Managing Genetic Algorithm Parameters to Improve SegGen, a Thematic Segmentation Algorithm

Neslihan Sirin Saygili
Institute of Science
Galatasaray
University Istanbul,
Turkey
11411020@ogr.gsu.edu.tr

Tankut Acarman
Institute of Science
Galatasaray
University
Istanbul, Turkey
tacarman@gsu.edu.tr

Tassadit Amghar
LERIA
Universite´ d'Angers
Angers, France
amghar@univ-angers.fr

Bernard Levrat
Institute of Science
Galatasaray
University Istanbul,
Turkey
& LERIA,
Université d'Angers
Angers, France
blevrat@gsu.edu.tr

***Abstract*—SegGen [1] is a linear thematic segmentation algorithm grounded on a variant of the Strength Pareto Evolutionary Algorithm [2] and aims at optimizing the two criteria of the Salton's [3] definition of segments: a segment is a part of text whose internal cohesion and dissimilarity with its adjacent segments are maximal. This paper describes improvements that have been implemented in the approach taken by SegGen by tuning the genetic algorithm parameters according with the evolution of the quality of the generated populations. Two kinds of reasons originate the tuning of the parameters and have been implemented here. First as it could be measured by the values of global criteria of the population quality, the global quality of the generated populations increases as the process goes and it seems reasonable to set values to parameters and define new operators, which favor intensification and diminish diversification factors in the search process. Second since individuals in the populations are plausible segmentations it seems reasonable to weight sentences in the current segmentation depending on their distance to the boundaries of the segment they belong to for the calculus of similarities between sentences implied in the two criteria to be optimized. Although this tuning of the parameters of the algorithm currently rests on estimations based on experiments, first results are promising.**

Keywords—***thematic segmentation; genetic algorithm; multi-objective optimization problem.***

I. INTRODUCTION

Due to the huge increase in the number of available text databases in recent years, the need for efficient searching methods has become a major challenge for information retrieval. Moreover efficiency in the accessibility to the relevant information, satisfying user's information need is now becoming a crucial issue in the choice of a searching system. Because documents of a document search base are not specifically built for the task they are used for, and this is particularly the case of the web, users generally consider only small parts of documents returned as response to their queries as being relevant. This is one of the reason for which researches were initiated in the aim to give access to parts of returned documents and this constitute a subfield of the domain of information retrieval known as *passage retrieval* [4, 5].

Using automatic text summarization in digital libraries offers potential benefits but this is dependent on having tools like efficient text segmenters to built the abstracts. Reference [6] remarks that segmentation is a good way to thoroughly ensure the representation of the various topics of a document in a summary. In information retrieval, the burden to retrieve information relevant to a query in large texts is a drawback due in particular to the fact that the documents have not specifically been conceived to answer to the particular query for which they are furnished as a response. This is another reason for which the thematic text segmentation of textual documents has taken more importance in this domain and has given rise to *passage retrieval*, as a subfield of information retrieval, where the aim is to access directly to the parts of the documents containing the relevant information more efficiently than in traditional information retrieval where only whole texts are considered [3].

Thematic segmentation can be characterized as the process of separating written text into meaningful homogeneous units in accordance with the criteria stated in Salton's definition [3] which states that segmentation consists of splitting a text into parts, the segments, such that the internal cohesion of segments and the dissimilarity between adjacent segments is maximum. Following this definition, automatic text segmentation could be seen as determining the most important thematic breaks by setting the boundaries in a document

guided by these criteria. What can be considered as being boundaries depends on the segmentation units, which can be words, sentences, paragraphs or text passages giving rise to different kinds of segment from segments being parts of sentences to chapters of a book.

SegGen [1] proposes an original and efficient way to cope with the problem of linear text segmentation since it states the segmentation problem as a bi-objective optimization problem grounded on the criteria of the Salton's definition of segments previously evocated. To solve this problem, SegGen uses an implementation of the multi-objective algorithm SPEA [2], a classical multi-objective algorithm.

This paper presents the first results of new improvements in the approach taken by to SegGen [1]. These improvements are guided by two main ideas, which inspired autonomous search [7]. The first one rests on some general principles of *autonomous search*, which consists in modifying the parameters and operators of the genetic algorithm along with the increasing quality of the generated population through the generations. The second improvement is also to take into account the increasing of quality of the population as the process evolves, but to do so with taking into account the nature of the coding of individuals which in this case are segmentation instances represented by binary vectors corresponding to the positions of the boundaries of the segmentations.

Section 2 presents motivation of our work and preliminaries. Section 3 describes the various improvements we have added to SegGen and Section 4 details the first results of this experimental work, which appears to be promising. We conclude with some interesting tracks for our future work on this research and particularly by using learning methods to automatically tune the values of the various parameters instead of the empirical guess we have done in the current state of this research.

II. MOTIVATIONS AND PRELIMINARY WORKS

One of the main drawback to the majority of existing segmentation methods is that the criteria used to set boundaries between segments are local boundaries. It means that similarities between sentences are examined locally nearby the potential segments and do not consider the whole potential segmentation. There are many segmentation methods that rely on statistical approaches, such as TextTiling [8], C99 [9], DotPlotting [10], Segmenter [11]. The common point of statistical segmentation methods is that they determine the thematic changes via lexical inventory variations, so for example they set the boundaries by using sliding windows on the text to measure the variation of the level of local cohesion, setting the boundaries where local cohesion is the lowest. In such methods, thematic similarities between segments are calculated on the basis of the distribution of the meaningful lexical inventory in each segment. And for that, most of the existing segmentation methods determine a sliding window for finding out dissimilarity measures in consecutive positions of the sliding window or the evolution of a measure of its cohesion. For instance; TextTiling [8] algorithm uses a sliding window, which determines *block*s in the text, and calculates the value of the dissimilarity of adjacent blocks based on differences between lexical inventories in adjacent blocks (in fact it uses a vector representation of textual units and the measure cosine for that). Thematic changes are detected on the base of the evolution of the dissimilarities between adjacent sliding blocks. Thus, significant vocabulary changes are seen at points with subtopic change. However, the efficiency of such methods is very dependent of the dimension of the size of the sliding windows. Reference [12] indicates that small modifications of the window size could greatly influence the position setting of the boundaries between segments leading to over or under segmentation of the text depending on a too small or too large window size.

Contrary to these algorithms which rest on sliding windows and set the boundaries between segments on local criteria, SegGen algorithm permits to have a global view on all the potential segments to take a decision since all the boundaries between potential segments are set at the same time rendering. Details of SegGen algorithm are explained in the following section.

A presentation of SegGen

SegGen algorithm uses genetic algorithms for text segmentation. In SegGen, the main aim is to find out the subtopics, which create internal coherence and are distinguished from other parts of the text. Hence, the algorithm has two objective functions such as internal cohesion and dissimilarity between adjacent parts. Due to the existing of two objective functions, SegGen can be classified as a multi objective algorithm. On the other hand, SegGen is implemented as a variation of Strength Pareto Evolutionary Algorithm [2], for this reason the algorithm uses Pareto optimality that it is not possible to achieve any one solution better without making at least one solution worse off. At the start, SegGen represents individuals as binary vectors that means there are "1"s and "0"s, if x_i = 1 there is a boundary between sentence i and i+1, else there is no boundary between these sentences. The optimization objectives of SegGen are internal cohesion of segments $C(\vec{x}) \in [0,1]$ and dissimilarity between adjacent segments $D(\vec{x}) \in [0,1]$. SegGen formulates its optimizer as (1),

$$O = \{\, \vec{x} \in \{0,1\}^{ns-1} \mid \nexists\, \vec{x'} \in \{0,1\}^{ns-1}, \; ((C(\vec{x}) < C(\vec{x'})) \wedge \left(D(\vec{x}) < D(\vec{x'})\right)\} \qquad (1)$$

As shown in *Fig. 1*, due to in need of maximum values of the similarity of internal cohesion and dissimilarity between adjacent segments, as for that given example the greater values are better than smaller values, points that lie on the Pareto frontier line are non dominated by any other, and smaller value points are dominated by frontier points. SegGen algorithm uses an external archive $\bar{P}$ to keep the non-dominated individuals with reference to both criteria and a current population P_t. Individuals selected from these two

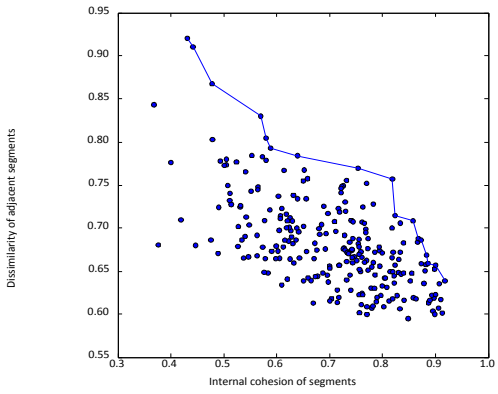

Fig. 1. Some Pareto frontiers are produced by SegGen.

populations to produce new generations due to genetic operations. Within this period, calculation of the fitness function consists of two steps. First step is calculation of hardness value of each non-dominated individual in $\bar{P}$. The hardness value of an individual is count of dominating individuals in P_t. Concurrently, the fitness value of an element, which belongs to $\bar{P}$, equals the inverse of own hardness value. The second part of the process is summing of fitness values of all dominated element from $\bar{P}$. The new generation individuals substitute the current population and are used to update $\bar{P}$. At the end of the entire iterations, a set of potential results in the external archive $\bar{P}$.

III. TUNING PARAMETERS OF SEGGEN

As mentioned previous chapter, SegGen is a text segmentation method that benefits from genetic algorithm to solve bi-criteria optimization problem. However, the genetic algorithm used by SegGen is exact a generic type of genetic algorithm. As evocated previously, SegGen represents the individuals of the population as binary vectors that means there are "1"s and "0"s, if x_i = 1 there is a boundary between sentence i and i+1, else there is no boundary between these sentences. A boundary indicates that a thematic changing occurs between adjacent sentences. In our study, we attempt to create a genetic algorithm that is a specific interpretation for text segmentation. Thus, the interpretation of an individual as a potential segmentation more appropriate than only bit string representation. As usual in genetic algorithms, crossover operator is widely used to lead population to converge on the good solutions and mutation operator is mostly used to provide exploration. In this study, a reason of tuning of the parameters could be measured by the values of global criteria of the population quality, the global quality of the generated populations increases as the process goes and it seems reasonable to set values to parameters and define new operators, which favor intensification and diminish diversification factors in the search process.

A. Tuning Mutation Operator

We change probabilities of mutation in general and more specifically. SegGen represents the individuals of population via binary vectors. The "1" bits of the vector indicate that there is a boundary between two sentences in the text that means a thematic changing occurs between adjacent sentences. We can effectively use the boundary adding or dropping by bit flip for getting the accurate result in mutation operator. Genetic algorithm is one form of local search that starts from initial configuration and makes evolution-based changes to the configuration until reaching the goal. Since the methods are attempting to optimize a set of objectives but will mostly find local maxima rather than a global maximum, local search methods are also known as local optimization [13]. In this study, we will change the mutation probability so that we can release the algorithm if it sticks to the local maxima so it can jump to the global maximum. The departing point is the fact that mutation causes random changes on individuals by its nature. As new and differing individuals join the population, increasing diversity of the population offers a chance for reaching more qualified individuals. There is low probability of mutation in early generations of the program. In subsequent generations, mutation probability is either increased or stabilized taking into consideration average quality of the population. If the program is close to the goal, we may be confident on the boundaries; if not we have to diversify the solution. In this way, we increase the possibility of reaching the goal by increasing the mutation probability.

Proposed tuning mutation operator includes two types of mutation in addition to the mutation types used by SegGen. First, it adds a boundary into the selected individual. Our new mutation operator can change the existing of boundary in the selected individual. We can effectively use the boundary adding by new mutation for getting the accurate result in mutation search. The second type of proposed mutation is based on with a probability Pmut that shifts selected boundary to next sentence on given individual. It shifts the two selected boundaries to next sentences. It simply shifts the two selected "1" bites to the next position in the individual vector. Because, text segments usually have more than one sentence, shifting process could help to find segment boundary in the segmentation progress. Moreover, if selected one is a qualified individual, cost of the shifting boundary process is smaller than recreating qualified individual.

B. Tuning Crossover Operator

Crossover is a genetic algorithm operator that recombines two individuals to produce two new individuals. Crossover operators are common to lead population to converge on a specific point in landscape. First type is a multipoint crossover instead of uniform mono-point crossover operator. We use two common boundary points of selected parents because the generated individuals have to keep existing boundaries on some part of the document to be defined. Due to the keeping existing boundaries in crossover operation and increasing variety, tuning crossover operator process provides a multipoint crossover more specific than ordinary multipoint

crossover. Second type is keeping number of boundary crossover. Due to the fact that when individual size has a greater value, types of gene sequence will increase. So, parent individuals that selected for crossover operator have similar number of boundary give a clue about similarity of two individuals. Thus, relying on similar number of boundary indicates a chance of reach the goal individual.

C. Tuning the Fitness Function

On a given segmentation, the similarity measures between sentences have to be changed to give different weights to sentences depending on their proximities with a boundary but this has to take place during the ongoing process. At the beginning of the process we could only have a low confidence on the position of the boundaries and so have no reason to be treated differently from other sentences in the calculus of similarity, with regard to our study, the calculus of cohesion of segments and dissimilarity between adjacent segments. On subsequent processes, quality of populations increases and we may reasonably think that boundaries are roughly in their final position. So we can be more confident and take this into account in the calculus of similarity. The segmentation in the current population is of a better quality as population evolves but, there is no reason to think that boundaries a more or less in their final position. So since this is near the boundaries that thematic changes occurs, cohesion has to be measured with less or no influences of them. Therefore, the idea of tuning the fitness function came into view. We gave different importance value to sentences depends on their positions. We figured out that near the boundaries and adjacent sentences of the near boundaries have logarithmic importance values (negative values). Thus, we created weighted factor individuals. Due to thematic changes occurs near the boundaries, weighted factor evaluation process provides that these boundary points have less importance on the calculus of cohesion of segments and dissimilarity between adjacent segments.

D. Extraction of Solution

Extraction of results process requires a few additional processes. When the algorithm meets the stop criterion, the external archive contains more than one potential segmentation and we have to extract best segmentation from this potential result set. SegGen uses a linear aggregation function of internal cohesion of segments $C(\vec{x}) \in [0,1]$ and dissimilarity between adjacent segments $D(\vec{x}) \in [0,1]$ [1]:

$$Agg(\vec{x}) = C(\vec{x}) + \alpha \times D(\vec{x}) \quad (2)$$

The coefficient α weights the second objective compared to the first in (2). We used aggregation method of SegGen, thus we extracted best aggregation score of individual from potential result set. In the extraction process, we consider aggregation evaluation in experimental studies of SegGen and α is obtained around 5. Then, we select aggregation score greater than 4.9. After this filter process we choose best score of filtered result set. On the other hand, aggregation score of weighted fitness function does not provide sufficient selectivity in the extraction process. After get a number of results, we observed that there is a relationship between aggregation score of basic fitness function and aggregation score of weighted fitness function. When the aggregation score of basic individual is higher, aggregation score of weighted individual is smaller than others. Since the existing of relationship between basic and weighted aggregation score, we obtained the selection value that the difference between two aggregation score is less than 0.79.

IV. Experimental Results

We used test texts, which consist of articles from the Associated Press published all the year around 1989 [14]. We concatenated sample articles which have various topics, selected from set of 350 documents. We created two corpora T1(30, 2) and T2(30,2) in order to use in experimental process are composed by T(ns,nb) that ns is number of sentences and nb is average boundaries. We used a criterion *WindowDiff* [15] that is a metric using in text segmentations, as an evaluation metric. It considers the number of boundaries between two sentences separated from a distance k, as shown in formula,

$$Windiff(hyp, ref) = \frac{1}{N-k} \sum_{i=0}^{N-k} (|b(ref_i, ref_{i+k}) - b(hyp_i, hyp_{i+k})|) \quad (3)$$

$b(x_i, x_j)$ is the number of boundaries between i and j in a segmented text x which consists of N sentences, ref points to the segmentation of reference and hyp the one found with the method to evaluate.

A. Experimental Process

Two different groups are used in experimental process. First group consists of five versions of algorithm. These are:

- *Basic*: The basic version of the algorithm.
- *M1*: We previously mentioned about adding two types mutations into algorithm. This version comprises of applied two new mutations into basic version.
- *M2*: This version consists of changing mutation probability besides two new mutation types.
- *C*: The C version has tuned crossover operator.
- *M2C*: The version composes applying combination of new mutation types, changeable mutation probability and crossover.

The second group includes two versions of weighted fitness function changes besides basic type. These are:

- *Basic*: The basic version of the algorithm.
- *Weighted*: Weighted fitness function is applied in this version.
- *M2C-Weighted*: Weighted fitness function is applied besides combination of new mutation types, changeable mutation probability and crossover.

Due to depending on random parameters by genetic operators of SegGen, each version of algorithm executes 10 times on each corpus, and then best results are extracted from results.

B. Results

TABLE I. BASIC SEGGEN AND TUNING GENETIC OPERATORS

Windiff	***Basic***	***M1***	***M2***	***C***	***M2C***
T1(30,2)	31.7	30.9	21.7	31.8	27.4
T2(30,2)	28.2	37	29	20.3	20.1

TABLE II. BASIC SEGGEN AND WEIGHTED VALUE

Windiff	***Basic***	***Weighted***	***M2C-Weighted***
T1(30,2)	31.7	26	33
T2(30,2)	28.2	22	32.1

First results of this experimental study of the algorithm obtained on the evaluation corpora are promising, as shown in Table II and I. Our point of view the lower values are better, because Windiff indicates difference between reference segmentation and the method to evaluate. Even if the suiting of the parameters of the algorithm currently builds upon empirical values, in the Table I, tuned genetic operator versions of the algorithm results seem better than results of basic version of the algorithm. Especially, combination of all proposed tuning approaches is better than single versions. Regarding the results in Table II, calculation of weighted value of sentences that in accordance with their position in the whole text, also is promising. Using weighted value method with tuned genetic operators will be better, because these results are first empirical results and tuning of genetic operator process uses random parameters. But it shows that the approach still needs some improvement such as combination of tuning genetic operators and weighted value of sentences.

V. CONCLUSION

Automatic text segmentation identifies the most important thematic breaks by setting the boundaries in a document guided by given criteria, such as the internal cohesion of segments and the dissimilarity between adjacent segments is maximum. Contrary to most of existing algorithms that create boundaries sequentially and set the boundaries between segments on local criteria, SegGen algorithm permits to take a decision since all the boundaries between potential segments are set at the same time rendering, this provides a global view on the text. This paper presents the first results of new improvements in the approach in SegGen. The first improvement builds on modifying the parameters and operators of the genetic algorithm along with the increasing quality of the generated population through the generations. The other improvement is also to consider the increasing of quality of the population as the process evolves with taking into account the nature of the coding of individuals, which in this case are segmentation instances, represented by binary vectors corresponding to the positions of the boundaries of the segmentations. Even though, the parameters of the algorithm in first results rests upon empirical values, first results are promising and we are convinced they will be better by automatically fixing the values of the various parameters in using a kind of learning method, so the algorithm will provide that new improving approach of SegGen instinctively fixes the values of the various parameters instead of the empirical guess we have done in the current state of this research.

REFERENCES

[1] S. Lamprier, T. Amghar, B. Levrat and F. Saubion. “SegGen: a genetic algorithm for linear text segmentation”. In *Proc. of the 20th International Joint conference on Artifical Intelligence,* 2007, pp. 1647-1652.

[2] E. Zitzler. “Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications.” Phd thesis, Swiss Federal Institute of Technology Zurich, Dec. 1999.

[3] G. Salton, A. Singhal, C. Buckley and M. Mitra. “Automatic text decomposition using text segments and text themes.” *Hypertext’96*, ACM, 1996, pp. 53-56.

[4] J.P. Callan. “Passage-level evidence in document retrieval.” In Bruce W. Croft and Cornelius J. Van Rijsbergen, editors, *Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Spring-Verlag, pp. 302 – 310, Dublin, Ireland, Jul. 1994.

[5] M. A. Hearst, C. Plaunt. “Subtopic structuring for full-length document access.” In *Proceedings of the Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 56–68. Association for Computing Machinery, 1993.

[6] D. McDonald, H. Chen. “Using sentence-selection heuristics to rank text segments in txtractor.” *JCDL’02*, ACM Press, pp. 28-35, 2002.

[7] Y.Hamadi, E.Monfroy, and F.Saubion “What is Autonomous Search?”, TechReport n° MSR-TR-2008-80, Microsoft Research, 2008.

[8] M. A. Hearst. “Texttilling: segmenting text into multi-paragraph subtopic passages.” *Comp. Ling.*, vol. 23(1), pp. 33-64, 1997.

[9] F.Y.Y. Choi. “Advances in domain independent linear text segmentation.” *Proc. of the ACL*, Morgan Kaufmann Publishers Inc., pp. 26-33, 2000.

[10] J.C. Reynar. “Topic Segmentation: Algorithms and Applications.” Phd thesis, University of Pennsylvania, Seattle, WA, 2000.

[11] M. Kan, J. Klavans and K. McKeown. “Linear segmentation and segment significance.” *6th Workshop on Very Large Corpora (WVLC-98)*, ACL SIGDAT, pp. 197-205, 1998.

[12] S. Lamprier, T. Amghar, B. Levrat and F. Saubion. “Toward a more global and coherent segmentation of texts.” *Applied Artificial Intelligence*, vol. 22(3), pp. 208-234, Mar 2008.

[13] B. Coppin, Artificial Intelligence Illuminated, 1st ed., Sudbury MA: Jones and Bartlett Publishers, 2004, pp. 126.

[14] D. Harman. “Overview of the first trec conference.” In *SIGIR’93*, ACM Press, pp. 36-47, 1993.

[15] L. Pevzner, M. A. Hearst. “A critique and improvement of an evaluation metric for text segmentation.” *Comp. Ling.*, vol. 28(1), pp. 19-36, 2002.