Combining a Multi-document Summarization System with a Genetic Algorithm

Aurélien Bossard, Christophe Rodrigues

Combining a Multi-Document Update Summarization System –CBSEAS– with a Genetic Algorithm

Aurélien Bossard, Christophe Rodrigues

Abstract In this paper, we present a combination of a multi-document summarization system with a genetic algorithm. We first introduce a novel approach for automatic summarization. CBSEAS, the system which implements this approach, integrates a new method to detect redundancy at its very core in order to produce summaries with a good informational diversity. However, the evaluation of our system at TAC 2008 —Text Analysis Conference— revealed that system adaptation to a specific domain is fundamental to obtain summaries of an acceptable quality.
The second part of this paper is dedicated to a genetic algorithm which aims to adapt our system to specific domains. We present its evaluation by TAC 2009 on a newswire articles summarization task and show that this optimization is having a great influence on both human and automatic evaluations.

1 Introduction

As more information becomes available online, people confront a new problem: disorientation due to the abundance of information. Document retrieval and text summarization systems can be used to address this problem. While document retrieval engines can help a user to filter out documents, summarization systems can extract and present the essential content of these documents.

Recently, the DUC —Document Understanding Conference— now known as TAC —Text Analysis Conference[1]— evaluation campaigns have proposed to evaluate automatic summarization systems. These competitions have led to recent improvements in summarization and its evaluation.

Aurélien Bossard, Christophe Rodrigues
Laboratoire d'informatique de Paris Nord, CNRS UMR 7030
Université Paris 13, 93430 Villetaneuse, FRANCE
e-mail: firstname.lastname@lipn.univ-paris13.fr

[1] http://nist.tac.gov

In this paper, we present our system, called CBSEAS —Clustering Based Sentence Extractor for Automatic Summarization— and its adaptation to the newswire article summarization task: the use of a genetic algorithm which aims at finding automatically the best suited parameter combination as input of the system.

We first give a quick overview of existing automatic summarization systems. In a second section, we describe our system. We then present our method for parameters optimization, based on a genetic algorithm. In a last section, we discuss the results obtained by our system: its performance on the summarization task, and the influence of the parameters values.

2 Automatic Extractive Summarization Overview

The extractive approaches to automatic summarization consist in selecting the most pertinent sentences or phrases and assemble them together to create a summary. This section gives an overview of this kind of approaches.

2.1 Feature-based approaches

Edmundson [7] defined textual clues which can be used to determine the importance of a sentence. In particular, he set a list of cue words, such as ”hardly” or ”impossible”, using term frequency, sentence position (in a news article for example, the first sentences are the most important) and the number of words occuring in the title. These clues are still used by recent systems, like the one of Kupiec [12].

This kind of approaches does not take into account the overall content of the documents. That is why automatic summarization has evolved into sentence selection using the “centrality” feature: the sentence importance relatively to the overall documents content.

2.2 Centrality-based approaches

Other systems focus on term frequency. Luhn [15] led the way of frequency-based sentence extraction systems. He proposed to build a list of important terms. The importance of a term depends on wether or not its frequency belongs or not to a predefined range. The more a sentence presents words belonging to this list, the more important it is. Radev [19] took advantage of the advances in text statistics by integrating the tf.idf metric to Luhn’s method. The list of important terms, that Radev calls ”centroid”, is composed of the n terms with the highest *tf.idf* –the *tf.idf* metric was introduced by Salton[20]. The sentences are ranked according to their similarity to the centroid. Radev also included a post-processing step to eliminate

redundancy from the summary. He implemented this method in an online multi-document summarizer, MEAD[2] [18].

Radev further improved MEAD using another sentence selection method which he named "Graph-based centrality" [8]. It consists in computing similarity between sentences, and then selecting sentences which are considered as "central" in a graph where nodes are sentences and edges are similarities. The most central sentences are those which have been visited most after a random walk on the graph. This method is inspired by the concept of *prestige* in social network.

The clue-based, term frequency-based and "graph-based centrality" methods are efficient when selecting the sentences which reflect the global content of the documents to be summed up. Such a sentence is called "central". However, these methods are not designed to generate good summaries according to informational diversity. Now, informational diversity is almost as important as centrality when evaluating a summary. Indeed, a summary should contain all the important pieces of information which should not be repeated.

2.3 Dealing with diversity

In multi-document summarization, the risk of extracting two sentences conveying the same information is greater than in a single-document summarization problematic. Moreover, identifying redundancy is a critical task, as information appearing several times in different documents can be qualified as important.

The previously presented systems are dealing with redundancy as a post-processing step. Goldberg [9], assuming that redundancy should be the key concept of multi-document summarization, offered a method to deal with redundancy at the same time as sentence selection. For that purpose, he used a "Markov absorbing chain random walk" on a graph representing the different sentences of the corpus to summarize.

MMR-MD, introduced by Carbonnel in [5], is a measure which needs a passage clustering: all passages considered as synonyms are grouped into the same clusters. MMR-MD takes into account the similarity to a query, coverage of a passage (clusters that it belongs to), content in the passage, similarity to passages already selected for the summary, belonging to a cluster or to a document that has already contributed a passage to the summary.

The problem of this measure lies in the clustering method: in the literature, clustering is generally fulfilled using a threshold. If a passage has a similarity to a cluster centroid higher than a threshold, then it is added to this cluster. This makes it a supervised clustering method.

Considering that diversity is the main issue in multi-document summarization, we want our method to first deal with diversity, grouping sentences in clusters according to the information they convey. The diversity management has to be unsu-

[2] http://www.newsinessence.com/clair/meaddemo/demo.cgi

pervised in order to be adapted to every type of documents. Our method will then apply local centrality-based selection methods to extract one sentence per cluster.

3 CBSEAS: A Clustering-Based Sentence Extractor for Automatic Summarization

We want to specifically manage the multi-document aspect by considering redundancy as the main issue of multi-document summarization. Indeed, we consider the documents to summarize as made up by groups of sentences carrying the same information. In each of these clusters, one sentence can be considered as central. Extracting this sentence, and not another one, in every cluster can lead to summaries in which the risk of redundancy is minimized. The summaries generated with this method may carry a good informational diversity. We here briefly present our system, which is further described in [2].

3.1 Pre-processing

All sentences go through a POS tagger, TreeTagger[3]. While studying news corpora, we identified several categories of news. Only a few of them present some particularities which make them worthwhile for an automatic summarization system. Details are available in [4]. Documents are classified using a keywords/structure clue based categorizer, into four categories:

- Classic news (1: presentation of the event, 2: the premisses, possibly 3: the consequences or projection in the future);
- Chronologies (list of related events ordered chronologically, *cf* Figure 1);
- Comparative news (the state of the article topic in different places or at different times, *cf* Figure 1);
- Enumerative news (an enumeration of facts, recommandations...).

The last three categories are very interesting for an automatic summarizer. In fact, they make up at most 5% of the total number of newswire articles in AQUAINT-2[4]. But, in the training corpus of the "Update Task", they contain 80% of the pertinent information. Moreover, they are written in a concise style, and can be easily inserted into a summary.

[3] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/

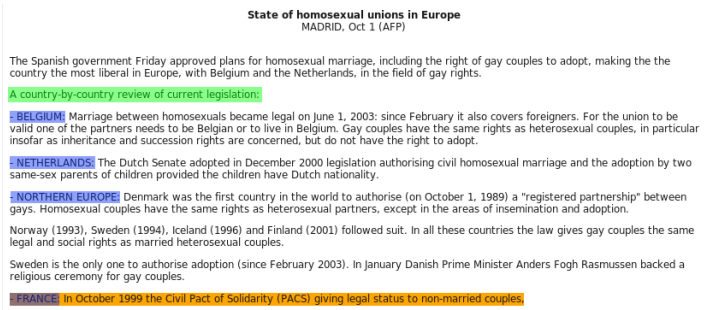
[4] AQUAINT-2 is a corpus built by NIST and composed of 900.000 news articles from different sources (AFP, APW, NYT...)

$$sim(s_1, s_2) = \frac{\sum_{mt} weight(mt) \times fsim(s_1, s_2)}{\frac{fsim(s_1, s_2) + gsim(s_1, s_2)}{\sum_{mt} weight(mt)}} \quad (1)$$

$$fsim(s_1, s_2) = \sum_{n_1 \in s_1} \sum_{n_2 \in s_2} tsim(n_1, n_2) \times \frac{tfidf(n_1) + tfidf(n_2)}{2} \quad (2)$$

$$gsim(s_1, s_2) = card\left((n_1 \in s_1, n_2 \in s_2) \mid tsim(n_1, n_2) < \delta\right) \quad (3)$$

where mt are the morphological types, s_1 and s_2 the sentences, tsim the similarity between two terms using WordNet and the JCn similarity measure [11] and δ a similarity threshold.



State of homosexual unions in Europe
MADRID, Oct 1 (AFP)

The Spanish government Friday approved plans for homosexual marriage, including the right of gay couples to adopt, making the the country the most liberal in Europe, with Belgium and the Netherlands, in the field of gay rights.

A country-by-country review of current legislation:

- BELGIUM: Marriage between homosexuals became legal on June 1, 2003: since February it also covers foreigners. For the union to be valid one of the partners needs to be Belgian or to live in Belgium. Gay couples have the same rights as heterosexual couples, in particular insofar as inheritance and succession rights are concerned, but do not have the right to adopt.

- NETHERLANDS: The Dutch Senate adopted in December 2000 legislation authorising civil homosexual marriage and the adoption by two same-sex parents of children provided the children have Dutch nationality.

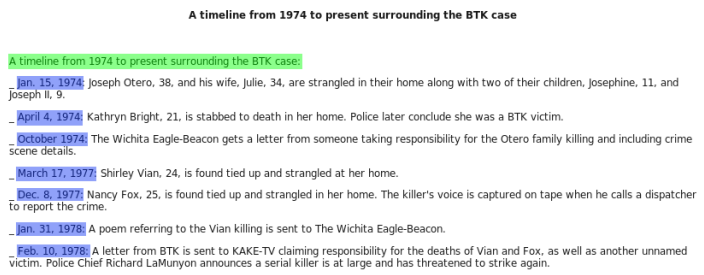
- NORTHERN EUROPE: Denmark was the first country in the world to authorise (on October 1, 1989) a "registered partnership" between gays. Homosexual couples have the same rights as heterosexual partners, except in the areas of insemination and adoption.

Norway (1993), Sweden (1994), Iceland (1996) and Finland (2001) followed suit. In all these countries the law gives gay couples the same legal and social rights as married heterosexual couples.

Sweden is the only one to authorise adoption (since February 2003). In January Danish Prime Minister Anders Fogh Rasmussen backed a religious ceremony for gay couples.

- FRANCE: In October 1999 the Civil Pact of Solidarity (PACS) giving legal status to non-married couples.

(a) A comparative news



A timeline from 1974 to present surrounding the BTK case

A timeline from 1974 to present surrounding the BTK case:

_ Jan. 15, 1974: Joseph Otero, 38, and his wife, Julie, 34, are strangled in their home along with two of their children, Josephine, 11, and Joseph II, 9.

_ April 4, 1974: Kathryn Bright, 21, is stabbed to death in her home. Police later conclude she was a BTK victim.

_ October 1974: The Wichita Eagle-Beacon gets a letter from someone taking responsibility for the Otero family killing and including crime scene details.

_ March 17, 1977: Shirley Vian, 24, is found tied up and strangled at her home.

_ Dec. 8, 1977: Nancy Fox, 25, is found tied up and strangled in her home. The killer's voice is captured on tape when he calls a dispatcher to report the crime.

_ Jan. 31, 1978: A poem referring to the Vian killing is sent to The Wichita Eagle-Beacon.

_ Feb. 10, 1978: A letter from BTK is sent to KAKE-TV claiming responsibility for the deaths of Vian and Fox, as well as another unnamed victim. Police Chief Richard LaMunyon announces a serial killer is at large and has threatened to strike again.

(b) A chronology

Fig. 1 News examples

3.2 Sentence pre-selection

First, our system ranks all the sentences according to their similarity to the documents centroid, composed of the m terms with the highest *tf.idf*. In the case a user query is provided, the sentences are ranked according to their relevance to the query. We then select the best ranked sentences, using an empiric threshold. This method has been changed with the integration of the genetic algorithm, as shown in Sec. 4.

3.3 Sentence clustering

Similarity between sentences is computed using a variant of the "Jaccard" measure, shown in Equations 1, 2 and 3. Other similarity measures exist, such as cosine similarity, but this measure allows us to take into account the similarity between two different terms in the sentence similarity computation. This point is important as linguistic variation could otherwise not be managed.

Once the similarities are computed, we cluster the sentences using fast global k-means (description of the algorithm is in Figure 2) using the similarity matrix.

for all $e_j inE$ *%%Initialize the first cluster with all the elements*
    $C_1 \leftarrow e_j$
for i from 1 to k do
    for j from 1 to i
        center(C_j) $\leftarrow argmax_{e_m} \sum_{e_n \in C_j} sim(e_m, e_n)$
    for all e_j in E
        $e_j \rightarrow C_l | C_l maximizes sim(center(C_l, e_j)$
    add a new cluster: C_i. It initially contains only its
    center, the worst represented element in its cluster.
done

Fig. 2 Fast global k-means algorithm

3.4 Sentence final selection

After this clustering step, we select one sentence per cluster in order to produce a summary that maximizes the informational diversity. The selected sentence has to be central in the document and relevant to the query. The system chooses the sentence that maximizes a weighted sum of four scores :

- Similarity to user query/*centroid*;
- Similarity to cluster center;
- Important sentence score (implemented after TAC 2008 campaign);
- Difference in length between the scored sentence and the desired sentence length.

The "Important sentence score" is the inverse of the sentence position in the document if the sentence is part of a "classic news", or 1 if the sentence is part of the body of a news classified as a chronology, an enumerative news or a comparative news.

3.5 Managing update for TAC "Update Task"

Sometimes, a user wants to know what is new about a topic since the last time he has read news about it. That is why the TAC 2008 and TAC 2009 "Update Task" consisted in summarizing a first document set, then summarizing what is new in a second document set.

CBSEAS –Clustering-Based Sentence Extractor for Automatic Summarization– clusters semantically close sentences. In others terms, it creates different clusters for semantically distant sentences. Our clustering method can also be used to differenciate sentences carrying new pieces of information from sentences carrying already known pieces of information, and so for managing update. In fact, sentences carrying old pieces of information are semantically close from the sentences that a user has already read.

CBSEAS has proven to be efficient at grouping together semantically close sentences and differentiate semantically far ones. In fact, the results obtained by CBSEAS on TAC 2008 Opinion Task are good, as CBSEAS appears at the third place for avoiding redundancy in the summaries [3]. This is another reason for using our clustering method to differentiate update sentences from non-update ones.

Before trying to identify update sentences, we need to modelize the pieces of information that the user requesting the update summary has already read. We can then confront the new documents to this model in order to determine if sentences from these documents carry new pieces of information. So the first step of our algorithm is to cluster the sentences from the documents the user has already read –which we call D_I– into k_I groups, as in Sec. 3.3 for the generation of a standard summary.

The model thus computed –M_I– is then used for the second step of our algorithm, which consists in determining if a sentence from the new documents –D_U– is to be grouped with the sentences from D_I, or to create a new cluster which will only contain update sentences. *Fast global k-means* algorithm, slightly modified, can be used to confront elements to a previously established model in order to determine if these elements can be an integral part of the model. We here describe the second clustering part of our update algorithm.

First, our algorithm selects the sentences from D_U same as for D_I (*cf* Sec. 3.2). Then, it computes the similarities between sentences from D_U with the cluster centers of M_I and between all the sentences from D_U. Then it adds the new sentences to M_I, and iterates *fast global k-means* from the k_I iteration with the following constraints:

- The sentences from D_I can not be moved to another cluster; this is done to preserve the M_I model which encodes the old pieces of information. It also avoids to disturb the semantic range of the new clusters that bear novelty.
- The cluster centers from M_I can not be recomputed; as the semantic range of a cluster depends directly on its center, this prevents the semantic range of M_I clusters from being changed by the integration of new elements from D_U.

In order to favor sentences from the second set of document being part of the update clusters, a negative weight can be assigned to the similarities between sentences belonging to the first document set and sentences belonging to the second.

Once the update clusters have been populated, the update summary is generated by extracting one sentence per update cluster, as in Sec. 3.4.

4 Optimizing CBSEAS parameters

News article summarization differs from scientific article summarization or technical report summarization. When aiming at finding similar sentences in order to detect central sentences in a technical report, a system should not focus on the same markers as for blogs or novel summarization. Dealing with scientific articles, centrality could not be the best indicator of sentence importance. Teufel has shown in [21] that examining the rhetorical status of a sentence —its position in the document structure, if it contains cue phrases...— is a good way to figure out if it should appear in the final summary.

Our participation to both the "Update Task" (*cf* Sec. 3.5) and the "Opinion Task" —Summarizing opinions found in blogs— of TAC 2008 showed us that our system can be competitive; it ranked second on the "Opinion Task", but its poor behavior on the "Update Task" showed that adaptation Splays a crucial role in performing better on this task. For this purpose, we have first implemented a score that takes into account specific news structure traits (*cf* Sec. 3.4), and have chosen to use a learning technique that automatically adapts CBSEAS' weights according to a scoring method.

TAC 2008 campaign provided us a corpus, manual reference summaries, and an automatic evaluation framework: ROUGE[5]. ROUGE is a package of automatic evaluation measures using unigram co-occurrences between summary pairs [13]. When computing ROUGE scores between an automatic summary and one or more manual summary, we can efficiently evaluate the information content of the automatic summary. Also, our system takes fourteen parameters as input:

1. number of sentences desired as output;
2. average desired sentence length ;
3. weights of proper names, (4.) nouns, (5.) adjectives, (6.) adverbs, (7.) verbs and (8.) numbers in the similarity function (*cf* Sec. 3.3);
9. number of pre-selected sentences from the first and the (10.) second document sets ;
11. weight of similarity to cluster center, (12.) important sentence score, (13.) and length difference in the final sentence selection scoring (*cf* Sec. 3.4);
14. reduction of similarities between first document set and second document set sentences (*cf* Sec. 3.4).

We have all it takes for an environment interactive learning method.

[5] http://berouge.com

4.1 Overview of parameters optimization for automatic summarization

In the field of trainable summarizers, systems combine basic features and try to find the best weight combination using an algorithm that adapts weights to maximize a fitness score. Kupiec [12] and Aone [1] used similar features to Edmundson [7] and optimized the weight of every feature using a trainable feature combiner using Bayesian network. MCBA [23] added two scores: a centrality score —intersection of sentence keywords and the other sentences keywords on the union of sentence keywords and the other sentences keywords)— and the similarity to title. The best weight combination is approximated using a genetic algorithm. Osborne used a gradient search method to optimize the feature weights[17].

In a more statistical-oriented approach, the PYTHY system [22] used standard features and different frequency-based features. The search for the best weight combination was based on a dynamic programming solution for the knapsack problem described in [16].

4.2 What type of algorithm?

In our case, we cannot prove the regularity and continuity of a function from the hypothesis space to the summary score. Indeed, the parameters we use are not only weights for linear features combination. Now, function continuity is a pre-required for gradient search methods to work correctly. Moreover, as some parameters operate at different steps of our algorithm and on different aspects of sentence selection, building up a probabilistic model of hypothesis space that takes into account parameters dependencies is too complicated. The number of parameters (14) emphasizes the hugeness of the search space. Consequently, a genetic algorithm seems an appropriate method to learn the best parameters combination.

Genetic algorithms have been introduced by John Holland [10]. Holland aims at using species natural adaptation metaphor in order to automatically realize an optimal adaptation to an environment. The main idea is to generate individuals, and by means of mutation and crossing over selected individuals, to father a new generation of individual that will be more adapted to its environment than the previous one.

4.3 ROUGE-SU4 metric liability

We are using ROUGE-SU4 metric to automatically evaluate the quality of the summaries. We won't describe this metric, but one can find details about it in [13]. The liability of this metric is crucial for the genetic algorithm. During TAC 2008 campaign, three evaluations have been conducted:

- an entirely manual evaluation: assessors had to fill a grid with scores such as non-redundancy, readability, overall responsiveness[6], grammaticality, readability;
- pyramid evaluation [14], which consists in manually comparing the information available in the automatic summaries with the information available in the reference summaries;
- ROUGE evaluation.

Amongst the ten best ranked systems in responsiveness score, only four appeared in the top ten of ROUGE-SU4 scores. However, five out of the six other systems from this top ten ranked between the average and the poorest system in readability. This means that readability has a great influence on a human assessor judging the responsiveness. We noticed that systems ranked low in readability were using rewriting rules or sentence compression methods that make summaries less readable. Here is an extract of a summary created by one of these systems: *"The A380 will take over from the Boeing 747 (...?). The Airbus official said he had not seen any sign (of what?). Airbus says the A380 will produce half (as what?) as the 747. Most airports originally thought to accommodate (...?) the A380. The A380 is designed to carry 555 passengers. The plane's engineers will begin to find out (what?)."*. One can see that this summary, although it obtained good ROUGE scores, is not understandable. The summarization system has removed phrases that are essential for sentences comprehension.

ROUGE-SU4 is a good metric to evaluate different summaries created by extraction systems that do not modify extracted sentences when summarizing documents such as newswire articles, where sentences are all syntactically correct. So this metric is adapted to our optimization problem.

4.4 Our genetic algorithm

4.4.1 The individuals

Each individual is composed of 14 parameters, which are described in Section 4. We empirically set their variation space. The Table 1 shows the space in which they fluctuate.

4.4.2 Individuals selection method

The evaluation of one individual is for us a time costly operation. That is the reason why we have chosen a tournament selection method, which has the advantage to be easily parallelized. For each generation of γ individuals, μ tournaments between λ individuals are organized. The winner of each tournament is selected to be part of

[6] Overall responsiveness is the answer to the question : "How much would you pay for this summary?"

$$\delta_i = \begin{cases} \left\lceil log(val_i - min_i) \times rand(0,1) \right\rceil, & val_i \neq min_i, rand_i(0,1) < lower_i \quad (4) \\ 1, & val_i = min_i, rand_i(0,1) < lower_i \quad (5) \\ \left\lceil log(val_i - max_i) \times rand(0,1) \right\rceil, & val_i \neq max_i, rand_i(0,1) > lower_i \quad (6) \\ 1, & val_i = max_i, rand_i(0,1) > lower_i. \quad (7) \end{cases}$$

where val_i is the value of parameter i,
and

$$lower_i = \frac{val_i - min_i}{max_i - min_i}, \quad (8)$$

with i from 1 to 14.

Table 1 Parameters' variation space

parameter	min	max	step
num. of sentences	1	20	1
av. length	1	20	1
num. of pre-selected sent.	1	200	1
num. of pre-selected sent. update	1	200	1
nouns weight	1	300	1
proper names weight	1	300	1
verbs weight	1	300	1
adjectives weight	1	300	1
adverbs weight	1	300	1
numbers weight	1	300	1
cluster center sim weight	1	300	1
important sent. score weight	1	300	1
length difference score weight	1	300	1
update sim reduction	0	1	0.01

the next generation parents. Another advantage of this method lies in the fact that it preserves diversity because the selected individuals are not forced to be the best ones. This prevents the algorithm from getting stuck in a local minimum.

4.4.3 Mutation operator

As we do not know what parameters are dependent one to another, we want to change several parameters at the same time. In order to avoid a too heavy variation due to the simultaneous mutation of several parameters, we have chosen to limit the variation quantity (δ_i) of a parameter, weakening the probability to obtain a strong variation. We do that by using a logarithmic variation described in Equations 4 and 8.

4.4.4 Creating a new generation

Each generation is composed of 100 individuals. The algorithm organizes twenty tournaments with fifteen randomly selected representatives. This seems to be a good compromise between quick evolution and diversity preservation. Each new generation is composed of the twenty winners, forty individuals created by mutating the winners, and the last forty created by randomly crossing the winners.

4.5 Training and evaluation data

TAC 2008 and 2009 "Update Task" consisted in creating two abstracts for forty-eight pairs of document sets. As computing a summary is time expensive, we decided to limit the training data to nine pairs of document sets. The evaluation data is composed of the forty other pairs of document sets.

5 Evaluation

TAC 2008 campaign has shown that automatic evaluation was still not as trustable as manual evaluation when dealing with summaries [6]. Although automatic evaluation proves to be useful to quickly judge the quality of a summary or to act as a fitness score for a learning algorithm, we cannot entirely rely on automatic evaluation. Our goal is to figure out at what point the optimization of the parameters really improves the quality of the automatic created summaries. We propose here two ways to do this: using ROUGE scores to see if the optimized parameters have led to an enhancement on the evaluation data, and letting an assessor judge if there is a visible improvement of the summaries quality.

We selected the best manually evaluated summarizer from TAC 2008, and our summarizer CBSEAS before and after the optimization. We selected fifteen pairs of document sets, and submitted the results of both of the three systems to an assessor, giving the automatically created summaries random ids, in order to avoid the assessor being able to identify the origin of summaries.

We then asked two questions to the assessor:

- Which one of the three summaries reflects best the documents content? (this summary gets the score 6)
- Compared to the best summary, give a score between 1 and five to the two other ones:
  - 5: the summary is almost as informative as the best one;
  - 4: the summary is a bit less informative than the best one;
  - 3: the summary is less informative than the best one;
  - 2: the summary is really less informative than the best one;
  - 1: no comparison is possible, the best summary overtakes this one.

We participated to TAC 2009 in order to validate that our system is performing better and to evaluate its competitiveness.

6 Results and discussion

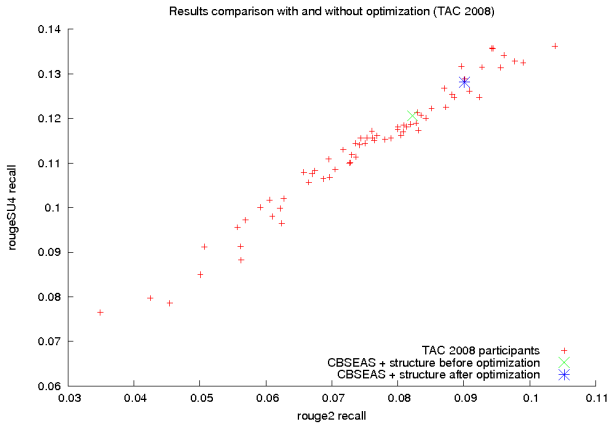


Fig. 3 ROUGE scores comparison of CBSEAS with TAC 2008 other participants

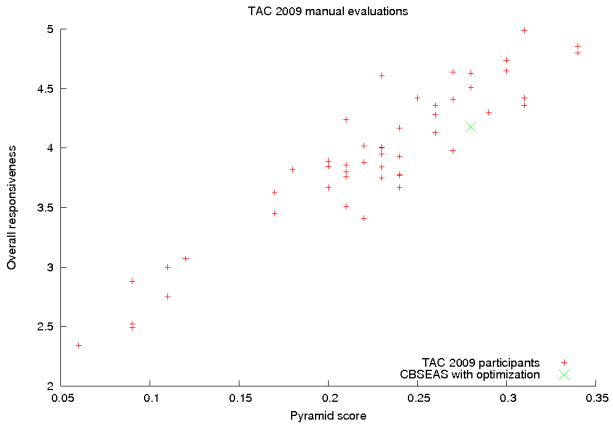


Fig. 4 ROUGE scores comparison of CBSEAS with TAC 2009 other participants

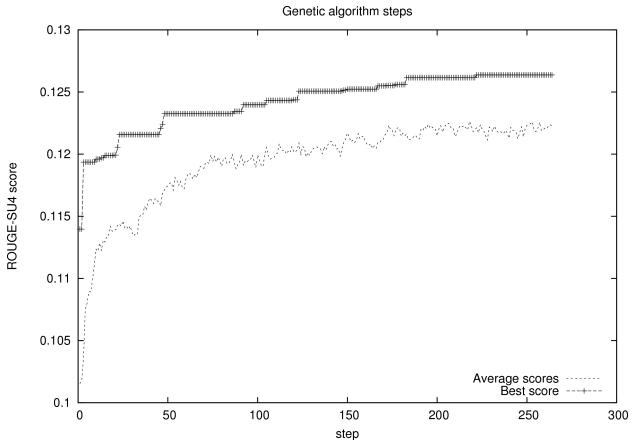


Fig. 5 Average of individual scores, and best individual for each generation

Table 2 Winning set of parameters

parameter	value
num. of sentences	14
av. length	8
num. of pre-selected sent.	47
num. of pre-selected sent. update	83
nouns weight	171
proper names weight	29
verbs weight	207
adjectives weight	270
adverbs weight	12
numbers weight	66
cluster center sim weight	7
important sent. score weight	258
length difference score weight	72
update sim reduction	0.87

The Table 2 shows the combination of features selected by the genetic algorithm after 80 generations. It points out that setting a low weight of the proper names weight has a positive influence on the summary ROUGE scores. Also, the more important types seem to be the common names, adjectives and verbs. Adverbs are having a lesser influence on the summary quality.

The weight of proper names is so small because most of the selected sentences contain the same proper names, due to the fact that pre-selected sentences are close to the user query. This query is indeed most of the time oriented by named entities. So, having proper names playing an important role in sentence similarity computation brings noise to the similarity measure and affects negatively the clustering algorithm. In a more general way, this validates the observation of Aone et al. [1]: decreasing the impact of proper names in the sentence selection method for automatic news summarization increases the quality of the summaries.

Setting the variable "update sim reduction" in a way that strenghtens the similarities between sentences from the first and the second set of documents leads to the generation of higher scored summaries. This means that decreasing the probability that a sentence from the second document set will appear in an update cluster improves the quality of the update management.

It is interesting to note that the feature "similarity to cluster center" gets the lowest weight in the last step of our algorithm. As recent works have proven the pertinence of graph-based methods for automatic summarization, this tends to prove that our similarity score is not adapted to such a feature. Other similarity measures should be reassessed in order to increase the impact of this feature.

Table 3 Manual evaluation

	Best TAC system	CBSEAS w/o optimization	Optimized CBSEAS
Standard summaries			
Number of times winning	9	2	4
non winning summaries average score	4.7	3.9	4.3
Update summaries			
Number of times winning	8	2	5
non winning summaries average score	5	3.7	4.5
Overall scores			
Number of times winning	17	4	9
non winning summaries average score	4.8	3.8	4.4

We observe that manual evaluation presented in Table 3 and automatic evaluation agree: optimizing our parameters for this task has led to an important improvement of the summaries quality, but CBSEAS still does not overtake the best automatic systems of TAC 2008. This has been confirmed by our participation to TAC 2009 and the manual results of this conference, as shown by Fig. 4 (Pyramid and overall responsiveness evaluations). However, the system ranks among the best quarter of all participating systems.

7 Conclusion

In this article, we presented our approach to generic multi-document summarization and update management, and the integration of news articles structure to our system, CBSEAS. We also presented a way to optimize the system we have developed via a genetic algorithm. The results obtained by both manual and automatic evaluations have shown us that the quality of our summaries has greatly improved. The impact of

domain characteristics are important when automatically summarizing documents. The use of a genetic algorithm to optimize the features treatment in our systems has revealed some counter-intuitive observations. Although a human judgment is necessary, we cannot exclude automatic ways to find the best parameters combination for a given task. The results of TAC 2009 also show that our system still needs some improvements to rank among the very best systems. More linguistic methods, such as sentence compression or sentence reranking should be investigated to improve the overall quality of the summaries generated by CBSEAS.

8 Acknowledgement

Special thanks to Thibault Mondary and the GipiLab for having accepted to spend some time evaluating manually our work.

References

1. Aone, C., Okurowski, M.E., Gorlinsky, J.: Trainable, scalable summarization using robust nlp and machine learning. In: Proceedings of the 17th international conference on Computational linguistics, pp. 62–66. Association for Computational Linguistics, Morristown, NJ, USA (1998). DOI http://dx.doi.org/10.3115/980845.980856
2. Bossard, A.: CBSEAS, a new approach to automatic summarization. In: Proceedings of the SIGIR 2009 Conference - Doctoral Consortium. Boston, USA (2009)
3. Bossard, A., Généreux, M., Poibeau, T.: Description of the LIPN System at TAC 2008: Summarizing Information and Opinions. In: Proceedings of the 2008 Text Analysis Conference (TAC 2008) TAC 2008, pp. 282–291. Gaithersburg, United States (2008). URL http://hal.archives-ouvertes.fr/hal-00397010/en/
4. Bossard, A., Poibeau, T.: Integrating document structure to an automatic summarizer. In: RANLP 2009. Borovets, Bulgaria (2009)
5. Carbonell, J., Goldstein, J.: The use of mmr, diversity-based reranking for reordering documents and producing summaries. In: SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference, pp. 335–336. ACM, New York, NY, USA (1998)
6. Dang, H.T., Owczarzak, K.: Overview of the TAC 2008 update summarization task (DRAFT). In: Notebook papers and results of TAC 2008, pp. 10–23. Gaithersburg, Maryland, USA (2008)
7. Edmundson, H.P., Wyllys, R.E.: Automatic abstracting and indexing—survey and recommendations. Commun. ACM **4**(5), 226–234 (1961)
8. Erkan, G., Radev, D.R.: Lexrank: Graph-based centrality as salience in text summarization. Journal of Artificial Intelligence Research (JAIR) (2004)
9. Goldberg, A.: Cs838-1 advanced nlp : Automatic summarization (2007). URL http://www.avglab.com/andrew/
10. Holland, J.H.: Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence. University of Michigan Press (1975)
11. Jiang, J.J., Conrath, D.W.: Semantic similarity based on corpus statistics and lexical taxonomy. In: International Conference Research on Computational Linguistics (ROCLING X), pp. 9008+ (1997). URL http://adsabs.harvard.edu/cgi-bin/nph-bib_query?bibcode=1997cmp.lg....9008J

12. Kupiec, J., Pedersen, J., Chen, F.: A trainable document summarizer. In: SIGIR '95: Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 68–73. ACM, New York, NY, USA (1995). DOI http://doi.acm.org/10.1145/215206.215333
13. Lin, C.Y.: Rouge: a package for automatic evaluation of summaries. In: Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004). Barcelona, Spain (2004)
14. Lin, C.Y., Cao, G., Gao, J., Nie, J.Y.: An information-theoretic approach to automatic evaluation of summaries. In: Proceedings of the main conference on HLTC NACACL, pp. 463–470. Association for Computational Linguistics, Morristown, NJ, USA (2006)
15. Luhn, H.: The automatic creation of literature abstracts. IBM Journal **2**(2), 159–165 (1958)
16. McDonald, R.: A study of global inference algorithms in multi-document summarization. In: G. Amati, C. Carpineto, G. Romano (eds.) ECIR, *Lecture Notes in Computer Science*, vol. 4425, pp. 557–564. Springer (2007)
17. Osborne, M.: Using maximum entropy for sentence extraction. In: Proceedings of the ACL-02 Workshop on Automatic Summarization, pp. 1–8. Association for Computational Linguistics, Morristown, NJ, USA (2002). DOI http://dx.doi.org/10.3115/1118162.1118163
18. Radev, D., Allison, T., Blair-Goldensohn, S., Blitzer, J., Çelebi, A., Dimitrov, S., Drabek, E., Hakim, A., Lam, W., Liu, D., Otterbacher, J., Qi, H., Saggion, H., Teufel, S., Topper, M., Winkel, A., Zhu, Z.: MEAD - a platform for multidocument multilingual text summarization. In: Proceedings of LREC 2004. Lisbon, Portugal (2004)
19. Radev, D., Winkel, A.: Multi document centroid-based text summarization. In: In ACL 2002 (2002)
20. Salton, G., McGill, M.: Introduction to modern information retrieval (1983)
21. Teufel, S., Moens, M.: Summarizing scientific articles - experiments with relevance and rhetorical status. Computational Linguistics **28**, 2002 (2002)
22. Toutanova, K., Brockett, C., Gamon, M., Jagarlamudi, J., Hisami, S., Vanderwende, L.: The pythy summarization system: Microsoft research at DUC 2007. In: Proceedings of the HLT-NAACL Workshop on the Document Understanding Conference (DUC-2007). Rochester, USA (2007)
23. Yeh, J.Y., Ke, H.R., Yang, W.P.: Chinese text summarization using a trainable summarizer and latent semantic analysis. In: ICADL '02: Proceedings of the 5th International Conference on Asian Digital Libraries, pp. 76–87. Springer-Verlag, London, UK (2002)