Unsupervised Extractive Multi-Document Text Summarization

Verónica Neri Mendoza, Yulia Ledeneva [0000-0003-0766-542X] and René Arnulfo García-Hernández [0000-0001-7941-377X]

Autonomous University of the State of Mexico
Instituto Literario No. 100, 50000, Toluca, Mexico
veronica.nerimendoz@gmail.com, yledeneva@yahoo.com
renearnulfo@hotmail.com

Abstract. Multi-Document Text Summarization (MDTS) consists of generating an abstract from a group of two or more number of documents that represent only the most important information of all documents. Generally, the objective is to obtain the main idea of several documents on the same topic. In this paper, we propose a new MDTS method based on a Genetic Algorithm (GA). The fitness function is calculated considering two text features: sentence position and coverage. We propose the binary coding representation, selection, crossover and mutation operators to improve the state-of-the-art results. We test the proposed method on DUC02 data set, specifically, on Extractive Multi-Document Text Summarization (EMDST) task demonstrating the improvement over the state-of-art methods. Two different tasks for each of the 59 collection of documents (in total 567 documents) are tested. In addition, we test different configurations of the most used methodology to generate EMDST summaries. Moreover, different heuristics such as topline, baseline, baseline-random and lead baseline are calculated.

Keywords: Extractive Multi-Document Text Summarization (EMDTS), Genetic Algorithm, Heuristics.

1 Introduction

The extensive use of Internet has caused the enormous growth in the usage of digital information. Currently, there are a great variety of users of online information services with a huge amount of unstructured digital information [10, 18]. The user accesses the information through queries, but the precision is always an issue due to the information overload. One way to resolve this issue is by generating summaries [7, 12].

The general process of summarization consists of rewriting the full text into a brief version [20]. Automatic Text Summarization (ATS) is a task of Natural Language Processing (NLP). ATS consists in selecting the most important units which could be paragraphs, sentences, part of sentences or keywords from a document or collection of documents using the state-of-the-arts methods or commercial systems.

ATS methods can be abstractive or extractive. In the abstractive text summarization, the summaries are composed from fusing and generating new text that describes the most important facts [11]. In the extractive text summarization, the sentences or other parts of a text are extracted and concatenated to compose a summary [14, 18].

Depending on the number of documents, summarization techniques can be classified in two tasks: Single-Document Text Summarization (SDTS) and MDTS. The main goal of the MDTS is to allow to the users to have an overview about the topics and important information that exists in collection of documents within relatively a short time [2, 4, 25]. The MDTS has gained interest since mid-1990s [5], starting with the development of evaluation programs such as Document Understanding Conferences (DUC) [26] and Text Analysis Conferences (TAC) [29].

In this paper, we consider the methodology for building the final summary that considers all sentences of all documents [3, 18, 22, 32]. In this paper, a new MDTS method based on a GA is proposed.

The organization of the paper is as follows: The Section 2 The methodologies used in MDTS. The section 3 Explained the method proposed, The Section 4, shows experimental configuration and results. The section 4 showed the comparison with other methods of state-of-the-art, and heuristics. Finally, in the section 5 the conclusions are presented.

2 Multi-Document Text Summarization Methodologies

In this paper, we consider the most frequently used methodologies that we divide in two groups of works: methodology without considering all sentences [36, 43, 44, 45] and methodology with considering all sentences [1, 6, 20, 29, 34, 35]. The first methodology consists in building the individual summary for each document, and then construct the final summary. The second methodology consists in join all documents of collection in only one document, and then builds only one final summary. In this section, we describe these two methodologies and explain why we apply the second methodology in this paper.

2.1 Methodology without considering all sentences

The first methodology uses so called “meta” summarization procedure for generating multi-document text summaries [24], described as follows:

1. Composing single summary:
   In the first step, for each document of the collection of documents the relevant sentences are independently detected, in other words, relevant sentences are detected locally. The result of this step is a collection of individual extractive summaries.
2. Composing multiple-document summary:

In the second step, each summary is merged composing “meta” summary, and summarized through the same or a different algorithm used in the previous step [30, 31].

This methodology is represented in the figure 1. Is showed that each document of the collection of documents the relevant sentences are independently detected.

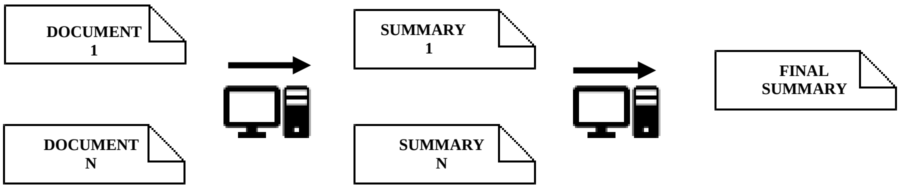

Figure 1. Methodology without considering all sentences [30, 31].

The hypothesis of this methodology was that the final multi-document summaries would be of higher quality since only relevant information is considered for MDTS. However, this methodology does not consider all the sentences of the collection, so cannot reach the upper bound.

2.2 Methodology considering all the sentences

This methodology consists in following steps:

1. Combining the all documents from a given collection of documents:
   In the first step, a new single document is created containing all the documents from a given collection of documents.
2. Composing a single summary:
   In the second step, a summary of this new single document is generated.

This methodology is represented in the figure 2. Where is merging the all documents from a given collection.

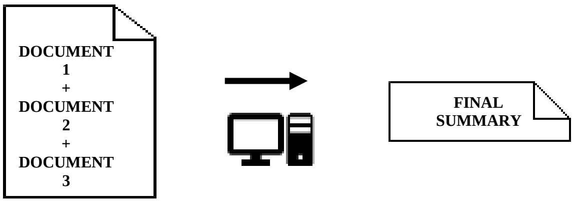

Figure 2. Methodology considering all the summaries [1, 9, 23].

That is, in a new single document, the set of all the sentences that the document collection contains is represented as $D = \{S_1, S_2, \ldots, S_n\}$, where S corresponds to the i sentence of the document collection and n is the total number of sentences in this collection. Likewise, a sentence is represented by the set $S_i = \{t_{i1}, t_{i2}, \ldots, t_{ik}, \ldots, t_{io}\}$, where t_{ik}, is the $k-th$ term of the sentence S_i and o is the total number of terms in the sentence [1, 9, 23].

In this paper, we use the second methodology because we can consider all sentences for the final summary. The most recent state-of-the-art methods also use this methodology.

3 Proposed Method

3.1 Pre-processing

The proposed method consists of three steps. In the first step, the documents of the collection were chronologically ordered, then the original text is adapting to the entry of the format of the GA, where the original text is separated in sentences. Also, the text pre-processing is applied to the collection of documents. Firstly, the text was divided into words separated by commas, then some tags were placed in the text to be able to differentiate quantities, emails, among others, and finally the lexical analysis is carried out [8, 21].

3.2 Text model

The goal of text modeling is to predict the probability of natural word sequences. It assigns high probability on word sequences that occur, and low probability on word sequences that never occur. The simplest and most successful form for text modeling is the n-gram model. n-gram is defined as a subsequence of consecutive elements in a given sequence [15, 21].

3.3 Genetic algorithm

The basic configuration of GA is defined as follows [6]: the initial population is randomly generated, while the population of other generations are generated from some selection/reproduction procedure. The search process terminates when a termination criterion is met. Otherwise a new generation will be produced, and the search process continues. The termination criterion can be selected as a maximum number of generations, or the convergence of the genotypes of the individuals. Genetic operators are constructed according to the problem to be solved, so the crossover operator has been applied to the generation of summaries.

Encoding. The binary encoding is used for each individual, where each sentence of the document constitutes a gene. The values 1 and 0 determine if the sentence will appear or no in the final summary. The initial population is randomly generated [8, 21].

Selection Operator. *Roulette* selects individuals from a population according to their aptitude, and is intended to select stronger individuals (with greater value in the fitness function) [21].

Crossover Operator. This operator has been used in [8]. It was designed for ATS, where each individual represents a selection of sentences. The process of cross over is randomly select parents, only those with genes with a value of 1, and this value is assigned to the new individual. Genes with a value of 1 in both parents will be more likely to be chosen. To meet the condition of the summary, a gene is selected to be part of a new individual, the number of words is counted [21].

Mutation Operator. This operator performs the mutation according to a certain probability as described in [8, 21].

Stop Condition. The stop condition that was applied for the term of the GA is the maximum number of generations. For the execution of the GA, consideration must be given to the number of words that the summary must have. In this case, the lengths of 10, 50, 100 and 200 words were used.

The number of individuals and the number of generations are automatically calculated by the GA through the equations 1 and 2 respectively. The number of individuals is determined by the number of sentences that the document contains by means of the following equation [21]:

$$Number\ individuals = Number\ Sentences * 2 \tag{1}$$

The number of generations is calculated trough the following equation:

$$Number\ Generations\ =\ 4 * 15\ *\ Number\ Sentences \tag{2}$$

Fitness Function. The fitness function was used in the method [8, 21]. In this fitness function are evaluated two features, position sentences and coverage. The main idea is that if all the sentences (see the equation 3) had the same importance, it is could draw a line with the points that make up those coordinates as it is showed in equation 4.

$$\{X_1, X_2, X_3, \ldots X_n\} \tag{3}$$

$$\{(X_1, y), (X_2, y), (X_3, y), \ldots (X_n, y)\} \tag{4}$$

The idea for assigning more importance for the first sentences, would be consider the first sentence with the importance X_n, the second with importance $X_n - 1$.

Since the placement of the line indicates its importance, the midpoint of that line can be used to determine the slope of the line; thus, softening the importance of sentences. This would allow us to know how important a sentence is with respect to the following. For this can use the general equation of the slope of the line.

For a text with n sentences, if the sentence i is selected for the summary then its relevance is defined as $t(i - x) + x,\ where\ x = 1 +\ (n - 1)/2$ and t is the slope to be discovered. With the objective to normalize the measurement of the position of the sentence ($Sentence\ Importance$), the importance of the first k sentences is calculated, where k is the number of selected sentences. Then the formula to calculate the importance of the first sentences would be as follows:

$$Sentence\ importance = \frac{\sum_{|c_i|=1}^{n} t(i-x)+x}{\sum_{j=1}^{k} t(j-x)+1}, x = 1 + \frac{(n-1)}{2} \tag{5}$$

However, it is not the only value by which the GA should be governed since it would try to obtain only the first sentences. It is also necessary to evaluate that the summary has different ideas, that is, it is not repetitive, but at the same time it has important words ($Precision - Recall$). To measure both things the fitness function makes the summation of the frequencies of the n-grams that the summary weigh how significant are the n-grams obtained is the same but considering the original text, in this case only the most frequent n-grams according to the number of minimum words. This weighing are Precision and Recall. Precision defines as a sum of the frequencies of the n-grams consider the original text, expressed as follows:

$$\sum Original\ text\ frequency \tag{6}$$

Recall defines as a sum of the frequencies of the different n-grams of summary:

$$\sum Frequency\ Summary \tag{7}$$

Therefore, the formula for obtaining Precision-Recall is:

$$Presicion - Recall = \frac{\sum Original\ text\ frequency}{\sum Frequency\ Summary} \tag{8}$$

Finally, to obtain the value of the fitness function, the following formula is applied, which is multiplied by 1000.

$$FA\ =\ Presicion\_Recall * Sentence\ Importance\ * 1000 \tag{9}$$

4 Experimentation and Results

We test the proposed method based on the MDTS using the dataset provided in DUC [26]. Traditionally, text summarization evaluation involves human judgments of different quality metrics, for example, coherence, conciseness, grammaticality, readability, and content [19]. We use ROUGE[1] to evaluate the proposed method, which is widely applied by DUC for performance evaluation [16]. It measures the performance of a summary by counting the unit overlaps between the candidate summary and a set of reference summaries.

4.1 Dataset

In order to empirically evaluate the summarization results, DUC02 dataset is used which is benchmark data set of DUC for automatic summarization evaluation. Table 1 gives a brief description of DUC02 [17].

Table 1. Description of DUC02 [17].

Features	Description
Number of documents	567
Number of collection of documents	59
Documents in each cluster	From 5 to 14
Summary length	200 and 400words

4.2 Configuration of experiments

In each experiment, we followed the standard sequence of steps explained in the Section 3. The Table 2 presents the best obtained result of the proposed method with different slop value, for two different summary lengths, and selection operator *(Roulette*). The sentence selection considered parameter is *k-best+first* which consists in selecting the first sentences of the text, until the desired size of the summary is reached [13, 14].

[1] ROUGE (Recall-Oriented Understudy for Gisting Evaluation), toolkit (version 1.5.5.)

Table 2 presents the best obtained result of the proposed method with different slop value, for two summary lengths.

Table 2. Results with several parameters of proposed method.

Summary Length	Value of Slope	Results		
		Recall	Precision	F-Measure
200 words	-0.73	0.483	0.476	0.479
200 words	-0.72	0.488	0.480	0.484
200 words	-0.71	0.479	0.472	0.476
200 words	-0.70	0.482	0.475	0.479
200 words	-0.69	0.479	0.472	0.475
400 words	-0.73	0.565	0.555	0.559
400 words	-0.72	0.572	0.561	0.566
400 words	-0.71	0.549	0.538	0.543
400 words	-0.70	0.566	0.555	0.560
400 words	-0.69	0.569	0.559	0.563

4.3 Comparison to the-state-of-the-art methods

In this section, we describe and then compare the proposed method to the described methods, heuristics and commercial tools. In this paper, we consider the state-of-the-art methods that use the same methodology described in the section 2.2.

4.4 Description of the state-of-the-art methods

- WFS-NMF [33]**:** It is extends of Document clustering based on nonnegative matrix factorization model and provides a good framework for weighting different terms and documents.

- BSTM [32]: Bayesian Sentence-based Topic Models (BSTM) explicitly models the probability distributions of selecting sentences given topics and provides a principled way for the summarization task.

LexRank [32]: LexRank computes sentence importance based on the concept of centrality in a graph representation of sentences. In this model, a connectivity matrix based on intra sentence cosine similarity is used as the adjacency matrix of the graph representation of sentences.

NMF [68]**:** Considers a selection of theorical and empirical features on a document-sentence matrix, and selects the sentences associated with the highest weights to form summaries

TE + WF [18]: This method applies prior recognition of the textual entailment as a previous step to the words frequency in the summarization process. TE (Textual Entailment) consists of using textual implication in text summarization that has been considered as a useful approach for obtaining a preliminary summary, where the sentences have not associated with any other sentence of document. WF (Word Frequency) The sentences that contains the words with the most frequency from the source document (without stop-words) are considered for the final summary.

4.5 Description heuristics

Topline [28]: It is a 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, since it selects sentences considering one or several gold-standard summaries.

Baseline-first [28]: Take the first sentence in the 1st, 2nd, 3rd, etc. document collection in chronological sequence until you have the target summary size.

Baseline-random [28]: It is the state-of-the-art heuristic that randomly selects sentences to present them as an extractive summary to the user.

Baseline-first-document: Take the first sentences in the 1st document of a document collection, until you have the target summary size. This heuristic is proposed in this work.

Lead Baseline [30, 66]: It is a heuristic that take the first 200 and 400 words in the last document in the collection, where documents are assumed to be chronologically ordered.

4.6 Comparison

Since, any method can be worse than randomly choosing sentences (baseline-random), the advance is recalculated as 0%. The best possible performance, topline, it is considered as 100%. Using baseline-random and topline is possible to recalculate the F-measure results in order to see an advance compared to the worst and the best results. In the tables 3 and 4, the results of F-measure of ROUGE-1, ROUGE-2 and Advance are presented.

For the task of 200, there are 5 unsupervised and 1 supervised method, and 4 heuristics calculated in the state-of-the-art as (topline, baseline-first, lead baseline. We calculate heuristics (topline, baseline-first, baseline-random, baseline-first-document and lead-baseline. In the table 4, we see the results of state-of-art method and heuristics. As topline heuristic shows, an extensive margin exists between the best method and the best possible result to obtain. The difference is 67.10%. Also is observed that none method-of-state-art has managed to overcome the heuristic Baseline-first. In the table 3, is showed the parameters that were used to get the results.

Table 3. Parameters was used

Feature	Description
Selection operator	Roulette
Text representation	Bigrams
Elitism	3
Value of slope	0.72

Table 4. Comparison of results to other methods and heuristics for 200 words.

Type of Method	Method	ROUGE-1	ROUGE-2	Advance (%)
Unsupervised Methods	WFS-NMF [33]	49.900	25.800	30.63%
	BSTM [32]	48.812	24.571	27.64%
	Proposed	48.455	21.765	26.66%
	LexRank [32]	47.963	22.949	25.31%
	NMF [32]	44.587	16.280	16.04%
Supervised Methods	TE + TS [18]	41.811	13.466	8.42%
Heuristics	Topline [27]	75.163	66.512	100%
	Baseline-first [27]	50.726	36.979	32.90%
	Baseline-first-Document	40.500	13.648	1.75
	Baseline-random [27]	38.742	9.528	0%
	Lead Baseline	38.195	11.680	-1.50%

For the task of 400 words summary length, we did not find the state-of-the-art and heuristics to compare. We calculate heuristics (topline, baseline-first, baseline-random, baseline-first-document and lead-baseline). In the table 6, we see the results of all state-of-art method and heuristics. As topline heuristic shows, an extensive margin exists between the best method and the best possible result to obtain. The difference is 65.21%. We hope that this experiment serves as a reference for the future works. And in the table 5, is showed the parameters that were used to get the results.

Table 5. Parameters was used

Feature	Description
Selection operator	Roulette
Text representation	Bigrams
Elitism	3
Value of slope	0.72

Table 6. Comparison of results to other methods, and heuristics for 400 words.

Type of Method	Method	ROUGE-1	ROUGE-2	Advance (%)
Unsupervised Methods	Proposed	56.636	28.679	27.85%
Heuristics	Topline [27]	78.836	63.255	100%
	Baseline-first [27]	58.771	34.772	34.79%
	Baseline-firsti-document	44.437	16.461	-11.79%
	Baseline-random [27]	48.066	15.951	0

Lead Baseline	42.518	14.221	-18.03%

5 Conclusions

In this paper, we proposed the method for EMDTS based on GA. The fitness function was calculated considered sentence position and coverage. We proposed the binary coding representation, selection, crossover and mutation operators two different tasks for each of the 59 collection of documents of DUC02 data set (specifically EMDST task) were tested. We tested different configurations of the most used methodology to generate EMDST summaries. Moreover, different heuristics such as topline, baseline, baseline-random and lead baseline were calculated. Although the method did not overcome the Baseline-First heuristic, the results obtained provide a point of reference for future research. For future work we will use more language-independent features as redundancy reduction, sentence length and similarity with the title [49]. Also, we will consider other text models like sn-grams [50] and MFS [51] [52].

References

1. Alguliev, R.M. et al.: Multiple documents summarization based on evolutionary optimization algorithm. Expert Syst. Appl. 40, 5, 1675–1689 (2013).
2. Bakkar, H. et al.: Multi-document Summarizer. Presented at the (2018).
3. Cao, Ziqiang; Wei, F.L.S.Z.M.: Ranking with Recursive Neural Networks and Its Application to Multi-Document Summarization. 7 (2015).
4. Carbonell, J., Goldstein, J.: The use of MMR, diversity-based reranking for reordering documents and producing summaries. In: SIGIR 98. pp. 335–336 ACM Press, New York, New York, USA (1998).
5. Das, D., Martins, A.F.T.: A Survey on Automatic Text Summarization. (2007).
6. Du, K.L., Swamy, M.N.S.: Search and optimization by metaheuristics: Techniques and algorithms inspired by nature. (2016).
7. Ferreira, R. et al.: A multi-document summarization system based on statistics and linguistic treatment. Expert Syst. Appl. 41, 13, 5780–5787 (2014).
8. García-Hernández, R.A., Ledeneva, Y.: Single Extractive Text Summarization Based on a Genetic Algorithm. 374–383 (2013).
9. Jung, C. et al.: Multi-document summarization using evolutionary multi-objective optimization. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion on - GECCO '17. pp. 31–32 ACM Press, New York, New York, USA (2017).
10. Kaushik, A., Naithani, S.: A Comprehensive Study of Text Mining Approach. (2016).
11. Kumar Bharti, S. et al.: Automatic Keyword Extraction for Text Summarization in Multi-document e-Newspapers Articles. (2017).
12. Ledeneva, Y. et al.: Experimenting with Maximal Frequent Sequences for Multi-Document Summarization. 45, ISSN 1870-4069, 233–244 (2010).

13. Ledeneva, Y. et al.: Terms Derived from Frequent Sequences for Extractive Text Summarization. (2008).

14. Ledeneva, Y.N., García-Hernández, R.A.: Generación automática de resúmenes - Retos, propuestas y experimentos. (2017).

15. Ledeneva, Y.N., Gelbukh, A.: Automatic Language-Independent Detection of Multiword Descriptions for Text Summarization. Instituto Politécnico Nacional (2013).

16. Lin, C.-Y.: ROUGE: A Package for Automatic Evaluation of Summaries. 34, 12, 1213–1220 (2011).

17. Lin, H., Bilmes, J.: Multi-document summarization via budgeted maximization of submodular functions. 9.

18. Lloret, E. et al.: Incorporating Textual Entailment Recognition in Single-and Multi-Document Summarization Systems. (2008).

19. Mani, I.: Automatic Summarization. John Benjamins Publishing Company, Amsterdam (2001).

20. Mani, I., Bloedorn, E.: Multi-document Summarization by Graph Search and Matching. (1997).

21. Matías, M.G.A.: Generación Automática De Resúmenes Usando Algoritmos Genéticos. Universidad Autónoma del Estado de México (2013).

22. Mcdonald, R.: A Study of Global Inference Algorithms in Multi-Document Summarization. (2007).

23. Mendoza, M. et al.: A New Memetic Algorithm for Multi-document Summarization Based on CHC Algorithm and Greedy Search. Presented at the November 16 (2014).

24. Mihalcea, R., Tarau, P.: A Language Independent Algorithm for Single and Multiple Document Summarization. In: Proceedings of IJCNLP 2005, 2nd International Join Conference on Natural Language Processing. pp. 19–24 (2005).

25. Nayeem, M.T., Chali, Y.: Extract with Order for Coherent Multi-Document Summarization. (2017).

26. Over, P., Dang, H.: DUC in context. Inf. Process. Manag. 43, 6, 1506–1520 (2007).

27. Rojas, S.J. et al.: Calculating the significance of automatic extractive text summarization using a genetic algorithm. J. Intell. Fuzzy Syst. 35, 1, 293–304 (2018).

28. Rojas Simón, J. et al.: Calculating the Upper Bounds for Multi-Document Summarization using Genetic Algorithms. Comput. y Sist. 22, 1, (2018).

29. Saggion, H., Poibeau, T.: Automatic Text Summarization: Past, Present and Future. Presented at the (2013).

30. Stein, G.C. et al.: Multi-Document Summarization : Methodologies and Evaluations. 16–18 (2000).

31. Villatoro-Tello, E. et al.: Multi-Document summarization based on locally relevant sentences. In: 8th Mexican International Conference on Artificial Intelligence - Proceedings of the Special Session, MICAI 2009. pp. 87–91 IEEE (2009).

32. Wang, D. et al.: Multi-document summarization using sentence-based topic models. In: ACL and AFNLP. p. 297 (2010).

33. Wang, D. et al.: Weighted Feature Subset Non-Negative Matrix Factorization and its Applications to Document Understanding. IEEE Int. Conf. Data Min. (2010).