Integrating Document Structure into a Multi-Document Summarizer

Aurélien Bossard and Thierry Poibeau
Laboratoire d'Informatique de Paris-Nord
CNRS and Université Paris 13
99, avenue Jean-Baptiste Clément — F-93430 Villetaneuse
{firstname.lastname}@lipn.univ-paris13.fr

Abstract

In this paper, we present a novel approach for automatic summarization. CBSEAS, the system implementing this approach, integrates a method to detect redundancy at its very core, in order to produce more expressive summaries than previous approaches. The evaluation of our system during TAC 2008 —the Text Analysis Conference— revealed that, even if our system performed well on blogs, it had some failings on news stories. A *post-mortem* analysis of the weaknesses of our original system showed the importance of text structure for automatic summarization, even in the case of short texts like news stories. We describe some ongoing work dealing with these issues and show that first experiments provide a significant improvement of the results.

Keywords

Multi-document Summarization; Text structure; Evaluation; Text Analysis Conference.

1 Introduction

During the past decade, automatic summarization, supported by evaluation campaigns and a large research community, has shown fast and deep improvements. Indeed, the research in this domain is guided by strong industrial needs: fast processing despite ever increasing amount of data.

We have developed a system called CBSEAS that integrates a new method to detect redundancy at its very core, in order to produce more expressive summaries than previous approaches. We have evaluated our system by participating in two tasks of TAC 2008 (the Text Analysis Conference):

- Opinion Task (Summarizing opinions found in blogs);
- Update Task (News stories summarization and detecting updates).

We obtained very competitive results during TAC 2008 on the "Opinion Task". However, our system did not rank as well on the "Update Task". A *post-mortem* analysis of the weaknesses of our original system revealed the importance of text structure for automatic summarization, even in the case of short texts like news stories.

Therefore, we will only focus on the "Update task" in this paper. We present our approach for automatic summarization and the first results of our current work dealing with the detection of document structure along with its integration for the production of summaries. The reader who wants to get information on the system we have developed for the Opinion task —for which we obtained among the best results— may refer to the system description in the TAC 2008 proceedings, see [1].

The rest of this paper is structured as follows: we first give a quick overview of the state of the art. We then describe our system, focusing on the most important novel features implemented and on the results obtained for the TAC 2008 "Update" task. Lastly, we show that news stories structure is meaningful and we detail some preliminary techniques that improve the results.

2 Related Works

Interest in creating automatic summaries began as soon as in the 1950s with the work by Luhn at IBM [8]. Following this line of research, Edmundson [3] proposed a set of features in order to assign a score to each sentence of a corpus and rank them accordingly: the sentences which get the highest scores are the ones to be extracted. The features that Edmundson used were the sentence position (in a news stories for example, the first sentences are the most important ones), the presence of proper names and keywords in the document title, the presence of indicative phrases and the sentence length.

More recently, research has mainly focused on multi-document summarization. In this context, a central issue consists in eliminating redundancy since the risks of extracting two sentences conveying the same information is more important than in the single-document paradigm. Moreover, identifying redundancy is a critical task, as information appearing several times in different documents is supposed to be important.

The "centroid-based summarization" method developed by Radev and his colleagues [9] is probably the most popular one in the field. It consists in identifying the centroid of a cluster of documents, that is to say the terms which best describe the documents to summarize. Then, the sentences to be extracted are the ones that are closest to the centroid. Radev implemented this method in an online multi-document summarizer called MEAD.

Radev further improved MEAD using a method inspired by the concept of *prestige* in social networks. This method called “graph-based centrality” [4] 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. Sentence selection is then performed by picking the sentences which have been visited most after a random walk on the graph. The main limitation of this method is that it only selects central sentences, which means that most of them can be redundant. It is thus necessary to add a module to detect redundancy before producing the final summary.

In order to avoid dealing with redundancy as a post-processing task, various methods have been proposed to integrate redundancy detection during the summarization process itself. For example, Goldberg [10] uses a “Markov absorbing chain random walk” on a graph representing the different sentences of the corpus to summarize.

MMR-MD, introduced by Carbonnel in [2], is a measure that needs a “passage” (snippet) clustering: all passages considered as paraphrases are grouped into the same clusters. MMR-MD takes into account the similarity to a query, the coverage of a passage (clusters that it belongs to), the content of the passage, the similarity to passages already selected for the summary, the fact that it belongs 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 threshold has to be specifically defined for each new corpus, which is the main weakness of this approach.

Our method is inspired from these last series of work: we think that it is crucial to integrate redundancy identification as soon as possible, and not as a last processing step. The main novelty of our approach is that we try to better characterize the content of news stories depending on their type. Most summarizers keep using standard features introduced in [3] to rank the sentences and do not take into account the document structure itself. Our goal is to determine the impact of the type and structure of news stories in automatic summarization, since these features have rarely been used.

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

We give in this section a brief overview of our TAC-2008 summarization system. Since we are most interested in the improvements we have added to the system since then, we will not give the full details but the reader may have a look at our TAC-2008 paper to get a more thorough description [1].

```
for all $e_j inE$
    $C_1 \leftarrow e_j$
for i from 1 to k do
    for j from 1 to i
        center($C_j$) $\leftarrow e_m | e_m maximizes \sum_{e_n in C_j} sim(e_m, e_n)$
    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. 1: *Fast global k-means algorithm*

We assume that, for multi-document summarization, redundant pieces of information are the most important elements to produce a good summary. Therefore, the sentences which carry those pieces of information have to be extracted. Detecting groups of sentences conveying the same information is the first step of our approach. The developed algorithm first establishes the similarities between all sentences of the documents to summarize, and then apply a clustering algorithm — fast global k-means [6] — to the similarity matrix in order to create clusters in which sentences convey the same information.

First, our system ranks all the sentences according to their similarity to the documents centroid, or to the user query if there is one. We have chosen to build up the documents centroid with the m most important terms, importance being reflected by the tf/idf of each terms. We then select, to create a n sentences long summary, the n^2 best ranked sentences. We do so because the clustering algorithm we use to detect sentences conveying the same information, fast global k-means, behaves better when it has to group n^2 elements into n clusters. The similarity with the centroid is a weighted sum of terms appearing in both centroid and sentence, normalized by sentence length.

Once the similarities are computed, we cluster the sentences using fast-global kmeans (description of the algorithm is in figure 1) using the similarity matrix. It works well on a small data set with a small number of dimensions, although it has not yet scaled up as well as we would have expected.

This clustering step completed, we select one sentence per cluster in order to produce a summary that contains most of the relevant information/ideas from the original documents. We do so by choosing the central sentence in each cluster. The central sentence is the one which maximizes the sum of similarities with the other sentences of its cluster. It should be the one that characterizes best the cluster in terms of conveyed information.

The overall process of our summarization system is shown in fig. 2.

4 CBSEAS at TAC 2008

In this section, we briefly describe the TAC 2008 “Update” task and the adaptation we had to implement in order to make our system compliant with the task requirements. Here again, the interested reader can

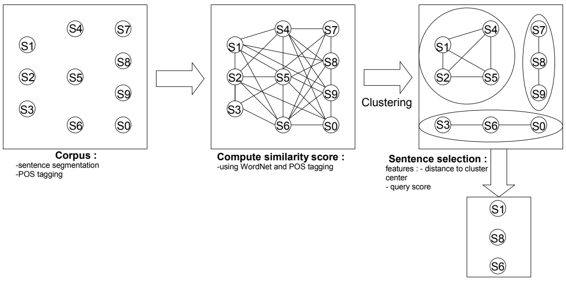
Corpus :
-sentence segmentation
-POS tagging
Compute similarity score :
-using WordNet and POS tagging
Clustering
Sentence selection :
features : - distance to cluster center
- query score

Fig. 2: *Summarization system*

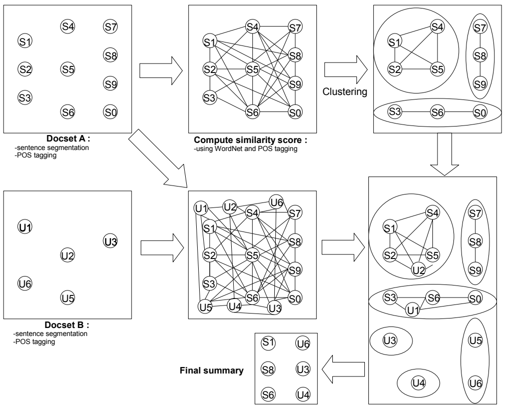

Fig. 3: *CBASES Update system*

refer to [1] for more details.

4.1 Description of the Update Task

The "Update task" consists in generating two types of summaries for each evaluation topic. Each topic is composed of a user query and of two groups of documents. Documents are extracted from the AQUAINT-2 corpus (a collection of news stories issued by several press agencies). The first type of summary is the "standard" one, a simple summary of the first document set. The second type of summary is more complex: it has to summarize the information found in the second document set that was not already present in the first document set. Summaries are to be 100 words long at most.

For the *Update task*, two evaluations were given to participants: the first one using PYRAMID, the second one using ROUGE scores [5]. The PYRAMID score depends on the number of basic semantic units the summary contains which are considered as important by human annotators (the importance of a semantic unit depends on the number of times it appears in the summaries generated by human annotators). Summaries have also been scored using five different scores attributed manually for grammaticality, non-redundancy, structure, fluency and overall responsiveness (responsiveness is a subjective score corresponding to the question "How much would you pay for that summary?"). ROUGE metrics are based on n-gram comparison between the automatic summary and a reference summary which has been written by TAC annotators.

4.2 Adaptation of CBSEAS for the "Update Task"

Our system, CBSEAS, is a "standard" summarization system. We had to adapt it in order to deal with the specific requirements of TAC 2008.

The adaptation for the "Update Task" mainly consisted in managing update. The first step, summarizing the first document set, is done using CBSEAS as it stands. After the selection of sentences for the first document set, we re-compute all sentence similarities including the new sentences (i.e. sentences from the second document set).

We cluster the first document set and mark all the concerned sentences as immobile. Using fast global kmeans, we continue the clustering process by adding new clusters, with the following constraints:

- sentences from the first document set cannot be moved to another cluster;
- the cluster centres from the first clustering must not be recalculated.

Doing so, sentences from the second document set which are supposed to be (semantically) close to sentences from the first document set are added to old clusters, whereas sentences which appear to bring novelty are added to new clusters. These new clusters include the sentences from which the second summary will be produced. Fig. 3 gives an overview of the "Update" system.

The way *update* is managed is very specific to our system, and taking into account its results should distort the pure summarizing results obtained by CBSEAS. For this reason, we will only present in 4.3 the results obtained by the summaries of the first documents sets.

4.3 TAC 2008 Results for the "Update task"

Contrary to our system for the Opinion Task that behaved quite well, the system used for the "Update Task" did not obtain good results. Results presented in fig. 4 are results computed with ROUGE [5], an automatic scoring measure that compares automatic summaries to a gold standard. Manual evaluation has been provided to participants, but the results presented here are those of TAC for CBSEAS 0.2 and results obtained after TAC on the same evaluation dataset for CBSEAS 0.5. We only provide ROUGE results since those can be calculated automatically and, therefore, provide a good basis for comparison (even if we are conscious that a manual evaluation would give a more thorough insight on our results).

CBSEAS 0.2 obtained poor results. Our system tried to always create summaries under the 100 words

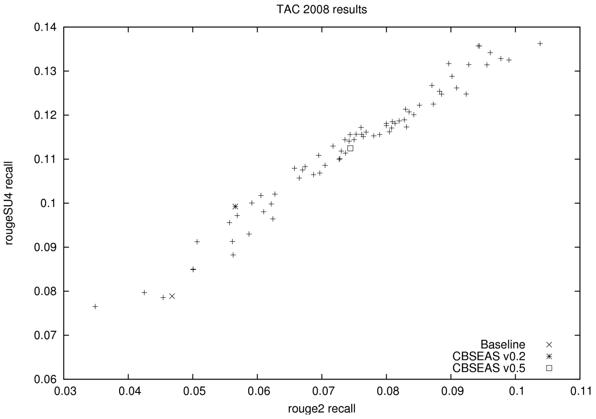

Fig. 4: *Results of our system on the "Update" task*

limit given by TAC organizers, eliminating one by one all the sentences, starting from the worst ranked. Our summaries were 67 words long on average. We corrected that point in CBSEAS 0.5, cutting every summary exactly at 100 words, even if this means removing abruptly the end of a sentence. One can see that CBSEAS 0.5, just by using this trick, obtained significantly better results than CBSEAS 0.2. However, CBSEAS 0.5 is still not among the ten best systems. One hypothesis is that most of the evaluated systems used features that were specifically tuned for this task and this type of texts. For example, favouring the first sentence of an article is a strategy often used. Our system does not use such information.

However, the studies on which those features are based neither take into account the document type nor its structure. That is why we propose to study news stories structure and integrate it to our summarizing system (see section 5).

4.4 Studying the Impact of the Clustering Step

As most of the actual summarization systems do not use any clustering techniques to group related sentences together, we wanted to check whether this clustering phase does improve the summaries quality or not. For this purpose, we made two different tests: one running the system and attributing random classes to the sentences, and the other selecting the n best ranked sentences using the same scoring function, as shown above.

These two tests allow us to see if we get more benefit by eliminating redundant sentences and eventually false positive redundant sentences that could be essential to the creation of a summary than selecting the best ranked sentences, and if our clustering algorithm behaves well for this task in combination to our sentence similarity measure.

We can see in fig. 5 the results of these two experiments, marked as "Random" and "No-Clust". These results prove that the clustering step has an impact on the quality of our summaries. If the redundancy is well managed, the overall results obtained by CBSEAS v0.5 on the two tests tend to show that our system does not necessarily integrate the best suited sentences into the final summary, and does not optimize the number of information that can be found in a text (using for example sentence compression techniques).

In what follows, we propose a method to improve sentence selection, not simply based on statistical measures, but taking into account the document structure in order to weight sentence centrality.

5 Analysing News Story Structure

In this section, we present the work done on the analysis of news story structure.

5.1 Categorizing news stories

A recent study on news stories has been held by N. Lucas [7]. In this study, Lucas proposes the following news story categorization:

- "Commented" news stories (made of two different parts; first part: factual explanation; second part: projection in the future of the expected evolution of the current situation)
- "Elaborated" news stories (concerning more than one event)
- "Action" news stories (reporting a line of events directly linked together; according to N. Lucas, market newswires also belong to this type of news).

She also stated that "commented" news always follow the same temporal presentation. First, the author presents the current event. Then, he gives an explanation of this event based on past events. In the end, the author tells the reader what could or will be the consequences of this event in the future. Identifying these three parts can be very useful, as they follow the classic rule of news writing: the first sentence is the most important one.

Studying the AQUAINT-2 corpus, we identified more types of news:

- Opinion reviews,
- Speech reports,
- Chronologies,
- Comparative news,
- Enumerative news.

The last three categories are very interesting for an automatic summarization task. In fact, they make up at most 5% of the total number of news stories in AQUAINT-2 but, in the training corpus of the "Update Task", they contain 80% of the relevant information. Moreover, they are written in a concise style, and the sentences these news contain have specific characteristics that make them easier to categorize than the other ones:

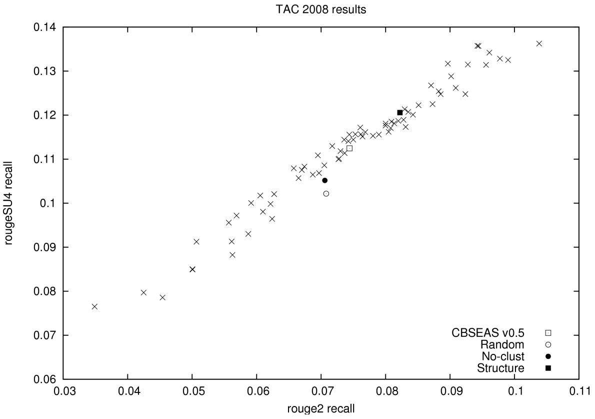

Fig. 5: *Post-Campaigns Experiments*

- Chronologies have almost all their paragraphs beginning with a time reference;
- Chronologies often start with a key phrase such as "*Here is a timeline of events surrounding the election:*";
- Comparative news and enumerative news contain lists which are well structured;
- The elements of a list in a comparative news begin with terms that belong to the same category (for example, country names).

We have implemented a simple categorizer which classifies the news in four groups: chronologies, comparative news, enumerative news and classic news. We plan to develop a more complete system that classifies all the news into the different categories we have identified.

We have evaluated our categorizer on a part of AQUAINT-2 (300 documents) that has been manually annotated. We obtained 100% precision and 81% recall for chronologies, 73% precision and 65% recall for comparative newswire, and 65% precision and 67% recall for enumerative newswire.

We have integrated the categorization to CBSEAS, and forced the system to favor sentences extracted from non-classic news, giving them a 15% bonus on the scoring function. This method is too discriminating, but this is only a preliminary study. We compared the ROUGE-SU4 scores of summaries of groups of documents which contain at least one non classic news and noted a 10% improvement. This ranks our system 21st instead of 39th.

These results encourage us to keep on studying the news structure and integrating it to CBSEAS.

5.2 Future work

Our news categorizer still needs to be worked on: the method to categorize news only recognizes the three categories which are the simplest ones to identify. The other categories have their own properties and ranking sentences by importance using document structure is different from one category to another. News structure and temporality are bound together. Using machine learning techniques on temporaly annotated documents can be a solution to categorize news.

6 Conclusion

We have presented a new approach for multi-document summarization. It uses an unsupervised clustering method to group semantic related sentences together. It can be compared to approaches using sentence neighbourhood [4], because the sentences which are highly related to the highest number of sentences are those which will be extracted first. However, our approach is different since sentence selection is directly dependent on redundancy analysis. This is the reason why redundancy elimination, which is crucial in multi-document summarization, takes place at the same time as sentence selection. We also proposed a way to improve the quality of news summaries using the news story structure. We showed, by integrating some basic structure traits in the summarization process, that it really boosts the quality of the summaries.

References

[1] A. Bossard, M. Généreux, and T. Poibeau. Description of the LIPN Systems at TAC2008: Summarizing Information and Opinions. In *Text Analysis Conference 2008, Workshop on Summarization Tracks*, National Institute of Standards and Technology, Gaithersburg, Maryland USA, 2004.

[2] J. Carbonell and J. Goldstein. 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 on Research and development in information retrieval*, pages 335–336, New York, NY, USA, 1998. ACM.

[3] H. P. Edmundson and R. E. Wyllys. Automatic Abstracting and Indexing—Survey and Recommendations. *Commun. ACM*, 4(5):226–234, 1961.

[4] G. Erkan and D. R. Radev. LexRank: Graph-based Centrality as Salience in Text Summarization. *Journal of Artificial Intelligence Research (JAIR)*, 2004.

[5] C.-Y. Lin. ROUGE: a Package for Automatic Evaluation of Summaries. In *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*, Barcelona, Spain, 2004.

[6] S. López-Escobar, J. A. Carrasco-Ochoa, and J. F. M. Trinidad. Fast Global *k*-Means with Similarity Functions Algorithm. In E. Corchado, H. Yin, V. J. Botti, and C. Fyfe, editors, *IDEAL*, volume 4224 of *Lecture Notes in Computer Science*, pages 512–521. Springer, 2006.

[7] N. Lucas. The Enunciative Structure of News Dispatches, a Contrastive Rhetorical Approach. In *Proceedings of the ASLA Conference*, pages 154–164, 2005.

[8] H. Luhn. The Automatic Creation of Literature Abstracts. *IBM Journal*, 2(2):159–165, 1958.

[9] D. Radev, T. Allison, S. Blair-Goldensohn, J. Blitzer, A. Çelebi, S. Dimitrov, E. Drabek, A. Hakim, W. Lam, D. Liu, J. Otterbacher, H. Qi, H. Saggion, S. Teufel, M. Topper, A. Winkel, and Z. Zhu. MEAD — a Platform for Multidocument Multilingual Text Summarization. In *Proceedings of LREC 2004*, Lisbon, Portugal, May 2004.

[10] X. Zhu, A. Goldberg, J. Van Gael, and D. Andrzejewski. Improving Diversity in Ranking using Absorbing Random Walks. *Proceedings of HLT-NAACL*, pages 97–104, 2007.