_{ix} \times s_{jx}}{\mid s_i \mid \times \mid s_j \mid} \tag{1}$$

As it was mentioned an $idea_t \in I$ is represented by one or more sentences, therefore similar sentences must represent a part of a same $idea_t$. Also, in a document there are topics that could be subtopics of another topic, making some topics more relevant than others. Then, a good sentences cluster is compound by relevant sentences which are similar with each other. Two measures are defined to evaluate the sentences clusters. Coverage measures the relevance of a sentence and Redudancy measures how similar or compact is a group of sentences.

$$coverage(s_i) = \sum sim(s_i, S) \tag{2}$$

$$redundancy(s_j) = \sum_{s_k \in ss} sim(s_j, s_k) \quad (3)$$

Coverage defined in (2) models how relevant is a sentence s_i in the text. The $\arg_i \max(coverage(s_i))$ will be such s_i which fulfills the condition $\sum_{s_j \in S} sim(s_i, s_j) = 1$ always TRUE. Then the higher is the coverage, the better representation is the sentence of the analyzed text. On the other hand, **Redundancy** defined in (3) models how much a sentence s_j belongs to a topic represented by a subset $ss_x \in S$. And in the same way that happens with the coverage, the higher is the redundancy of s_j, the better is s_j a representation of the sentences subset $ss_x \in S$. Finally, if $\mid ss_x \mid < \mid S \mid$ then $redundancy(s_i) < coverage(s_i)$.

The objective function use α as a relevance factor between redundancy and coverage. By default the value of $\alpha = 0.5$ is used ,so redundancy and coverage are equally relevant in the summary. The complete objective function to be optimized is defined in (4).

$$\arg_{s_k, s_m, \ldots} f(s_k, s_l, \ldots) = \sum_{x=1}^{k} h(ss_x)$$

$$h(ss_x) = \sum_{s_y \in ss_x} (1-\alpha) \left(\frac{redundancy(s_y)}{\mid ss_x \mid} \right) - (\alpha) \left(\frac{coverage(s_y)}{\mid S \mid} \right) \quad (4)$$

, where k is the expected number of topics.

The function $f(s_k, s_l, \ldots)$ will be maximized to find the set of meta-topics or cluster centers $s_k, s_l, \ldots$ that maximize the $h(ss_x)$ of each cluster or group. And theoretically $h(ss_x)$ reach their maximum when all sentences in ss_x belongs to the same topic.

IV. SENCLUS Algorithm

The proposed objective function presented in equation (4) is a multimodal function, therefore it can be solved using a niching strategy.

ECSAGO is a genetic clustering algorithm which is robust to noise and has the ability to detect the number of clusters automatically using niching. The advantage of genetic algorithms over other methods is that, with a good set of genetic operators, a good solution could be found in a time t; and t depends on the termination criteria for the algorithm, configured at the beginning. Also, genetic operators like selection, mutation and crossover allow to explore the function landscape and refine the promissory areas until find the local or global optimal .

ECSAGO has been used for document clustering showing good results[14], but SENCLUS take all ECSAGO advantages to solve the proposed objective function defined in (4) which is not density based as the ECSAGO fitness function. The ECSAGO fitness function is the density of the hypothetical cluster which is completely different to SENCLUS fitness function based on redundancy and coverage. Also, because SENCLUS fitness function is not based on density, the SENCLUS sigma is different and it is used as a topic border. These were the reasons to create SENCLUS.

SENCLUS keeps the concept of a dense clusters that represent topics along with Deterministic Crowding, restricted mating and the HAEA to adapt the relevance of each operator to decide if it should be used more or less often. SENCLUS adopt a radius called *sigma* used to model the topics boundaries in the vector space representation. In other words, SENCLUS decides if a sentence belongs or not belongs to cluster using the condition defined in(5)

$$IF\ sim(s_i, c_j) > sigma_{c_j} \quad THEN\ s_i \in c_j \quad (5)$$

, where c_j is a meta-sentence or cluster center.

After the sentences were clustered, the clustering results are analyzed. A relevance function is used to give a score to each sentence, and by this score the sentences will be ranked.

The $score(s_j)$ calculates the similarity between the sentence and each cluster center $sim(s_j, c_i)$.

The pipeline design for extractive summary generation is shown Figure 2 and SENCLUS pseudo code is shown in Algorithm 1.

A. Representation

Each individual represents a potential meta-sentence that represents a topic. These individuals are initialized randomly selecting vector representations of sentences present in the text, using sentences as documents and terms as dimensions. Each individual has a length n, where n is the number of terms present in the text, and each gene is a float number representing the term relevance.

B. Fitness function

The fitness value for the i^{th} candidate center c_i, is defined using the function :

$$f(c_i) = (1-\alpha)\, redundancy(c_i) - (\alpha) coverage(c_i) \quad (6)$$

,where S is the set of sentences extracted from the text, redundancy is (3) and coverage (2).

The fitness value of each individual requires *sigma* to allocating sentences in the clusters or topics using the function defined in (5). Also, sigma allow soft clustering and it delimits each cluster for the Deterministic Crowding.

The cluster radius or *sigma* represents the topic scope in the vector space. The radius is updated with the mean difference between the similarity of each sentence against the cluster center and the coverage of the sentence. The radius will reach their maximum when the cluster sentences belong to only one topic with a high confidence represented with a good sentence coverage and a high similarity between cluster sentences and cluster center.

The candidate center c_i radius $sigma(c_i)$ is defined in (7).

Algorithm 1 SENCLUS pseudo code

Calculate coverage for each sentence in S
Select random sentences as initial population
Assign the $sigma_{initial}$ to the all the initial population
WHILE $generation < maxGenerations$:
 FOR *individual* IN *population*:
  $individual_{fitness} = calculateFitness(individual_{vector}, individual_{Sigma}, population)$
 $parents = generateCouples(population)$
 FOR *parentsCouple* in *parents*:
  IF $restrictedMating(parentsCouple)$:
   $children = applyOperatorHAEAwithCrossover(parentsCouple)$
  ELSE
   $children = applyOperatorHAEA(parentsCouple)$
  FOR *child* IN *children*:
   $child_{sigma} = updateSigma(child_{sigma}, child_{vector}, population)$
  $winners = deterministicCrowding(children, parentsCouple)$
  $replace(parentsCouple, winners, population)$
$sentencesScoring(population, S)$

Algorithm 2 sentences scoring

FOR *sentence* IN S:
 FOR *individual* in *population*:
  IF $similarity(sentence, ind) > individual_{sigma}$:
   $sentence_{clusters} = concat(sentence_{clusters}, individual_{id})$
FOR *sentence* IN S:
 FOR *clusterCenter* IN $sentence_{clusters}$:
  $sentence_{score} = similarity(sentence, clusterCenter) * (\frac{1}{sentence_{textPosition}})$
$sort(sentences_{score})$

$$sigma(c_i)_t = sigma(c_i)_{t-1} + \frac{\sum_{j \in c_i} sim(s_j, c_i) - \frac{sim(s_j,S)}{\mid S \mid}}{\mid c_i \mid} \quad (7)$$

C. Genetic Operators

a) One-point Crossover : A single crossover point on both parents organism strings is selected. All data beyond that point in either organism string is swapped between the two parent organisms. The resulting organisms are the children.

b) Two-point Crossover : Two-point crossover calls for two points to be selected on the parent organism strings. Everything between the two points is swapped between the parent organisms, rendering two child organisms.

c) Heuristic Crossover: A crossover operator that uses the fitness values of the two parent chromosomes to determine the direction of the search. The offspring are created according to the following equations:

$$child_a = \beta(Parent_{best} - Parent_{worst}) + Parent_{best}$$

$$child_b = \beta Parent_{best} + (1 - \beta)Parent_{worst}$$

$$0 \leq \beta \leq 1 \text{ random}$$

d) Mutation: It is analogous to biological mutation. Mutation alters one gene value in a chromosome from its initial state randomly.

e) Gaussian Mutation: Changes one component of the encoded real vector with a number randomly generated following a Gaussian distribution using as mean the old value of the component, and the given standard deviation.

D. Summary Sentences Selection

This scoring function exists because an extractive summary could not be formed by meta-sentences which are float vectors, so a set of the best sentences should be selected. After the sentences have their score, they are sorted and added from the top to the bottom, until there is no space in the summary.

The sentence scoring function is defined in (8).

$$score(s_j) = sim(s_j, c_{i,s_j \in c_i}) \times \left(\frac{1}{pos(s_j)}\right) \quad (8)$$

Finally the best r sentences are selected, where r depends on the summary length. The pseudo code is shown in Algorithm 2.

V. Application and Experiments

In this research SENCLUS was applied to generate single document extractive summaries.

With the lack of the clustering based algorithms for extractive text summarization, three popular clustering algorithms were coded to test their extracts quality against the proposed genetic clustering algorithm for extractive ATS. The selected clustering algorithms were: K-means, Genetic K-means and Non-Negative Matrix Factorization[27], [31].

The K-means and GK-means, generate hard clusters to which the scoring function $score(s_j)$ is applied to get extractive summaries.

To measure the extracts quality, the ROUGE[32] measure is used. The reasons to use it are: their proved usefulness as extract quality measure [9], [26] and their popularity. Then, it could be easier to compare the obtained results against other algorithms without repeating their experiments.

A. DUC 2002 data set

The DUC 2002 data set provided by the Document Understanding Conference[7], is a data set prepared for testing task of single and multiple document summarization. The documents of the DUC 2002 collection are categorized in subgroups and each subgroup has a set of control summaries which were generated by experts.

For single document summaries, the generated extracts summaries have a maximum of 200 words. The DUC 2002 composition details are described in Table I.

Table I
DUC 2002 Details

	DUC 2002
number of document collections	59
number of documents in each collection	10
data source	TREC
summary length	200 words

B. Preprocessing

Before apply the algorithm the text is parsed to extract the sentences, removing the special characters of the sentences, and then represent the sentences using the vector space model removing stops words and applying stemming to words. The overall pipeline can be seen in Figure 2.

C. Parameters and Experiments settings

The algorithm require 3 parameters to start. They are: population size, generations and initial-sigma.

The population size must a number greater or equal to the expected number of topics in the text. If this parameter is too small, it is highly possible that the detected topics are too broad and the extract could be bad. For this experiments the expected number of clusters or population size is set to $m/2$ being m the number of sentences in the text. This parameter is also used for the K-means, GK-means and NMF.

The number of generations, was found running different experiments, and the results showed that between 50 and 150 generations the extracts reach their maximum value. Greater values makes almost no difference in the final results. All the algorithms was tested with the same number of iterations.

And finally, the initial-sigma should be a small number so the algorithm could adjust the sigma correctly during the generations. If the sigma it too big, the experiments showed that is really hard that sigma can be adjusted correctly.

Each experiment was run 1000 times and the recommended initial parameters to run the algorithm are listed in Table II.

Table II
Initial parameters

Parameter	Value
Population Size	m/2
Max-Generations or Max-iterations	50, 150
initial-sigma	0.00001

D. Results and Discussion

The Table III list the best result for the set of best algorithm configurations using different genetic operators and scoring functions. Analyzing the results in Table III it can be established that in cases when only exploration was used, the results were around 0.42 ± 0.0031, but when exploration was mixed with exploitation the result improved up to 0.473 ± 0.0067. In some experiments, the factor $\left(\frac{1}{pos(s_j)}\right)$ was removed from the function $score(s_j)$ to check the relevance of the scoring function. This simplified score function without the mentioned factor is called **Basic** score function; the other one is mentioned as **Position** score function.

The Table IV compares the proposed genetic clustering algorithm against other algorithms over their reported results using the DUC2002 data set and ROUGE. The SENCLUS perform well compared against the best state of the algorithm and the reported results are not too far from the best ones.

Table IV
DUC2002 results

Algorithm	Rouge-1	Rouge-2
UnifiedRank[30]	0.4847	0.2146
MA-MultiSumm[18]	0.4828	0.2284
SENCLUS	**0.4795**	**0.2200**
DE[5]	0.4699	0.1236
FEOM[24]	0.4657	0.1249
NMF	0.4036	0.1611
K-means	0.3485	0.1300
GK-means	0.3377	0.1301

The text in Figure 3 is a document from DUC 2002 and the content in Figure 4 is the SENCLUS generated summary. The extract summary showed in Figure 4 is one of the best extractive summaries generated by the algorithm. A manual verification of the results indicates that in some cases the algorithm is not capable to generate good summaries because some documents have irregular

Figure 2. Pipeline Design

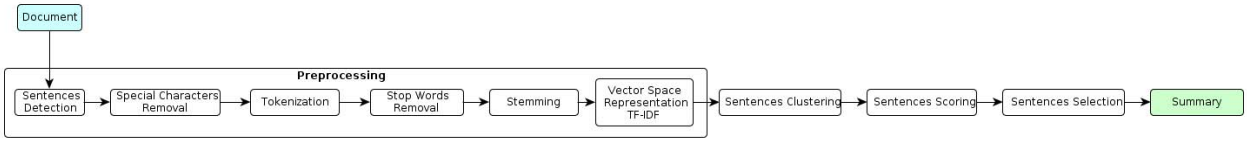


Table III
SENCLUS RESULTS SUMMARY

Rouge-1	Rouge-2	Iter	Score	Genetic Operators
0.47952	0.2200	150	Position	one-point-crossover, heuristic-crossover, standard-mutation
0.47951	0.2200	50	Position	one-point-crossover, heuristic-crossover, standard-mutation
0.4731	0.2159	150	Position	one-point-crossover, standard-mutation
0.4718	0.2142	50	Position	one-point-crossover, standard-mutation
0.4664	0.2112	150	Basic	one-point-crossover, gaussian-mutation
0.4659	0.2110	50	Basic	one-point-crossover, gaussian-mutation
0.4233	0.1900	150	Position	standard-mutation
0.4245	0.1890	50	Position	standard-mutation

structures which made impossible parse those documents correctly in a automatic way. This issue is strongly related with the structure of DUC 2002 documents which uses tags to name document sections. Then, it is possible to get better results with a cleaner data set, but in real problems is hard to find a clean data set therefore is better to report the results obtained from the original documents without manual modifications.

VI. Conclusions and Further Research

The results obtained from the DUC2002 showed that the novel approach generates good extractive summaries that are comparable with the state of the art algorithms applied to the same data set. The innovative features of SENCLUS are: a topics detection model using sentences clustering, the use of sigma to delimit a topic in the vector space, the summarization model, and all the advantages offered by the ECSAGO to solve clustering problems like the Deterministic Crowding (DC) and HAEA.

The proposed approach for topics detection constitutes an interesting modeling of the text summarization problem that could be developed to solve multi-document extractive summarization. Because the main objective is to detect the topics by clustering the sentences around them to give relevance to those sentences, for a multi-document text summarization it could be expected more separated clusters for texts talking about different topics. Therefore the problem could be solved in a similar way as the single document problem.

Finally, SENCLUS results could be improved using other sentences similarity functions or other genetic operators, postulates this work as a very promissory approach to the extractive text summarization problem.

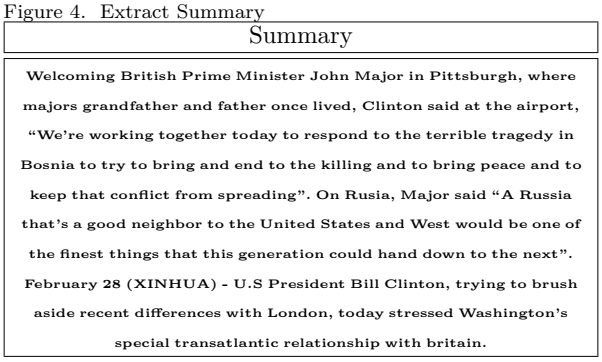
Figure 3. Text Example

Text
TITLE:President Clinton, John Major Emphasize 'Special Relationship'. Article Type:BFN [Text] Washington, February 28 (XINHUA) – U.S. President Bill Clinton, trying to brush aside recent differences with London, today stressed Washington's special transatlantic relationship with Britain. Welcoming British Prime Minister John Major in Pittsburgh, where major's grandfather and father once lived, Clinton said at the airport, "We're working together today to respond to the terrible tragedy in Bosnia to try to bring an end to the killing and to bring peace and to keep that conflict from spreading." For his part, Major said, pressure would be increased for the peace that every sensitive person wishes to see in that war-torn and troubled land. On Russia, Major said "A Russia that's a good neighbor to the United States and West would be one of the finest things that this generation could hand down to the next." Clinton will then share his Air Force One back to the nation's capital. Major will spend a night at the White House, the first foreign head of state to have this honor since Clinton became President. On Tuesday [1 March], the two leaders will begin their discussions on a wide range of issues including Russia, Bosnia, Northern Ireland and the world trade. The two will also discuss Northern Ireland and "what to do with NATO," Clinton said. Clinton and major will meet again in June in Europe during the commemoration of the 50th anniversary of D-Day of the second world war. Major said Clinton would visit Britain, and perhaps the Oxford University, Clinton's alma mater, during the June visit.

References

[1] Rasim M. Alguliev, Ramiz M. Aliguliyev, and Makrufa S. Hajirahimova. *Expert Systems with Applications*, 39(16):12460 – 12473, 2012.

Figure 4. Extract Summary



[2] Rasim M. Alguliev, Ramiz M. Aliguliyev, Makrufa S. Hajirahimova, and Chingiz A. Mehdiyev. Mcmr: Maximum coverage and minimum redundant text summarization model. *Expert Systems with Applications*, 38(12):14514 – 14522, 2011.

[3] Rasim M. Alguliev, Ramiz M. Aliguliyev, and Nijat R. Isazade. Desamc+docsum: Differential evolution with self-adaptive mutation and crossover parameters for multi-document summarization. *Knowledge-Based Systems*, 36(0):21 – 38, 2012.

[4] Rasim M. Alguliev, 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):213 – 222, 2011.

[5] Ramiz M. Aliguliyev. A new sentence similarity measure and sentence based extractive technique for automatic text summarization. *Expert Systems with Applications*, 36(4):7764 – 7772, 2009.

[6] Mohammed Salem Binwahlan, Naomie Salim, and Ladda Suanmali. Fuzzy swarm based text summarization 1.

[7] Document Understading Conference. Duc 2002 data set description and conference guidelines. http://www-nlpir.nist.gov/projects/duc/guidelines/2002.html, 2002.

[8] Pooya Khosraviyan Dehkordi, Dr. Farshad Kumarci, and Dr. Hamid Khosravi. 57 text summarization based on genetic programming.

[9] Bonnie J. Dorr, Christof Monz, Stacy President, Richard Schwartz, and David Zajic. Methodology for extrinsic evaluation of text summarization: Does rouge correlate. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 1–8, 2005.

[10] Mohamed Abdel Fattah and Fuji Ren. Ga, mr, ffnn, PNN and GMM based models for automatic text summarization. *Computer Speech & Language*, 23(1):126 – 144, 2009.

[11] Vishal Gupta and Gurpreet Lehal. A survey of text summarization extractive techniques. *Journal of Emerging Technologies in Web Intelligence*, 2(3), 2010.

[12] Karen . Jones. Automatic summarizing: factors and directions. In Inderjeet Mani and Mark T. Maybury, editors, *Advances in automatic text summarization*, chapter 1. The MIT Press, 1999.

[13] A. Kiani and M.R. Akbarzadeh. Automatic text summarization using hybrid fuzzy ga-gp. In *Fuzzy Systems, 2006 IEEE International Conference on*, pages 977–983, 2006.

[14] Elizabeth León, Jonatan Gómez, and Olfa Nasraoui. A genetic niching algorithm with self-adaptating operator rates for document clustering. In *Eighth Latin American Web Congress, LA-WEB 2012, Cartagena de Indias, Colombia, October 25-27, 2012*, pages 79–86, 2012.

[15] Elizabeth León, Olfa Nasraoui, and Jonatan Gómez. ECSAGO: evolutionary clustering with self adaptive genetic operators. In *IEEE International Conference on Evolutionary Computation, CEC 2006, part of WCCI 2006, Vancouver, BC, Canada, 16-21 July 2006*, pages 1768–1775, 2006.

[16] Marina Litvak, Mark Last, and Menahem Friedman. A new approach to improving multilingual summarization using a genetic algorithm. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 927–936, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[17] Elena Lloret and Manuel Palomar. Text summarisation in progress: a literature review. *Artificial Intelligence Review*, 37(1):1–41, 2012.

[18] Martha Mendoza, Susana Bonilla, Clara Noguera, Carlos Cobos, and Elizabeth León. Extractive single-document summarization based on genetic operators and guided local search. *Expert Syst. Appl.*, 41(9):4158–4169, 2014.

[19] Martha Mendoza, Carlos Cobos, Elizabeth León Guzman, Manuel Lozano, Francisco J. Rodríguez, and Enrique Herrera-Viedma. A new memetic algorithm for multi-document summarization based on CHC algorithm and greedy search. In *Human-Inspired Computing and Its Applications - 13th Mexican International Conference on Artificial Intelligence, MICAI 2014, Tuxtla Gutiérrez, Mexico, November 16-22, 2014. Proceedings, Part I*, pages 125–138, 2014.

[20] Michael W. Berry and Malu Castellanos, editors. *Survey of Text Mining II - Clustering, Classification, and Retrieval*. Springer London, 2008.

[21] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.

[22] Karolina Owczarzak, John M. Conroy, Hoa Trang Dang, and Ani Nenkova. An assessment of the accuracy of automatic evaluation in summarization. In *Proceedings of Workshop on Evaluation Metrics and System Comparison for Automatic Summarization*, pages 1–9, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[23] Ehsan Shareghi and Leila Sharif Hassanabadi. Text summarization with harmony search algorithm-based sentence extraction. In *Proceedings of the 5th International Conference on Soft Computing As Transdisciplinary Science and Technology*, CSTST '08, pages 226–231, New York, NY, USA, 2008. ACM.

[24] Wei Song, Lim Cheon Choi, Soon Cheol Park, and Xiao Feng Ding. Fuzzy evolutionary optimization modeling and its applications to unsupervised categorization and extractive summarization. *Expert Syst. Appl.*, 38(8):9112–9121, August 2011.

[25] Josef Steinberger and Karel Jezek. Text summarization: An old challenge and new approaches. In Ajith Abraham, Aboul-Ella Hassanien, André Ponce Leon F. de Carvalho, and Václav Snásel, editors, *Foundations of Computational, IntelligenceVolume 6*, volume 206 of *Studies in Computational Intelligence*, pages 127–149. Springer Berlin Heidelberg, 2009.

[26] Josef Steinberger and Karel Jezek. Evaluation measures for text summarization. *COMPUTING AND INFORMATICS*, 28(2), 2012.

[27] D. Tsarev, M. Petrovskiy, and I. Mashechkin. Using NMF-based text summarization to improve supervised and unsupervised classification. *Hybrid Intelligent Systems (HIS), 2011 11th International Conference on*, 5 December 2011.

[28] Peter D Turney, Patrick Pantel, et al. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37(1):141–188, 2010.

[29] Vishal Gupta and Gurpreet S. Lehal. A Survey of Text Mining Techniques and Applications. *JOURNAL OF EMERGING TECHNOLOGIES IN WEB INTELLIGENCE*, August 2009.

[30] Xiaojun Wan. Towards a unified approach to simultaneous single-document and multi-document summarizations. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1137–1145, Beijing, China, August 2010. Coling 2010 Organizing Committee.

[31] W. Xu, X Liu, and Y Gong. Document clustering based on non-negative matrix factorization. In *In proceeding of ACM SIGIR'03 (2003)*, 2003.

[32] Chin yew Lin. Rouge: a package for automatic evaluation of summaries. pages 25–26, 2004.