K.U.Leuven summarization system - DUC 2003

Roxana Angheluta, Marie-Francine Moens, Rik De Busser
Katholieke Universiteit Leuven
Interdisciplinary Center for Law & IT
Tiensestraat 41, B-3000 Leuven, Belgium
{roxana.angheluta, marie-france.moens, rik.debusser}@law.kuleuven.ac.be

1 Introduction

K.U.Leuven summarizer for DUC 2003 [1] is partially based on the summarizer developed for DUC 2002 [1]. It is a composite system that uses different strategies for the three tasks in which we participated: making very short single-document summaries (headlines), making multi-document summaries from a set of related documents and making viewpoint-oriented multi-document summaries. We combined topic segmentation with clustering techniques for the multi-document summaries and used topic segmentation, sentence scoring and compression for the first task. We obtained very good results for the headlines and good results for the other tasks.

2 Preprocessing

Having the experience of last year, this year we paid special attention to the preprocessing part. We tried to determine corpus-independent rules for cleaning the text.

We only retained the text between <TEXT> tags and split it into sentences with the sentence splitter provided at DUC 2002. The actual cleaning consisted in removal of the very short sentences, of the text between brackets and - in some cases - of the direct speech. While direct speech in many cases contains personal opinions, which are not interesting for summaries (e.g. *"I don't hold much hope," she added in a telephone.*), indirect speech often ads valuable information to the text (e.g. *China's government said Thursday that two prominent dissidents arrested this week are suspected of endangering national security _ the clearest sign yet Chinese leaders plan to quash a would-be opposition party*). We therefore retained sentences containing indirect speech. Since we tried to be as generic as possible, cleaning was sometimes rather crude.

In the preprocessing step we extracted also collocations from each document, using the likelihood ratio method described in [4].

3 Headlines

The first task was to create short single-document summaries of approximately 10 words, no specific format other than linear. For this task, it was important that the headlines cite the most relevant topics or entities in a text and also mention significant semantic

[1] We thank the Institute for the Promotion of Innovation by Science and Technology in Flanders for sponsoring this research. We also thank Patrick Jeuniaux for his help with the evaluation.

relationships between these entities. Such a strategy is reflected in our two-step algorithm.

3.1 Topic terms selection

In order to detect the most important entities in a document, we constructed the topic trees of the texts, using the topic segmentation algorithm developed by our group [7]. The algorithm is based on linguistic theories of sentence topic and focus. The topic trees offer a hierarchical representation of a text in the form of a tree and can be seen themselves as a kind of short summary. From the tree in figure 1 one can deduce that the text is about Monica Lewinsky and Bill Clinton, probably about offering her immunity. The summary is not perfect and someone with no prior knowledge about the case might have difficulty to understand its content, but it could be useful for quickly scanning and selecting interesting documents.

We selected the first (a number given by a parameter) most important topic terms from each tree, given by their coverage in the text (see figure 1, on the right part of each line there are the file pointers of the text spanned by each topic).

3.2 Headlines construction

In order to detect the relationships between terms, we selected important sentences from the document and reduced them to their main content. The importance of a sentence was measured by the overlap of its words with the topic terms. We ordered the sentences, using three heuristics in the case of a tie:

- proximity (sentences where the topic terms were close to each other were ranked higher)
- the number of words they contained (shorter sentences were ranked higher)
- sentence number (first sentences from the documents were ranked higher)

We concatenated the sentences obtained with the 3 heuristics and ended up with a small set of important sentences, in average 4 for each document (see below).

Monica Lewinsky has been given full immunity in exchange for testimony in Kenneth Starr's six-month investigation of her relationship with President Bill Clinton, Lewinsky's attorneys said Tuesday.
The source said the earlier proffer contained "a fair amount of information" dealing with Lewinsky's conversations with the president and his confidants about how they would deal with Mrs. Jones' sexual harassment lawsuit against Clinton.
Transactional immunity means that Lewinsky will not be prosecuted for any testimony that she gives to Starr's office regarding matters under investigation.

Sentences extracted from set d113e, doc. XIE19980729.0051

A further reduction of the selected sentences made use of a parser. After examining a number of different parsers available on the Internet, we opted for Charniak's parser [2], because of its performances and its clear and easy to parse output.

Each headline was constructed from the parse trees by outputting each subtree that connected different topic terms and pruning away embedded clauses. At the implementation level, this corresponds with splitting the sentences into clauses (see figure 2), ordering them by the overlap with the topic terms and outputting the substrings between the terms. This method proved to be a natural selection of the most important phrases, filtering out noise like "DT_The NN_source VBD_said ._.". We employed here the collocates of the topic terms, getting more comprehensible phrases (*President Bill Clinton* instead of *President*).

Finally, we cleaned the selected phrases by removing auxiliary verbs and determiners. We outputted the phrases till the desired length limit. We set the length-limit to 25 because the baselines were allowed to be as long.

```
Monica Lewinsky        0       1482
        President Bill Clinton 732     809
        means Monica Lewinsky  810     1482
                immunity        966     1042
                source  1043    1281
                president  Bill Clinton    1282    1482
```

Figure 1: Topic tree: set d113e; doc. XIE19980729.0051.TOC

NNP_Monica NNP_Lewinsky AUX_has AUX_been VBN_given JJ_full NN_immunity IN_in NN_exchange IN_for NN_testimony IN_in NNP_Kenneth NNP_Starr POS_'s JJ_six-month NN_investigation IN_of PRP$_her NN_relationship IN_with NNP_President NNP_Bill NNP_Clinton

,_, NNP_Lewinsky POS_'s NNS_attorneys VBD_said NNP_Tuesday ._.

PRP_they MD_would VB_deal IN_with NNP_Mrs. NNP_Jones POS_' JJ_sexual NN_harassment NN_lawsuit IN_against NNP_Clinton DT_the JJR_earlier NN_proffer VBD_contained "_" DT_a JJ_fair NN_amount IN_of NN_information "_" VBG_dealing IN_with NNP_Lewinsky POS_'s NNS_conversations IN_with DT_the NN_president CC_and PRP$_his NNS_confidants IN_about WRB_how

DT_The NN_source VBD_said ._.

PRP_she VBZ_gives TO_to NNP_Starr POS_'s NN_office VBG_regarding NNS_matters IN_under NN_investigation

NNP_Lewinsky MD_will RB_not AUX_be VBN_prosecuted IN_for DT_any NN_testimony IN_that

JJ_Transactional NN_immunity VBZ_means IN_that ._.

Figure 2: The clauses detected for set d113e, doc. XIE19980729.0051

a) ABNORMALITIES CONTRIBUTING TO DEVELOPMENT OF SCHIZOPHRENIA IN REGION; SCIENTISTS; STUDIES; RESEARCHERS;
b) PEOPLE PALMS AND FINGERPRINTS MAY BE USED TO DIAGNOSE SCHIZOPHRENIA AND OTHER MENTAL DISORDERS GROUP OF CHINESE PSYCHIATRISTS ANNOUNCED;
c) PATIENTS; TREATMENT; EFFECT; HOFFMAN; TMS; SCALE; STUDY; DR. MARK S. GEORGE; RESULTS; GOAL;

Figure 3: Very short single-document summaries for set d100a a) obtained from clauses (doc. APW20000507.0061) ; b) obtained from the first sentence of the document (doc. XIE19960807.0183); c) obtained from the topic terms and their collocates (doc. APW19990519.0113)

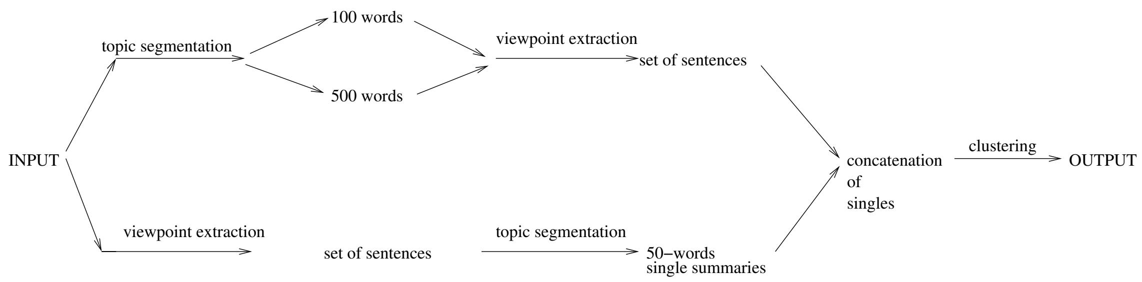

Figure 4: Workflow of the algorithms for construction of the viewpoint oriented multi-document summaries

In 71.64% of the cases we could send headline phrases built with the above approach. In the rest of the cases we lacked sentences with multiple important topic terms. In 7.37% we could chose the first sentence of the document that - based on our topic-segmentation algorithm - usually contains one important topic term and removed determiners and auxiliary verbs resulting in a headline whose length was acceptable. If the headlines were shorter than desired, they were augmented with topic terms. In the remaining 20.99% of the cases we used the collocated topic terms as headlines.

Three examples of a very short summary, corresponding with the three possible cases, are presented in figure 3.

4 Multi-document summaries

For the multidocument summaries we used last year's method and findings. Two main ideas were followed: 1) in order to obtain a good multi-document summary is better to start from single-document summaries than from the entire documents and 2) in particular it is better to start from short single summaries than from long ones. First, we constructed 50 word summaries with the topic segmentation algorithm. For picking important non-redundant sentences for the multi-document summaries, we used k-medoid clustering [5]. The complete algorithm is described in [1].

5 Multi-document summaries oriented by a viewpoint

This was a newly introduced task, linked with the idea of user-oriented summary: given each document cluster and a viewpoint description, create a short summary ($\approx$ 100 words) of the cluster from the point of view specified. The viewpoints were natural language strings and they described just a facet of the cluster of documents.

The idea we followed was to extract the most representative sentences for the document (we will call them important sentences) and from those to extract sentences representative for the viewpoints, or the other way around. The concatenation of such important viewpoint sentences from all the documents in a set was clustered afterwards to eliminate the redundancy. For measuring the importance of the sentences with respect to the viewpoints, we counted the overlap between sentence's words and viewpoint's important words (nouns, verbs and adjectives). The important sentences were extracted using the topic segmentation algorithm.

The workflow of the algorithm is illustrated in figure 4. The upper part of the figure shows first the application of the topic segmentation, from which 100 or 500 word summaries are constructed [1] and then extraction of the viewpoint sentences. The lower part of the figure corresponds with the other case: first viewpoint sentences are extracted and then topic segmentation is applied to generate a summary. In both cases the resulting sentences are clustered in the end.

An alternative method was not to use clustering, but rather just to select the first most important viewpoint sentences for the whole set (till the desired length of the summary) and to output them.

Since the topic segmentation algorithm is based on coherence of the text and extracting viewpoint's sentences did not lead to coherent texts, the results we got for this case were weaker than the rest. Seemingly the best results we got using first topic segmentation with output of 100 words single-summaries and then extracting viewpoint sentences. We sent this output for evaluation at DUC.

We also tried to enrich the list of nouns and verbs from the viewpoints with related terms that were extracted from WordNet or from an automatically constructed thesaurus [8]. However, because these enriched word lists were very noise (e.g. the thesaurus expanded *introduction* to *change, expansion, launch, overhaul, reduction, withdrawal, deployment, increase, modernization,...*), the resulting summaries were not very good.

6 Results and discussion

All summaries were evaluated for coverage and length-adjusted coverage conform with the measures described in [3]. The output of tasks 2 and 3 were also evaluated for quality and the output of the first task for usefulness.

The results for the first task are summarized in table 1 and plotted in 5. Our team has code 18. Letters A-J represent handmade summaries, 1-5 are baseline systems and 6-26 are participants. 13 teams participated in this task. As expected, manual summaries were better, with the exception of summaries J and A, which were beaten by system 8 [2]. The baseline is better than any of the automatic summaries, which shows that there is a lot of room for improvement. Compared to other automatic systems, we performed quite well on the coverage score, ending up on the third place (see table 1, first column). Since our summaries had a length average of around 15 words (with a maximum of 31 and a minimum of 2), we were penalized on length-adjusted coverage. There were two length-adjusted scores computed, the second deriving from the first (see table 1, second and third column). For details about how scores were computed, we refer to the DUC website [3].

In the usefulness evaluation (table 1, last column), we reached the second place. This might be explained by the fact that topic terms and their relationships are useful indicators for helping people to decide whether or not it is worth to read the entire article.

Sixteen teams participated in the second task. Like last year, we ended up in the middle of the group. Since we did not change anything to the summarization algorithm, this was to be expected.

For the third task (see table 2) ended up in the fourth position both for coverage and for length-adjusted coverage (see columns 2 and 4). Eleven teams participated. Cases in which viewpoints were a synthesis of all or some documents (e.g. *Chronology of "Peanuts" creator Charles Schulz's career* or *The Republican Presidential primaries of 2000 showed problems with the system as eleven announced candidates dropped out one by one until George W. Bush was left with no active opponents almost six months before the election* or *Steps leading up to the introduction of the euro* (sets d102, d108 and d127))were especially difficult and all teams performed rather poorly on these sets (see figure 6). One possible reason for our poor performance might be that many documents in this set were not very coherent and topic segmentation did not work well on them. For set d108, only some of the documents contained relevant information and it was often not explicitly present (need of information fusion from different sources). For this kind of task, extract-based summaries might not be very effective. Our system produced good summaries when the viewpoints contained keywords that were selected by the topic segmentation algorithm: *Hubble Space Telescope Service Mission* (set d112) (see figure 6).

7 Related research

For the headline task, our work is mostly related to the work of Knight and Marcu [6] and the work that

[2] As this system exceeded the length restrictions considerably, it was not compared with the others. We also did not consider it in our rankings

Summarizer code	Mean coverage	First mean length-adjusted coverage	Final mean length-adjusted coverage	Usefulness
E	0.60546448087432	0.40452459016394	0.36631693989071	3.1666666666667
F	0.6032967032967	0.40732967032967	0.358	3.1306306306306
B	0.57291666666667	0.38627604166667	0.35121875	3.4357798165138
G	0.56864864864865	0.37905945945946	0.33342162162162	3.1422222222222
C	0.55978835978836	0.38183597883598	0.33918518518518	3.4162790697674
D	0.52834224598931	0.35237967914439	0.32718181818182	2.9452054794521
H	0.51666666666667	0.34516666666667	0.3084375	3.137339055794
I	0.51595744680851	0.34483510638298	0.3006914893617	3.2573839662447
8	0.51332263242376	0.34222471910112	0.095894060995185	3.1298932384342
J	0.49052631578947	0.32697894736842	0.29747894736842	2.7067510548523
A	0.48804347826087	0.33913043478261	0.326125	3.207423580786
1	0.47980769230769	0.36767948717949	0.35922115384615	2.6325622775801
17	0.40160256410256	0.27353205128205	0.21917628205128	2.3784505788068
26	0.37788461538462	0.25503205128205	0.25435256410256	2.1094306049822
18	**0.35801282051282 (3)**	**0.24469711538462 (3)**	**0.17636217948718 (8)**	**2.1585040071238 (2)**
21	0.3176282051282	0.21183173076923	0.20181891025641	2.0275800711744
22	0.30608974358974	0.2304342948718	0.19252083333333	1.5729537366548
7	0.3	0.20459134615385	0.15263301282051	1.6966192170819
25	0.29198717948718	0.1946282051282	0.14569711538461	1.8236865538736
9	0.27467948717949	0.21730769230769	0.21265064102564	1.4466192170819
13	0.2474358974359	0.23510096153846	0.23510096153846	1.4119217081851
24	0.23510638297872	0.18974822695035	0.17800177304965	1.2
15	0.16602564102564	0.13166346153846	0.11314262820513	1.1442564559216
10	0.15337620578778	0.11438585209003	0.1117540192926	0.95462633451957

Table 1: Results for task1

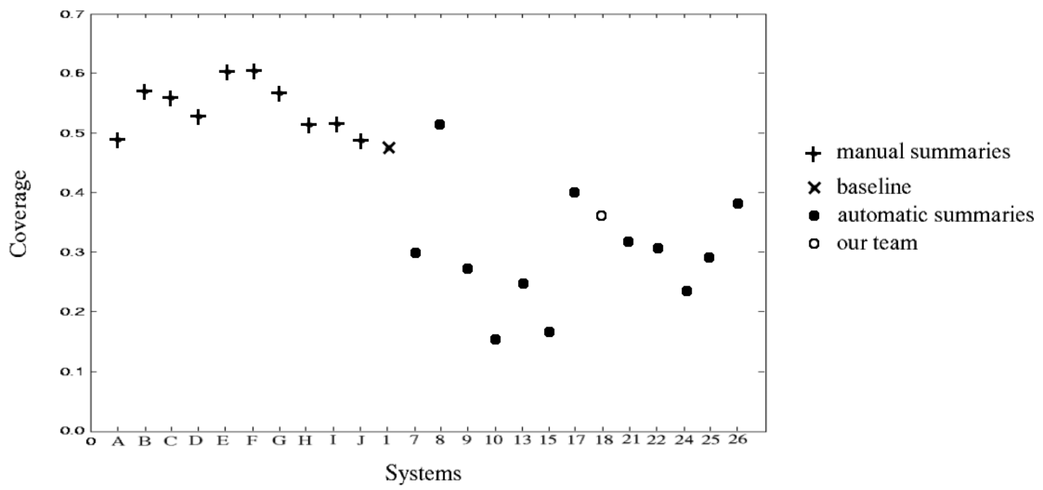

Figure 5: Results for task1 - coverage plot

Summarizer code code	Mean coverage	First mean length-adjusted coverage	Final mean length-adjusted coverage	Mean qq with non-zero answers
D	0.44566666666667	0.29755555555556	0.28777777777778	0.38888888888889
C	0.395	0.26322222222222	0.25511111111111	0.11111111111111
B	0.38144444444444	0.25411111111111	0.24622222222222	0.11111111111111
A	0.34155555555556	0.22755555555556	0.22088888888889	0
E	0.324	0.21688888888889	0.20855555555556	0.22222222222222
H	0.32288888888889	0.21522222222222	0.19333333333333	0.77777777777778
F	0.32111111111111	0.21477777777778	0.20122222222222	0.33333333333333
G	0.31044444444444	0.20733333333333	0.20477777777778	0.44444444444444
I	0.30366666666667	0.20288888888889	0.19555555555556	0.11111111111111
J	0.27022222222222	0.18022222222222	0.17866666666667	0.11111111111111
20	0.19863333333333	0.13406666666667	0.1233	1.0361
16	0.19266666666667	0.13043333333333	0.12063333333333	0.94833333333333
17	0.18996666666667	0.12666666666667	0.11063333333333	0.73333333333333
3	0.17153333333333	0.1158	0.10136666666667	1.1847
18	**0.1697 (4)**	**0.1255 (6)**	**0.1225 (4)**	**0.88333333333333 (5)**
22	0.1662	0.1282	0.1282	0.76666666666667
10	0.16556666666667	0.12836666666667	0.1283	0.74443333333333
23	0.15693333333333	0.11443333333333	0.11223333333333	1.0305666666667
11	0.15183333333333	0.11516666666667	0.11516666666667	0.7
21	0.14	0.099	0.0985	0.94643333333333
2	0.1299	0.088066666666667	0.079	0.9111
13	0.12873333333333	0.086533333333333	0.084766666666667	0.98333333333333
15	0.10196666666667	0.071266666666667	0.071266666666667	1.2693

Table 2: Results for task3

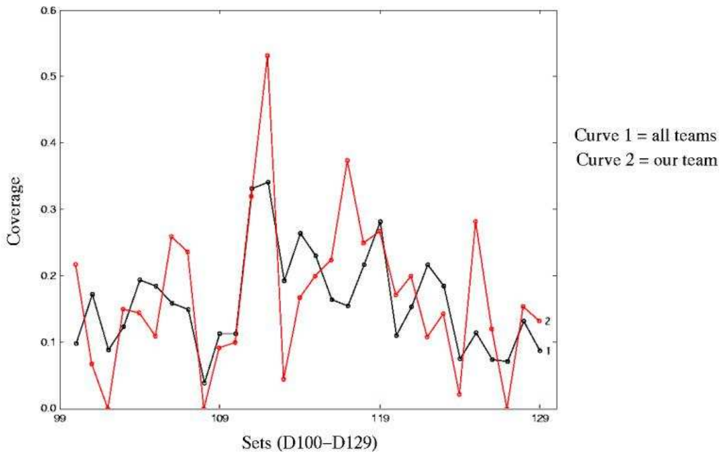

Figure 6: Task 3: plot of the mean coverage for our team (curve 2) comparing with all teams (curve 1) per set of documents

discusses the Hedge Trimmer presented at DUC 2003. Knight and Marcu use a training set of text/abstract pairs and train a classifier to find the best compression of a parsed sentence by separately using a statistical noisy-channel model and induction of decision rules to compress the parse tree. Hedge Trimmer - like our work - uses linguistically motivated heuristics to compress the parse tree. The difference of our work with Hedge Trimmer is that we select important sentences from the document as those that contain topic terms with the largest coverage as computed by our topic segmentation algorithm. Hedge Trimmer always uses the first sentence of the document. In both the second and the third task we use clustering techniques only for eliminating redundant content from a set of extracted sentences that are selected with our topic segmentation algorithm (second and third task) and viewpoint term overlap (third task). When clustering is used in text summarization, usually all sentences of the documents are clustered which does not always yield good results in multi-document summarization. Many text summarization algorithms require a training set of text/abstract pairs. Our algorithms do not need any training making them portable to many texts (e.g., for use on the World Wide Web) taking into account the constraints of the topic segmentation algorithm (i.e. texts are written in a SVO - subject-verb-object - language and are sufficiently coherent) and also the availability of a parser for sentence reduction.

8 Conclusions

We have presented our summarization system that was used for DUC 2003. We have shown the effectiveness of constructing headlines by first reducing the document to a number important sentences and then further reducing those sentences to their most important constituents by using syntactic parsing. We obtained good results both for user-oriented multi-document summarization and for generic multi-document summarization. Our techniques can be easily expanded to other SVO languages; we plan to try them on a Dutch corpus in the near future. We do not have a learning component, circumventing thus the need of large amount of training data.

References

[1] Angheluta R, De Busser R and Moens MF (2002) The Use of Topic Segmentation for Automatic Summarization. In Proceedings of the Workshop on Automatic Summarization, Philadelphia, Pennsylvania, USA, July 11-12, 2002. pp. 66-70.

[2] Charniak E (2000) A Maximum-Entropy-Inspired Parser. In Proceedings of NAACL, 2000.

[3] *Document Understanding Conference*, http://www-nlpir.nist.gov/projects/duc/ (visited 23.09.2002).

[4] Dunning T (1993) Accurate Methods for the Statistics of Surprise and Coincidence. Computational Linguistics, 19:61-74.

[5] Kaufman L, Rousseeuw P (1990) Finding Groups in Data, An Introduction to Cluster Analysis. A Wiley-Interscience Publication, John Wiley & Sons, Inc., New York.

[6] Knight K & Marcu D (2001). *Statistical-Based Summarization Step One: Sentence Compression (2000).* In Proceedings of AAAI-2001.

[7] Moens MF & De Busser R (2001). *Generic Topic Segmentation of Document Texts.* In Proceedings of the 24'th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, New York, pp. 418-419.

[8] Lin D (1998). *Automatic Retrieval and Clustering of Similar Words.* COLING-ACL98, Montreal, Canada, August, 1998.