2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

Combining Graph Connectivity and Genetic Clustering to Improve Biomedical Summarization

Héctor D. Menéndez Laura Plaza David Camacho

Abstract—**Automatic summarization is emerging as a feasible instrument to help biomedical researchers to access online literature and face information overload. The Natural Language Processing community is actively working toward the development of effective summarization applications; however, automatic summaries are sometimes less informative than the user needs. In this work, our aim is to improve a summarization graph-based process combining genetic clustering with graph connectivity information. In this way, while genetic clustering allows us to identify the different topics that are dealt with in a document, connectivity information (in particular, degree centrality) allows us to asses and exploit the relevance of the different topics. Our automatic summaries are compared with others produced by commercial and research applications, to demonstrate the appropriateness of using this combination of techniques for automatic summarization.**

I. Introduction

Information overload has become a serious problem for researchers in biomedicine-related disciplines [1]. The number of biomedical works that are published is growing exponentially, so that nowadays the PubMed database stores more than 19 million references to journal articles [2]. This information overload undermines scientists daily work, since they are unable to find and read all the relevant literature that is published.

Automatic summarization is a Natural Language Processing task (NLP) that aims to deal with this problem by automatically generating an abbreviate and accurate representation of a document (or set of documents) that serves as an indicative summary of the content of the document(s) [3]. In this way, summaries provide the most important information from the sources in order to help researchers to anticipate the content of the documents before deciding which of them to read further.

Graph-based methods are being increasingly used in biomedical data mining and summarization tasks [4]. Graphs have demonstrated to be very powerful tools for capturing and representing the semantics of texts, especially when dealing with highly specialized documents such as biomedical articles [5], [6]. In this way, the documents are usually represented as graphs of biomedical concepts (the nodes in the graph) and relations among them (the links in the graph). Concepts and relations are extracted from domain ontologies (such as the Unified Medical Language System (UMLS) [7], the Systematized Nomenclature of Medicine - Clinical Terms (SNOMED-CT) [8] or the Medical Subject Headings (MeSH) [9]). The graph structure is exploited by a clustering method to discover (usually based on the notion of degree centrality) the more salient concepts in the graph (the *centroids*), and this is next used to extract the most representative sentences from the original documents and to generate the textual summary [4], [10].

This work is focused on text summarization using an existing research summarization system [11] which applies graph theory to guide the selection of the most representative sentences according to the results of a clustering technique applied to the graph. In our previous work, a Genetic Graph-based Clustering (GGC) algorithm [12] is tested to analyze its performance in this domain. The main goal of this analysis was to evaluate the influence of the clustering technique when applied to the summarization process. While the original technique [11] is based on a degree centrality-based approach where the most connected concepts in the graph are consider as the centroids of the clusters, GGC does not use centroids, instead it separates the clusters according to the continuity of the concept relations, i.e., the connections between the concepts. The present work introduces a new methodology which combines both approaches on a single algorithm. On the one hand, the new algorithm, called Genetic Text Clustering (GTC), considers the data continuity generated by the concept network. On the other hand, the algorithm considers the importance (centrality) of the concepts which belongs to the cluster, guiding the search to find relevant clusters according to the concepts' relevance.

The methodologies are compared by generating summaries of 150 biomedical scientific articles from the BioMed Central full-text corpus for text mining research [13]. The automatic summaries are evaluated using ROUGE [14] metrics, which compare each automatic summary with one or more ideal or model summaries and compute different quality measures. Our results demonstrate the benefit of combining graph-based metrics, such as degree centrality, with genetic algorithms for summarization. The automatic summaries generated using this combination of techniques are, according to ROUGE metrics, significantly better than the ones generated using each technique separately, and also better than those produced by existing commercial and research summarizers.

The rest of the work is structured as follows: Section II introduces some related work on text summarization methods and clustering techniques. Section III describes the graph-

Héctor Menéndez and David Camacho are with the Computer Sciences Department, Escuela Politécnica Superior of Universidad Autónoma de Madrid, Spain (email: {hector.menendez, david.camacho}@uam.es). Laura Plaza is with the IR & NLP Group, Universidad Nacional de Educación a Distancia (UNED). (email: lplaza@lsi.uned.es)

This work has been partly supported by: Spanish Ministry of Science and Education under project TIN2010-19872 and Savier an Airbus Defense & Space project (FUAM-076914 y FUAM-076915).

978-1-4799-1488-3/14/$31.00 ©2014 IEEE

based summarization method. Section IV presents the Genetic Text Clustering algorithm. Section V presents the evaluation methodology. Section VI shows the experimental results. Finally, Section VII provides conclusions and future work.

II. Related Work

Our work takes ideas and techniques from different fields of Artificial Intelligence, including Text Summarization, Graph Clustering and Evolutionary Algorithms. In the next subsections, we review the related work in such fields that is close to our work. We also present some related works that have applied Evolutionary Algorithms to NLP tasks.

A. Text Summarization

Text summarization may be defined as the process of distilling the most important information from a document (or documents) to produce an abridged version for a particular user (or users) and task (or tasks) [15]. There are two main approaches to the problem of automatic summarization: extraction and abstraction. Extractive methods construct the summaries by selecting the most relevant sentences in the original documents, while abstractive ones build an internal representation and use natural language generation (NLG) techniques to write the summaries, so that abstracts may contain novel sentences, unseen in the original sources. Abstractive approaches require complex semantic representation, inference and natural language generation, which have not still reached a mature stage nowadays [16]. For this reason, most works in automatic summarization focus on extractive methods.

Traditional summarization systems include computing some simple heuristic rules to estimate the relevance of sentences, such as the position of the sentence in the document or the presence of some cue words [17], [18], counting the frequency of the words in the document to identify central terms [19], or training different machine learning models to deal with summarization as a classification task [20]. Recently, graph-based methods have attracted the interest of the summarization research community. Graphs allow for a more complete representation of text than traditional vectorial models that reflects the interaction between the different textual and semantic units. Graph-based methods usually represent the documents as graphs, where the nodes correspond to text units (such as words, phrases, sentences or even paragraphs), and the edges represent cohesion relationships between these units, or even similarity measures between them (e.g. the Euclidean distance). Once the graph that represents a document is created, the salient nodes are located in the graph and used to extract the corresponding units for the summary. Two commonly used metrics to identify salient information in this graph-based representation are degree centrality and eigenvector centrality [21], both based on connectedness.

LexRank [22] is the most popular example of a centroid-based method to multi-document summarization. It creates an undirected graph, where the nodes are the sentences (represented by their TF-IDF vectors) and the edges represent the cosine similarity between them. A very similar method is proposed by [23] to perform mono-document summarization. As in LexRank, the nodes represent sentences and the edges represent the similarity between them, measured as a function of their content overlap. Litvak et al. [24] also proposed an approach that uses a graph-based syntactic representation for keyword extraction, which can be used as a first step in summarization.

When dealing with biomedical documents, summarization works usually adapting generic approaches to work with domain-specific knowledge. In this line, [25] adapts the lexical chaining approach [26] to work with concepts from the UMLS. BioSquash [27] is a question-oriented extractive system for biomedical multi-document summarization. It constructs a semantic graph that contains concepts of three types: ontological concepts (general ones from WordNet and specific ones from the UMLS), named entities and noun phrases.

In [28], the authors represent a corpus of documents as a graph, where the nodes are the MeSH descriptors found in the corpus, and the edges represent hypernymy and co-occurrence relations between them. They cluster the MeSH concepts in the corpus to identify sets of documents dealing with the same topic and then generate a summary from each document cluster. Fiszman et al. [29] propose an algorithm that makes use of semantic predications provided by SemRep [30] to interpret biomedical text and on the use of lexical and semantic information from the UMLS to produce abstracts from biomedical scientific articles. This same method is adapted in a later work to summarize drug information in MEDLINE citations [31]. Ling et al. [32] focus on the genomic domain, and present a system that ranks sentences according to three features: the relevance of six gene aspects (such as the DNA sequence), the relevance of the documents where the sentences are taken from, and the position of the sentences in the document.

More recent is the work of Shang et al. [33], where the aim is to combine information retrieval techniques with information extraction methods to generate text summaries of sets of documents describing a certain topic. To do this, they use SemRep to extract relations among UMLS Metathesaurus concepts and a relation-level retrieval method to select the relations more relevant to query concepts. Finally, they extract the most relevant sentences for each topic based on the previous ranking of relations and the location of the sentences in different sections of the document.

B. Genetic and Graph Clustering

The clustering problem can be described as a blind search on a collection of unlabeled data, where the elements with similar features are grouped together in sets. There are three main techniques to deal with the clustering problem [34]: overlapping (or non-exclusive), partitional and hierarchical.

A popular clustering technique is K-means. Given a fixed number of clusters, K-means tries to find a division (or partition) of the dataset [35] based on a set of common features. Other approximation, such as Expectation-Maximization

(EM) [35], uses a variable number of clusters. More modern clustering techniques, such as Spectral Clustering (SC) [36], use a graph representation of the data instances and take advantage of the graph properties to calculate the final clusters.

Graph models are useful for diverse types of data representation. They have become especially popular, being widely applied in the social network area. Graph models can be naturally used in these domains, where each node or vertex can be used to represent an agent, and the edges are used to represent their interactions. Later, algorithms, methods and graph theory have been used to analyze different aspects of the network, such as: structure, behavior, stability or even community evolution inside the graph [37], [38], [39], [40]. During the last years, the use of graph-based methods in Natural Language Processing is also gaining growing recognition. There are a variety of textual structures that can be naturally represented as graphs, e.g. lexical-semantic word nets, dependency trees, co-occurrence graphs and hyperlinked documents, just to name a few.

A complete roadmap of graph clustering methods can be found in [41], where different clustering algorithms are described and compared using different kinds of graphs: weighted, directed, undirected. These methods are: cutting, spectral analysis and degree connectivity, amongst others (a complete analysis of connectivity methods can be found in [42]). This roadmap also provides an overview of computational complexity from a theoretical and experimental point of view of the studied methods.

In the present work, we combine a graph representation of biomedical textual data and a genetic algorithm to define the final clusters for the graph. Genetic algorithms have been traditionally used in optimization problems. The complexity of the algorithm depends on the codification and the operations that are used to reproduce, cross, mutate and select the different individuals (chromosomes) of the population [43]. The algorithm applies a fitness function which guides the search to find the best individual of the population.

Different approximation of genetic codifications to the clustering problem were deeply studied by Hruschka et al. [34]. They show the different codifications, operations and fitness functions applied in several genetic algorithms to solved the clustering problem.

This work is based on a Genetic Graph-based Clustering (GGC)[12] algorithm which is inspired on the Spectral Clustering algorithm (it takes the same similarity graph as a starting point) and improves the robustness of the solution. The algorithm takes part on the summarization process (described in the following Section) where it looks for the best groups of concepts inside the concept document graph (see Section 4).

C. Evolutionary Algorithms in Natural Language Processing

Evolutionary algorithms have been successfully applied to different NLP problems, from grammar induction to machine translation, through parameter optimization and search [44].

Smith and Witten [45], for instance, describe a genetic algorithm for grammar induction. The genotype is a context-free grammar whose fitness is evaluated on the basis of how well it covers a training set of sample strings. Selection is performed in inverse proportion to the grammar's size, while mutation is implemented by randomly choosing one grammar (individual).

Litvak et al. [46] propose a language-independent approach for extractive summarization based on the linear optimization of several sentence ranking measures using a genetic algorithm. An individual here is a vector of the weights of the different sentence ranking measures; and selection retains the best fifth of the individual solutions (i.e., those getting the maximal ROUGE value).

Rodríguez et al. [47] evaluate different implementations of evolutionary algorithms to find the alignment between two sentences for being used in statistical machine translation. Hall and Klein [48] propose a generative phylogenetic model for automatically identifying cognate words from unaligned word lists, given only the known family tree of languages.

In [49], genetic algorithms are also applied to two fundamental NLP applications: tagging, i.e., assignment of lexical categories to words; and parsing, i.e., determination of the syntactic structure of sentences.

III. Summarization Method

We use the summarization system presented in [11], which is briefly explained below and depicted in Figure 1. This system has been specially designed for summarization of biomedical literature. This new work modifies the clustering step of the original model using the GTC algorithm instead (see Section IV).

It consists of the following steps:

1) **Document preprocessing:** In this step, irrelevant sections of the document (i.e., those that do not provide important information for the summary, such as *Competing Interests* or *Acknowledgments*) are removed. Abbreviations and acronyms are detected and expanded, and the *title*, *abstract*, and *body* sections are separated.
2) **Concept recognition:** The text in the document body is mapped to concepts from the UMLS Metathesaurus and semantic types from the UMLS Semantic Network [50], using MetaMap [51]. MetaMap is a software to discover UMLS Metathesaurus concepts in text. MetaMap is invoked using the -y disambiguation option, which implements the Journal Descriptor Indexing methodology [52] and allows MetaMap to solve ambiguous mappings. UMLS concepts belonging to very general semantic types (e.g., *Spatial concept* or *Language*) are ignored.
3) **Document representation:** For each sentence, each UMLS concept is extended with their hypernyms. All the hierarchies for each sentence are then merged to create *a sentence graph*, where the nodes represent domain concepts and the edges represent *is-a* relations between them. Next, the different sentence graphs are merged to build a single *document graph*. In this graph,

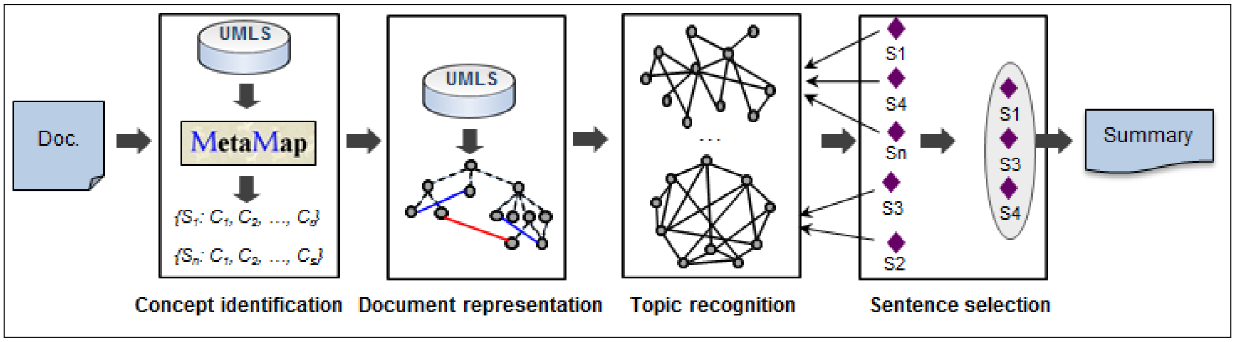

Fig. 1: Architecture of the graph-based summarization system.

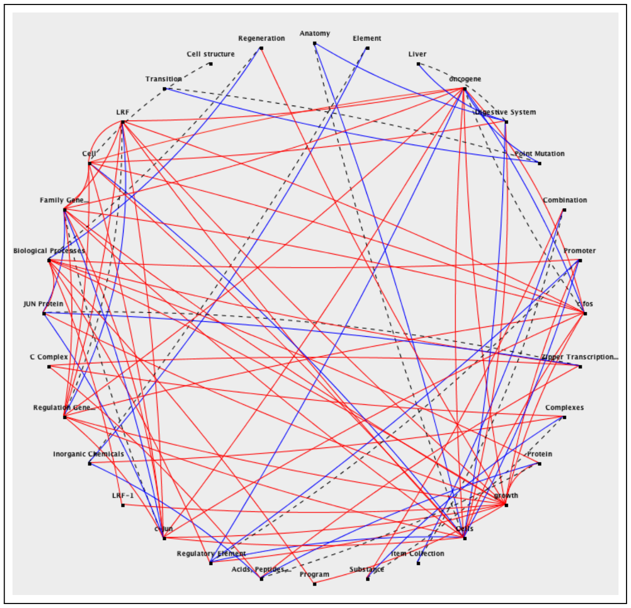

Fig. 2: Example document graph. Dashed lines represent hypernymy relations; red lines represent Metathesaurus relations; and blue lines represent Semantic Network relations.

new edges are added representing the following types of relations between UMLS concepts:

- Relations between semantic types from the UMLS Semantic Network.
- Relations between concepts from the UMLS Metathesaurus.

Next, each edge is assigned a weight in [0,1], as shown in equation 1. The weight of an edge *e* representing an *is-a* relation between two vertices, N_i and N_j (where N_i is a parent of N_j), is calculated as the ratio of the depth of N_i to the depth of N_j from the root of their hierarchy. The weight of an edge representing any other relation (i.e., *associated with* and *related to*) between pairs of leaf vertices is always 1.

$$d(N_i, N_j) = \begin{cases} \frac{depth(N_i)}{depth(N_j)} & is\_a \; relation \\ 1 & otherwise \end{cases} \qquad (1)$$

To illustrate this process, Figure 2 shows the document graph for the following text from [53]:

> *Interactions among LRF-1, JunB, c-Jun, and c-Fos define a regulatory program in the G1 phase of liver regeneration. In regenerating liver, a physiologically normal model of cell growth, LRF-1, JunB, c-Jun, and c-Fos among Jun/Fos/LRF-1 family members are induced posthepatectomy. In liver cells, high levels of c-Fos/c-Jun, c-Fos/JunB, LRF-1/c-Jun, and LRF-1/JunB complexes are present for several hours after the G0/G1 transition, and the relative level of LRF-1/JunB complexes increases during G1. We provide evidence for dramatic differences in promoter-specific activation by LRF-1- and c-Fos-containing complexes. LRF-1 in combination with either Jun protein strongly activates a cyclic AMP response element-containing promoter which c-Fos/Jun does not activate.*

4) **Clustering and topic recognition:** Once the graph has been generated, different clustering techniques can be applied to group the different concepts extracted from the text, with the aim of identifying the different *topics* or *themes* that are dealt with in the text. In this work, a Genetic Text Clustering (GTC) algorithm has been tested (see Section 4).
Regardless of the clustering algorithm that is applied, the *salience* of each node in the graph may be calculated, using the equation 2, as the sum of the weights of the edges that are connected to it. This salience is a measure of the node degree centrality.

$$Salience(N_i) = \sum_{j|N_j} d(N_i, N_j) \qquad (2)$$

5) **Sentence selection:** The last step consists of computing the similarity between each sentence graph (S_i) and each cluster (C_j), and selecting the sentences for the summary based on these similarities. To compute sentence-to-cluster similarity, we add the salience of the common concepts between the sentence graph and the cluster. Finally, a single score for each sentence is calculated, as the sum of its similarity to each cluster adjusted to

the cluster's size (see equation 3). The N sentences with highest scores are then selected for the summary.

$$Scr(S_j) = \sum_{C_i} \frac{sim(C_i, S_j)}{|C_i|} \tag{3}$$

IV. The Genetic Text Clustering (GTC) algorithm

Once the document graph is generated (see step 3 of the summarization method), a clustering algorithm is applied to separate the topics within it. This is done by grouping together in the same topic those concepts that are highly interconnected. The new algorithm combines the degree centrality of the nodes in the graph (as measured by their salience, see Equation 2) and the graph continuity in order to extract the main topics and keep the continuity among them. This algorithm is applied in three steps:

1) **Similarity Graph generation:** a similarity function (usually based on a kernel) is applied to the data instances (i.e., the domain concepts), connecting all the points with each other. It generates the Similarity Graph.
2) **Genetic search:** Giving an initial number of clusters $k_{clusters}$, the GA generates an initial population of possible solutions and evolves them using a fitness function to guide the algorithm to find the best solution. It stops when a good solution is found, or a maximum number of generations is reached.
3) **Clustering association:** The solution with the highest fitness value is chosen as a solution of the algorithm and the data instances are assigned to the $k_{clusters}$ clusters according to the solution chosen.

A. Encoding and Genetic operators

The Encoding is a simple label-based representation [34]. Each individual is a n-dimensional vector (where n is the number of data instances) which has integer values between 1 and the number of clusters. They represent a possible solution. i.e., a cluster selection for each data instance of the dataset.

During the evolution process, the operators can create invalid individuals. These individuals represent solutions where one or more clusters have no elements. In this problem of partitional clustering, these solutions are not valid because the number of clusters is initially given. In this work, no attempt to repair invalid solutions is done. Instead, to avoid the invalid individuals generation problem, they receive a 0 fitness value. The operators used can be briefly summarized as follows:

- **Selection**: The selection process selects a subset of the best individuals. These chromosomes are reproduced and also passed to the next generation. It is called a $(\mu + \lambda)$ selection [43], where μ represents those chromosomes which are chosen, and λ the new chromosomes generated.
- **Crossover**: The crossover operation exchanges strings of numbers between the two chromosomes (both strings have the same length). To reduce the search space, it previously *relabels* those individuals which have different numerical values but represent the same solution (i.e. if there are two chromosomes which represent the same solution but the labels of their clusters are different, these labels are changed in order to maximize the similarity between them).
- **Mutation**: The mutation randomly chooses different chromosomes to change the values of some of their alleles. The new value is a random number between 1 and the number of clusters.

B. The Fitness Function

The fitness function is an hybrid fitness divided in two parts: i) improving the data continuity degree and ii) improving the total salience of the clusters that are generated.

On the one hand, the continuity is guaranteed through a KNN (K-Nearest Neighbour) metric. To control the clusters salience, this metric has been added to the fitness (see Equation 5). It guarantees that the clusters are composed of concepts which are relevant in the graph. The K value for KNN is initially given by the user, nevertheless, in this work we have fixed it to 2 because it is the minimal value to guarantee the continuity and additionally it avoids over-fitting. The Genetic Algorithm maximizes the value of:

$$\frac{TotalKNN \cdot TotalSal}{|C|} \tag{4}$$

where:

$$TotalSal = \sum_{C_\alpha \in C} \frac{\sum_{N_i \in C_\alpha} Sal(N_i)}{max_{C_\alpha}\{\sum_{N_i \in C_\alpha} Sal(N_i)\}} \tag{5}$$

$$TotalKNN = \sum_{x \in C} \frac{|\{y | y \in \Gamma(x) \wedge y \in C_x\}|}{|\Gamma(x)|} \tag{6}$$

In these formulas, $Sal(N_i)$ represents the salience of the node i (see Equation 2), C represents the set of clusters, C_α represents a cluster and $\Gamma(x)$ represents the neighborhood of the element x.

V. Evaluation Methodology

One of the most difficult and costly tasks in text summarization is to evaluate the automatically generated summaries. Deciding whether a summary has a good quality is very subjective, and there is no agreement about the evaluation criteria that should be adopted [54]. Summarization evaluation techniques may be classified into two broad categories:

- **Intrinsic:** directly related to the quality of summarization.
- **Extrinsic:** concerned with the function or task in which the summaries are used, for instance, relevance assessment or reading comprehension.

This work is oriented to intrinsic summarization because the method used is not designed for any specific task. Intrinsic evaluation techniques test the summarization focusing on two desirable properties of the summary [55]:

- **Coherence:** refers to text readability and cohesion.
- **Informativeness:** measures how much information from the source is preserved in the summary.

The evaluation of the summaries may be manual, however, this process requires human judges that need to be expert in the domain of the documents. Human evaluation requires to read both the summaries and the original documents to interpret the texts and extract the salient information, which is very time-consuming. It has also been proven difficult and highly subjective [56]. As a consequence, automatic metrics are usually employed to evaluate the quality of automatic summaries. However, these metrics only measure informativeness [57]. Research in automatic evaluation of coherence is still very preliminary [58].

In this work, the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) package [14] is used to evaluate the informativeness of the automatic summaries. ROUGE is the *de facto* standard for summarization evaluation and compares an automatic summary (called *peer*) with one or more human-made summaries (called *models* or *reference* summaries) and uses the proportion of n-grams in common between the peer and model summaries to estimate the content that is shared between them. The ROUGE metrics produce a value in [0,1], where higher values are preferred, as they indicate a greater content overlap between the peer and model summaries. The following ROUGE metrics are used in this work: ROUGE-2 (R-2) and ROUGE-SU4 (R-SU4). R-2 evaluates bigram occurrence, while R-SU4 evaluates “skip bigrams”, that is, pairs of words having intervening word gaps no larger than four words.

A. Evaluation corpus

To evaluate the automatic summaries, we use a collection of 150 biomedical scientific articles randomly selected from the BioMed Central full-text corpus for text mining research [13]. This corpus contains approximately 85,000 papers of peer-reviewed biomedical research. As stated in [59], the document sample size is large enough to allow significant evaluation results.

As done in previous works [11], [25], the abstracts of the articles are used as gold standard (i.e., as model summaries for the ROUGE evaluation). Such abstracts, written by the authors of the articles, are supposed to summarize the main points of the documents.

B. Experiments

The previous work was focused on comparing different perspectives of the GGC application [60]. Two experiments where carried out in order to compare whether a deep or a relaxed search was necessary to find good solutions (see Table I for the parameter settings). Due to that GGC needs an initial number of clusters, different values of k were compared during the experiments. This work has used the information provided by these experiments and has compared both methods using the same relaxed search for both algorithms (see Table I). Using this information, we compared the results of both algorithms applied to 150 biomedical documents. In order to evaluate the adequacy of our approach, the summaries generated by the summarizer have been also compared to those produced by other summarization systems on the same evaluation collection. The first is a commercial application, *Microsoft AutoSummarize*, which uses a tradition term-frequency based approach. The second is a research application, LexRank [22] (see Section II-A). The rest are two baselines: *Lead* (which chooses the first sentences of the document to generate the summary) and *Random* (which chooses random sentences of the text to generate the summary).

Parameter	GGC & GTC
Breed	50
Crossover Prob.	0.8
Generation	500
Mutation Prob.	0.15
Population	900

TABLE I: Parameter values for both GGC and GTC algorithms.

	ROUGE-2	ROUGE-SU4
GGC k=2	0.261	0.244
GGC k=3	0.269	0.254
GGC k=4	0.273	0.255
GGC k=5	0.270	0.252
GGC k=6	0.264	0.248
GGC k=7	0.267	0.250
GGC k=8	0.264	0.245
GGC k=9	0.253	0.238
GGC Best k	**0.347**	**0.319**
GTC k=2	0.278	0.260
GTC k=3	0.260	0.245
GTC k=4	0.271	0.253
GTC k=5	0.269	0.254
GTC k=6	0.270	0.254
GTC k=7	0.278	0.261
GTC k=8	0.262	0.249
GTC k=9	0.274	0.254
GTC Best k	**0.357**	**0.329**
LexRank	*0.308*	*0.277*
LEAD	0.257	0.265
AutoSummarize	0.245	0.232
Random baseline	0.173	0.230

TABLE II: Results from the application of GGC and GTC algorithms for different values of k and the best value obtained. These results are compared with a commercial application (Microsoft AutoSummarize), a research prototype (LexRank), and two baselines (Lead and Random). The best scores are shown in bold and the second best results in italics.

VI. Results and Discussion

Table I shows the Genetic Parameter selection for both algorithms, these parameters have been selected in order to carry out a relaxed search. Table II presents the results for each algorithm separated by k-values. This table shows the average results for each value of k, for the new model, for the best k and for the other techniques. The best k solution represents the best solution per document comparing the different solutions per k value. The clustering analysis has been carried out 50 times per document and k value. The solution with higher fitness value has been chosen for the evaluation phase.

GGC (see Table II) shows good results compared with all baselines and systems according to ROUGE-2 metric;

however, the results are generally worse when the ROUGE-SU4 metric is applied. LexRank obtains better results than GGC (for k from 2 to 9). Choosing the best result for each value of k and document (see "Best k" in Table II), the results, using both metrics, are the best compared with the rest of algorithms.

GTC (see Table II) shows better results than GGC for both metrics. According to ROUGE-2 metric, the "Best k" value outperforms the results of the GGC algorithm. Also, according to ROUGE-SU4 metric, the value of the different k solutions is closer to the Lead baseline and LexRank values, which are the best (compared with k from 2 to 9), although, in this experiment, the rest of the algorithms are beaten. Choosing again the best result for each value of k (see "Best k" in Table II) the algorithm achieves the best scores in both metrics.

The k value is not always the same for both algorithms (GGC and GTC). This should be a consequence of the algorithm objectives. GGC is focused only on the cluster continuity while GTC is also focused on the centrality degree, thus generating different solutions.

These results show that GGC, which is totally focused on the cluster continuity, obtains good results; however, if the clustering is also focused on the total salience of the concepts, the results improve. GTC concentrates on the salience of the graph, joining concepts which are relevant within the graph. This improves the number of important concepts in the chosen sentences giving more information to the reader about the text; while GGC is focused only on the continuity of the concepts, joining these concepts which are related with each other. These results show that both approaches are mutually beneficial. Other important consideration of these results is the parameter selection. The parameters have been chosen for a relaxed search, i.e., they look for good solutions in the search space using a low number of individuals and population. This also concerns to the algorithm speed, making the algorithm to find a good solution faster than using a deep search.

VII. Conclusions and Future Work

This work has performs a Genetic Graph-based Clustering (GGC) algorithm, using the Graph Salience to increase the topic relevance during the clustering process. The new algorithm, called Genetic Text Clustering (GTC) has shown that the combination of continuity-based measures and degree centrality (salience) obtains better results than the original techniques. The new graph-based summarization process has been evaluated using 150 biomedical documents which has shown that the algorithm also obtains better results than other research and commercial techniques. The following main conclusions have been extracted:

- The topic importance is highly relevance during the summarization process. The combination with continuity-based clustering helps to determine the importance of the sentences in the summary by providing more information about the relevance of the different topics that are dealt with in the text.
- This new methodology beats classical and commercial algorithms.

There are also some issues which might be studied in the future:

- The fitness function should add some other metrics related to other properties of the graph such as the density.
- The invalid solutions generated during the crossover or mutation operations might be repaired.
- Finally, other summarization processes might be compared with the current methodology.

References

[1] B. Humphreys and D. McCutcheon, "Growth patterns in the national library of medicine's serials collection and its index medicus journals, 1966-1985," *Bulletin of the Medical Library Association*, vol. 82, no. 1, pp. 18–24, 1994.

[2] F. Fatehi, L. Gray, and R. Wootton, "How to improve your pubmed/medline searches: 1. background and basic searching," *J Telemed Telecare*, vol. 19, no. 8, pp. 479–486, 2013.

[3] P. Zweigenbaum, D. Demner-Fushman, H. Yu, and K. B. Cohen, "Frontiers of biomedical text mining: current progress," *Briefings in Bioinformatics*, vol. 8, no. 5, pp. 358–375, 2007.

[4] H. Zhang, M. Fiszman, D. Shin, B. Wilkowsk, and T. C. Rindflesch, "Clustering cliques for graph-based summarization of the biomedical research literature," *BMC Bioinformatics*, vol. 14, p. 182, 2013.

[5] M. Bundschus, M. Dejori, M. Stetter, and V. T. H.-P. Kriegel, "Extraction of semantic biomedical relations from text using conditional random fields," *BMC Bioinformatics*, vol. 9, p. 207, 2008.

[6] T. C. Rindflesch, M. Fiszman, and B. Libbus, *Medical informatics: Knowledge Management and Data Mining in Biomedicine*. Sprin, 2005, ch. Semantic interpretation for the biomedical research literature, pp. 399–422.

[7] "Unified medical language system," http://www.nlm.nih.gov/research/umls/.

[8] "Systematized Nomenclature of Medicine - Clinical Terms," http://www.ihtsdo.org/snomed-ct/.

[9] "Medical subject headings," http://www.nlm.nih.gov/mesh/.

[10] L. Plaza, A. Díaz, and P. Gervás, "Concept-graph based biomedical automatic summarization using ontologies," in *TextGraphs '08: Proceedings of the 3rd Textgraphs Workshop on Graph-Based Algorithms for Natural Language Processing*, 2008, pp. 53–56.

[11] ——, "A semantic graph-based approach to biomedical summarisation," *Artif. Intell. Med.*, vol. 53, no. 1, pp. 1–14, Sep. 2011.

[12] H. D. Menéndez and D. Camacho, "A genetic graph-based clustering algorithm," in *Intelligent Data Engineering and Automated Learning - IDEAL 2012*, ser. Lecture Notes in Computer Science, H. Yin, J. Costa, and G. Barreto, Eds. Springer Berlin / Heidelberg, 2012, vol. 7435, pp. 216–225.

[13] "Biomed central corpus," 2012. [Online]. Available: http://www.biomedcentral.com/about/datamining

[14] C. Y. Lin, "Rouge: A Package for Automatic Evaluation of Summaries," in *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, M. F. Moens and S. Szpakowicz, Eds. Barcelona, Spain: Association for Computational Linguistics, 2004, pp. 74–81.

[15] I. Mani and M. Maybury, *Advances in automatic texts ummarization*. The MIT Press, 1999.

[16] S. Ye, T.-S. Chua, M.-Y. Kan, and L. Qiu, "Document concept lattice for text understanding and summarization," *Information Processing and Managemen*, vol. 43, no. 6, pp. 1643–1662, 2007.

[17] H. P. Edmundson, "New Methods in Automatic Extracting," *Journal of the Association for Computing Machinery*, vol. 2, no. 16, pp. 264–285, 1969.

[18] R. Brandow, K. Mitze, and L. Rau, "Automatic condensation of electronic publications by sentence selection," *Information Processing and Management*, vol. 5, no. 31, pp. 675–685, 1995.

[19] H. P. Luhn, "The Automatic Creation of Literature Abstracts," *IBM Journal of Research Development*, vol. 2, no. 2, pp. 159–165, 1958.

[20] J. Kupiec, J. Pedersen, and F. Chen, "A trainable document summarizer," in *Proceedings of the 18th ACM/SIGIR Annual Conference on Research and Development in Information Retrieval*, 1995, pp. 68–73.

[21] S. P. Borgatti, "Centrality and network flow," *Social Networks*, vol. 27, p. 5571, 2005.

[22] G. Erkan and D. R. Radev, "LexRank: Graph-based lexical centrality as salience in text summarization," *Journal of Artificial Intelligence Research (JAIR)*, vol. 22, pp. 457–479, 2004.

[23] R. Mihalcea and P. Tarau, "TextRank - Bringing order into text," in *Proceedings of the Conference EMNLP 2004*, 2004, pp. 404–411.

[24] M. Litvak and M. Last, "Graph-based keyword extraction for single-document summarization," in *Proceedings of the International Conference on Computational Linguistics, Workshop on Multi-source Multilingual Information Extraction and Summarization*, 2008.

[25] L. Reeve, H. Han, and A. Brooks, "The use of domain-specific concepts in biomedical text summarization," *Information Processing and Management*, vol. 43, pp. 1765–1776, 2007.

[26] R. Barzilay and M. Elhadad, "Using lexical chains for text summarization," in *Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, 1997, pp. 10–17.

[27] Z. Shi, G. Melli, Y. Wang, Y. Liu, B. Gu, M. M. Kashani, A. Sarkar, and F. Popowich, "Question answering summarization of multiple biomedical documents," in *Proceedings of the Canadian Conference on Artificial Intelligence*, 2007, pp. 284–295.

[28] I. Yoo, X. Hu, and I.-Y. Song, "A Coherent Graph-Based Semantic Clustering and Summarization Approach for Biomedical Literature and a New Summarization Evaluation Method," *BMC Bioinformatics*, vol. 8, no. 9, p. S4, 2007.

[29] M. Fiszman, T. C. Rindflesch, and H. Kilicoglu, "Abstraction summarization for managing the biomedical research literature," in *Proceedings of the HLT-NAACL Workshop on Computational Lexical Semantics*, 2004, pp. 76–83.

[30] T. Rindflesch and M. Fiszman, "The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text," *Journal of Biomedical Informatics*, vol. 36, pp. 462–477, 2003.

[31] M. Fiszman, T. C. Rindflesch, and H. Kilicoglu, "Summarizing drug information in medline citations," in *Proceedings of AMIA Annu Symp*, 2006, pp. 254–258.

[32] X. Ling, J. Jiang, X. He, Q. Mei, C. Zhai, and B. Schatz, "Generating gene summaries from biomedical literature: A study of semi-structured summarization," *Inf. Process. Manage.*, vol. 43, no. 6, pp. 1777–1791, 2007.

[33] Y. Shang, Y. Li, H. Lin, and Z. Yang, "Enhancing biomedical text summarization using semantic relation extraction," *PLoS one*, vol. 6, no. 8, pp. 1–10, 2011.

[34] E. Hruschka, R. Campello, A. Freitas, and A. de Carvalho, "A survey of evolutionary algorithms for clustering," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 39, no. 2, pp. 133 –155, march 2009.

[35] D. T. Larose, *Discovering Knowledge in Data*. John Wiley & Sons, 2005.

[36] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, Dec. 2007.

[37] M. Dehmer, Ed., *Structural Analysis of Complex Networks*. Birkhäuser Publishing, 2010.

[38] S. Fortunato, V. Latora, and M. Marchiori, "Method to find community structures based on information centrality," *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, vol. 70, no. 5, pp. 056 104:1–13, 2004.

[39] M. C. V. Nascimento and A. C. P. L. F. Carvalho, "A graph clustering algorithm based on a clustering coefficient for weighted graphs," *J. Braz. Comp. Soc.*, vol. 17, no. 1, pp. 19–29, 2011.

[40] D. J. Watts, *Small worlds: The dynamics of networks between order and randomness*. Princeton, NJ: Princeton University Press, 1999.

[41] S. Schaeffer, "Graph clustering," *Computer Science Review*, vol. 1, no. 1, pp. 27–64, 2007.

[42] E. Hartuv and R. Shamir, "A clustering algorithm based on graph connectivity," *Information Processing Letters*, vol. 76, no. 4–6, pp. 175–181, 2000.

[43] D. Coley, *An Introduction to Genetic Algorithms for scientists and engineers*. World Scientific Publishing, 1999.

[44] L. Bungum and B. Gamback, "Evolutionary algorithms in natural language processing," in *Proceedings of the Norwegian Artificial Intelligence Symposium*, 2010, pp. 7–19.

[45] T. Smith and I. Witten, "Learning language using genetic algorithms," *Connectionist, Statistical, and Symbolic Approaches to Learning for Natural Language Processing, vol. 1040 of LNAI*, pp. 132–145, 1995.

[46] M. Litvak, M. Last, and M. 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*, 2010, pp. 927–936.

[47] I. Rodríguez, L. andGarcía Varea and J. Gámez, "On the application of different evolutionary algorithms to the alignment problem in statistical machine translation," *Neurocomputing*, vol. 71, no. 4-6, pp. 755–765, 2008.

[48] D. Hall and D. Klein, "Finding cognate groups using phylogenies," in *Proceedings of the 48th Annual Meeting of the Association for Computationa Linguistics*, 2010, pp. 1030–1039.

[49] L. Araujo, "Symbiosis of evolutionary techniques and statistical natural language processing," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 1, 2004.

[50] S. Nelson, T. Powell, and B. Humphreys, "The unified medical language system (umls) project." *Encyclopedia of library and information science.*, pp. 368–378, 2002.

[51] "Metamap," http://metamap.nlm.nih.gov/.

[52] S. M. Humphrey, W. J. Rogers, H. Kilicoglu, D. Demner-Fushman, and T. C. Rindflesch, "Word sense disambiguation by selecting the best semantic type based on journal descriptor indexing: Preliminary experiment," *J. Am. Soc. Inf. Sci. Technol.*, vol. 57, no. 1, pp. 96–113, Jan. 2006.

[53] J. Hsu, R. Bravo, and R. Taub, "Interactions among lrf-1, junb, c-jun, and c-fos define a regulatory program in the g1 phase of liver regeneration," *Mol Cell Biol*, vol. 12, no. 10, pp. 4654–4665, 1992.

[54] D. R. Radev, S. Teufel, H. Saggion, W. Lam, J. Blitzer, H. Qi, A. Çelebi, D. Liu, and E. Drabek, "Evaluation challenges in large-scale document summarization," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, 2003, pp. 375–382.

[55] I. Mani, "Summarization evaluation: An overview," in *Proceedings of the 2nd NTCIR workshop on research in Chinese and Japonese text retrieval and text summarization. Tokio, Japan: National Institute of Informatics*, 2001.

[56] K. Sparck-Jones and J. R. Galliers, *Evaluating Natural Language Processing Systems: An Analysis and Review*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1996.

[57] S. Tratz and E. Hovy, "Summarization evaluation using transformed basic elements," in *Proceedings of the 1st Text Analysis Conference (TAC)*, 2008.

[58] E. Pitler and A. Nenkova, "Revisiting readability: A unified framework for predicting text quality," in *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. Honolulu, Hawaii: Association for Computational Linguistics, October 2008, pp. 186–195.

[59] C. Y. Lin, "Looking for a few good metrics: Automatic summarization evaluation - how many samples are enough?" in *Proceedings of the NTCIR Workshop 4*, 2004.

[60] H. D. Menéndez, L. Plaza, and D. Camacho, "A genetic graph-based clustering approach to biomedical summarization," in *WIMS*, 2013, p. 10.