CBSEAS, a Summarization System
Integration of Opinion Mining Techniques to Summarize Blogs

Aurélien Bossard, Michel Généreux and **Thierry Poibeau**
Laboratoire d'Informatique de Paris-Nord
CNRS UMR 7030 and Université Paris 13
93430 Villetaneuse — France
{firstname.lastname}@lipn.univ-paris13.fr

Abstract

In this paper, we present a novel approach for automatic summarization. Our system, called CBSEAS, integrates a new method to detect redundancy at its very core, and produce more expressive summaries than previous approaches. Moreover, we show that our system is versatile enough to integrate opinion mining techniques, so that it is capable of producing opinion oriented summaries. The very competitive results obtained during the last Text Evaluation Conference (TAC 2008) show that our approach is efficient.

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.

In this paper, we present a novel approach for automatic summarization. Our system, called CBSEAS, integrates a new method to detect redundancy at its very core, and produce more expressive summaries than previous approaches. The system is flexible enough to produce opinion oriented summaries by accommodating techniques to mine documents that express different views or commentaries. The very competitive results obtained during the last Text Evaluation Conference (TAC 2008) show that our approach is efficient.

This short 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. Lastly, we give the details of the results obtained on the TAC 2008 Opinion Pilot task.

2 Related works

Interest in creating automatic summaries has begun in the 1950s (Luhn, 1958). (Edmundson and Wyllys, 1961) proposed features to assign a score to each sentence of a corpus in order to rank these sentences. The ones with the highest scores are kept to produce the summary. The features they used were sentence position (in a news article for example, the first sentences are the most important), proper names and keywords in the document title, indicative phrases and sentence length.

Later on, summarizers aimed at eliminating redundancy, especially for multi-documents summarizing purpose. Identifying redundancy is a critical task, as information appearing several times in different documents can be qualified as important.

Among recent approaches, the "centroid-based summarization" method developed by (Radev et al., 2004) consists in identifying the centroid of a cluster of documents, in other words the terms which best suit the documents to summarize. Then, the sentences to be extracted are the ones that contain the greatest number of centroids. Radev implemented this method in an online multi-document summarizer, MEAD.

Radev further improved MEAD using a different method to extract sentences: "Graph-based centrality" extractor (Erkan and Radev, 2004). 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. Sentence selection is then performed by picking the sentences which have been visited most after a random walk on the graph.

The last two systems are dealing with redundancy as a post-processing step. (Zhu et al., 2007), assuming that redundancy should be the concept on what is based multi-document summarization, offered a method to deal with redundancy at the

Proceedings of the EACL 2009 Demonstrations Session, pages 5–8,
Athens, Greece, 3 April 2009.

same time as sentence selection. For that purpose, the authors 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 (Carbonell and Goldstein, 1998), 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; an unsupervised clustering method is best suited for automatic summarization, as the corpora we need to summarize are different from one to another. Moreover, sentence synonymy is also dependent on the corpus granularity and on the user compression requirement.

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

We assume that, in multi-document summarization, redundant pieces of information are the single most important element to produce a good summary. Therefore, the sentences which carry those pieces of information have to be extracted. Detecting these 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, then applies a clustering algorithm — fast global k-means (López-Escobar et al., 2006) — 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. We have chosen to build up the documents centroid with the m most important terms, their importance being reflected by the tf/idf of each term. We then select the n^2 best ranked sentences to create a n sentences long summary. We do so because the clustering algorithm we use to detect sentences

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) $\leftarrow e_m | e_m maximizes \sum_{e_n inC_j} sim(e_m, e_n)$
  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

Figure 1: Fast global k-means algorithm

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.

Similarity between sentences is computed using a variant of the "Jaccard" measure. If two terms are not equal, we test their synonymy/hyperonymy using the Wordnet taxonomy (Fellbaum, 1998). In case they are synonyms or hyperonym/hyponym, these terms are taken into account in the similarity calculation, but weighted respectively half and quarter in order to reflect that term equality is more important than term semantic relation. We do this in order to solve the problem pointed out in (Erkan and Radev, 2004) (synonymy was not taken into account for sentence similarity measures) and so to enhance sentence similarity measure. It is crucial to our system based on redundancy location as redundancy assumption is dependent on sentence similarities.

Once the similarities are computed, we cluster the sentences using fast global k-means (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 in 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 information vehicled.

4 TAC 2008: The Opinion Summarization Task

In order to evaluate our system, we participated in the Text Analysis Conference (TAC) that proposed in 2008 an opinion summarization task. The goal is to produce fluent and well-organized summaries of blogs. These summaries are oriented by complex user queries, such as "Why do people like.....?" or "Why do people prefer... to...?".

The results were analyzed manually, using the PYRAMID method (Lin et al., 2006): the PYRAMID score of a summary depends on the number of simple semantic units, units considered as important by the annotators. The TAC evaluation for this task also included grammaticality, non-redundancy, structure/coherence and overall fluency scores.

5 CBSEAS Adaptation to the Opinion Summarization Task

Blog summarization is very different from a newswire article or a scientific paper summarization. Linguistic quality as well as reasoning structure are variable from one blogger to another. We cannot use generalities on blog structure, neither on linguistic markers to improve our summarization system. The other problem with blogs is the noise due to the use of unusual language. We had to clean the blogs in a pre-processing step: sentences with a ratio *number of frequent words/total number of words* below a given threshold (0.35) were deemed too noisy and discarded. Frequent words are the one hundred most frequent words in the English language which on average make up approximately half of written texts (Fry et al., 2000).

Our system, CBSEAS, is a "standard" summarization system. We had to adapt it in order to deal with the specific task of summarizing opinions. All sentences from the set of documents to summarize were tagged following the opinion detected in the blog post they originated from. We used for that purpose a two-class (positive or negative) SVM classifier trained on movie reviews. The idea behind the opinion classifier is to improve summaries by selecting sentences having the same opinionated polarity as the query, which were tagged using a SVM trained on the manually tagged queries from the training data provided earlier in TAC.

As the Opinion Summarization Task was to produce a query-oriented summary, the sentence pre-selection was changed, using the user query instead of the documents centroid. We also changed the sentence pre-selection ranking measure by weighting terms according to their lexical category; we have chosen to give more weight to proper names than verbs adjectives, adverbs and nouns. Indeed, opinions we had to summarize were mostly on products or people.

While experimenting our system on TAC 2008 training data, we noticed that extracting sentences which are closest to their cluster center was not satisfactory. Some other sentences in the same cluster were best fitted to a query-oriented summary. We added the sentence ranking used for the sentence pre-selection to the final sentence extractor. Each sentence is given a score which is the distance to the cluster center times the similarity to the query.

6 TAC 2008 Results on Opinion Summarization Task

Participants to the Opinion Summarization Task were allowed to use extra-information given by TAC organizers. These pieces of information are called snippets. The snippets contain the relevant information, and could be used as a stand-alone dataset. Participants were classified into two different groups: one for those who did not use snippets, and one for those who did. We did not use snippets at all, as it is a more realistic challenge to look directly at the blogs with no external help. The results we present here are those of the participants that were not using snippets. Indeed, systems using snippets obtained much higher scores than the other systems. We cannot compare our system to systems using snippets.

Our system obtained quite good results on the "opinion task": the scores can be found on figure 2. As one can see, our responsiveness scores are low compared to the others (responsiveness score corresponds to the following question: "*How much would you pay for that summary?*"). We suppose that despite the grammaticality, fluency and pyramid scores of our summaries, judges gave a bad responsiveness score to our summaries because they are too long: we made the choice to produce summaries with a compression rate of 10% when it was possible, the maximum length authorized otherwise.

Evaluation	CBSEAS	Mean	Best	Worst	Rank
Pyramid	.169	.151	.251	.101	5/20
Grammatic.	5.95	5.14	7.54	3.54	3/20
Non-redun.	6.64	5.88	7.91	4.36	4/20
Structure	3.50	2.68	3.59	2.04	2/20
Fluency	4.45	3.43	5.32	2.64	2/20
Responsiv.	2.64	2.61	5.77	1.68	8/20

Figure 2: Opinion task overall results

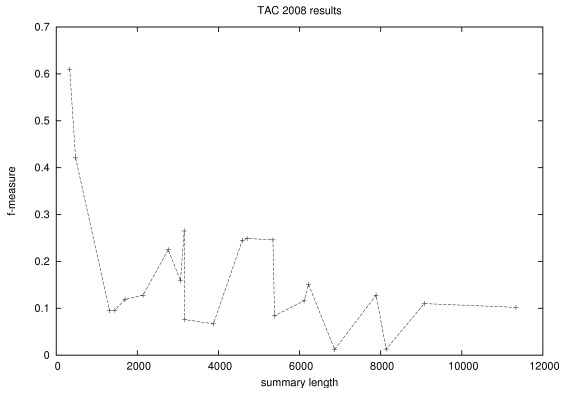

Figure 3: Opinion task results

However, we noticed that the quality of our summaries was very erratic. We assume this is due to the length of our summaries, as the longest summaries are the ones which get the worst scores in terms of pyramid f-score (fig 3). The length of the summaries is a ratio of the original documents length. The quality of the summaries would be decreasing while the number of input sentences is increasing.

Solutions to fix this problem could be:

- Define a better score for the correspondence to a user query and remove sentences which are under a threshold;
- Extract sentences from the clusters that contain more than a predefined number of elements only.

This would result in improving the pertinence of the extracted sentences. The users reading the summaries would also be less disturbed by the large amount of sentences a too long summary provides. As the "opinion summarization" task was evaluated manually and reflects well the quality of a summary for an operational use, the conclusions of this evaluation are good indicators of the quality of the summaries produced by our system.

7 Conclusion

We presented here a new approach for multi-document summarization. It uses an unsupervised clustering method to group semantically related sentences together. It can be compared to approaches using sentence neighbourhood (Erkan and Radev, 2004), 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 location. Also, redundancy elimination, which is crucial in multi-document summarization, takes place in the same step as sentence selection.

References

Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *SIGIR'98*, pages 335–336, New York, NY, USA. ACM.

Harold P. Edmundson and Ronald E. Wyllys. 1961. Automatic abstracting and indexing—survey and recommendations. *Commun. ACM*, 4(5):226–234.

Güneş Erkan and Dragomir R. Radev. 2004. Lexrank: Graph-based centrality as salience in text summarization. *Journal of Artificial Intelligence Research (JAIR)*.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Edward Bernard Fry, Jacqueline E. Kress, and Dona Lee Fountoukidis. 2000. *The Reading Teachers Book of Lists*. Jossey-Bass, 4th edition.

Chin-Yew Lin, Guihong Cao, Jianfeng Gao, and Jian-Yun Nie. 2006. An information-theoretic approach to automatic evaluation of summaries. In *Proceedings of HLT-NAACL*, pages 463–470, Morristown, NJ, USA.

Saúl López-Escobar, Jesús Ariel Carrasco-Ochoa, and José Francisco Martínez Trinidad. 2006. Fast global -means with similarity functions algorithm. In *IDEAL*, volume 4224 of *Springer, Lecture Notes in Computer Science*, pages 512–521.

H.P. Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal*, 2(2):159–165.

Dragomir Radev et al. 2004. MEAD - a platform for multidocument multilingual text summarization. In *Proceedings of LREC 2004*, Lisbon, Portugal.

Xiaojin Zhu, Andrew Goldberg, Jurgen Van Gael, and David Andrzejewski. 2007. Improving diversity in ranking using absorbing random walks. In *Proceedings of HLT-NAACL*, pages 97–104, Rochester, USA.