The LIA-Thales summarization system at DUC-2006

Benoit Favre[1,2], Frederic Bechet[2], Patrice Bellot[2], Florian Boudin[2],
Marc El-Beze[2], Laurent Gillard[2], Guy Lapalme[3], Juan-Manuel Torres-Moreno[2]
[1] THALES, MMP Laboratory, Colombes, France
[2] LIA, University of Avignon, France
[3] RALI, University of Montreal, Canada
{benoit.favre,frederic.bechet,patrice.bellot}@univ-avignon.fr
{florian.boudin,marc.elBeze,laurent.gillard,juan-manuel.torres}@univ-avignon.fr
lapalme@iro.umontreal.ca

Abstract

The LIA-Thales system is made of five different sentence selection systems and a fusion module. Among the five sentence selection systems used, two were originally developed for the Question-Answering task (QA) and three specifically built for DUC-2006. The outputs of the five systems are combined in a weighted graph where the cost functions integrate the votes given by the different systems to the sentences. The best path in this graph corresponds to the summary given by our system. Our experiments have shown that the fusion of the five systems always scores better on ROUGE and BE than each system alone. In the DUC-2006 evaluation, the LIA-Thales fusion system obtained very good results in the automatic evaluations and achieved good performance in human evaluations.

1 Introduction

The main originality of the LIA-Thales system is its use of a fusion process for combining the outputs of five summarization systems developed by our team and based on widely different sentence selection algorithms. Similar fusion processes have been shown to outperform the best system alone in other domains such as Automatic Speech Recognition or Speaker Recognition.

Among the five sentence selection systems used, three were originally developed for the Question-Answering task (QA). These systems use different similarity measures (essentially based on term-frequency measures of n-grams of word, lemma or stem) between the topics, considered as questions, and the documents. The last two have been developed as summarization systems: one using several metrics (e.g. Hamming measure, Gaussian-sigmoid functions) and a decision algorithm and another one implementing an MMR and LSA space approach.

After presenting these five systems, section 3 introduces the fusion process, section 4 describes the linguistic processing implemented and section 5 gives an overview of our results.

2 Presentation of the five sentence selection systems

2.1 System 1 (*S1*): the MMR over LSA system

Maximal Marginal Relevance (MMR) is a sentence selection approach that views the summarization process as an optimization problem: the maximization of coverage and the minimization of redundancy of selected information. Each iteration of this greedy algorithm aggregates to the solution the sentence that is most relevant (close) to the topic while being the furthest from the already selected sentences. This approach has been used successfully in (Goldstein et al., 2000).

Recently, (Murray et al., 2005) proposed computing MMR inter-sentence similarities using a Latent Semantic Space (LSA) hoping to reduce the poor modeling of sentence semantics by standard Vector Space Models. The infomap-nlp[1] toolkit is used to produce a reduced co-occurrence matrix of words from big text corpora.

Co-occurrence window	+/- 15 words
Vocabulary size	60k most frequent words
Anchor vocabulary size	3k most frequent words
Corpus	corresponding DUC year
Dimensions	200

Table 1: Infomap-nlp model parameters

Our implementation of MMR over LSA adds a few improvements : context modeling, topic interpolation and similarity normalization.

[1] http://infomap-nlp.sourceforge.net

Bag-of-words models fail to model entities referenced by pronouns in a sentence. We implemented a simple context modeling approach to avoid the cost of a full coreference resolution. The current sentence vector is blended with the previous sentence vector to create a short term memory of semantic context.

$$\vec{s_i} = (1-\alpha)\vec{s_i} + \alpha\vec{s_{i-1}} \quad (1)$$

We first used DUC topics in MMR to find the most relevant sentences in an iteration. Further experiments showed that interpolating DUC topics with document cluster centroids improved significantly the ROUGE scores on the 2005 development datasets.

$$\vec{t} = (1-\beta)\vec{q} + \beta\frac{1}{N}\sum_{i=1}^{N}\vec{s_i} \quad (2)$$

During an MMR iteration, a sentence gets its score from two quantities (similarity with the topic and dissimilarity with the previous selection) mixed using the hyper parameter $lambda$. Normalizing the similarity distributions ($\mu = 0$, $\sigma = 1$) before mixing them lead to improvements in ROUGE scores.

MMR lambda	$\lambda = 0.95$
Context blending factor	$\alpha = 0.05$
Query interpolation factor	$\beta = 0.9$

Table 2: Optimal parameters according to ROUGE-2-R and ROUGE-SU4-R, on the DUC2005 dataset

We observe in table 2 that according to ROUGE, non-redundancy capabilities of MMR get a low weight and that document cluster centroids tend to overweight topic descriptions.

Several other approaches were tested but as they did not meet our expectations they were not included in the submission :

- Wordnet based query expansion introduced too much noise when synsets are not disambiguated.
- Gigaword based LSA models that are too general at our level of LSA dimensionality.
- LSA corpus gathering using document expansion with the MG engine on the Gigaword corpus.
- Different Tf-Idf schemes for word weighting that never outperformed weightless LSA.

2.2 System 2: (*S2*) the CORTEX modified system

***CO**ndensation et **R**ésumés de **T**extes* (CORTEX) (Torres-Moreno et al., 2005) is a single-document extract summarization system using an optimal decision algorithm that combines several metrics. These metrics result from processing statistical and informational algorithms on the document vector space representation.

The DUC 2006 evaluation task is a real-world complex question (called topic) answering, in which the answer is a summary constructed from a set of relevant documents. The idea is to represent the text in an appropriate vectorial space and apply numeric processes to it. In order to reduce complexity, a preprocessing is performed to the topic and the document: words are filtered, lemmatized and stemmed. The CORTEX system can use up to $\Gamma = 11$ metrics (Torres-Moreno et al., 2002) to evaluate the sentence's relevance. We have tested empirically a wide range of combinations and finally choose 3 metrics:

- The angle between the topic and the sentence vector.
- The sum of Hamming weights of words per segment times the number of different words in a sentence.
- The sum of Hamming weights of the words multiplied by word frequencies.

The last two metrics use the Hamming matrix H, a square matrix $N_L \times N_L$, in which every value $H[i,j]$ represents the number of sentences in which exactly one of the terms i or j is present.

The system scores each sentence with a decision algorithm which relies on the normalized metrics. Two averages are calculated, a positive $\lambda_s > 0.5$ and a negative $\lambda_s < 0.5$ tendency (the case $\lambda_s = 0.5$ is ignored). The following algorithm combines the vote of each metric:

$$\sum^{s}\alpha = \sum_{v=1}^{\Gamma}(||\lambda_s^v|| - 0.5);\ \ ||\lambda_s^v|| > 0.5$$
$$\sum^{s}\beta = \sum_{v=1}^{\Gamma}(0.5 - ||\lambda_s^v||);\ \ ||\lambda_s^v|| < 0.5$$

Γ is the number of metrics and v is the index of the metrics. The value given to each sentence s is calculated with:

$$if(\sum^{s}\alpha > \sum^{s}\beta)$$

then $Score_s^{cortex} = 0.5 + \sum^s \alpha/\Gamma$
else $Score_s^{cortex} = 0.5 - \sum^s \beta/\Gamma$

We have adapted CORTEX to a user-oriented multi-document summarization system by introducing two new parameters: the topic-document similarity and the topic-sentence overlap. The CORTEX system is applied to each document of a topic set and the summary is generated by concatenating higher score sentences.

We have improved the system with the implementation of a sigmoid based smoothing algorithm. This smoothing

process updates the sentence scores according to the average sentence length.

The topics are parsed to create sub-topics composed of the title and one of the topic's narration sentences. For each document in the topic set, N documents to be handled by CORTEX are created, N being the number of sub-topics.

The similarity measure (Salton, 1989) allows us to rescale the sentence scores according to the relevance of the document from which they are extracted. This measure is the normalized scalar product of the Tf-Idf vectorial representations ($\vec{v_d}$, $\vec{w_t}$) of the document d and the topic t.

$$Similarity(t,d) = \frac{\sum \vec{v_d}.\vec{w_t}}{\sqrt{\sum \vec{v_d}^2 + \sum \vec{w_t}^2}}$$

The overlap assigns a higher ranking for the sentences containing topic words and makes selected sentences more relevant. The overlap is defined as the normalized cardinality of the intersection between the topic word set T and the sentence word set S.

$$Overlap(T,S) = \frac{card(S \cap T)}{card(T)}$$

The final score of a sentence s from a document d and a topic t is the following:

$$Score = \alpha_1 \, Score_{s,d}^{cortex} + \alpha_2 \, Overlap_{s,t} + \alpha_3 \, Similarity_{d,t};$$

with $\sum_i \alpha_i = 1$ and $\alpha_1 = 0.54$, $\alpha_2 = 0.36$, $\alpha_3 = 0.10$.

2.3 System 3 (*S3*): an n-term model allowing variable length insertion

This system relies on the simple idea that a term sequence found in a topic may be encountered in a document with some other words between the term members. By word term, we also mean *inflected forms*, lemmas or stems. The example of table 3 is extracted from the DUC'05 development corpus (topic d383j).

INFF	POS	LEMMA	STEM
what	WP	what	
drugs	**NNS**	**drug**	**drug**
are	VBP	be	
used	VVN	use	
to	TO	to	
treat	**VV**	**treat**	**treat**
what	WP	what	
mental	**JJ**	**mental**	**mental**
illness	**NN**	**illness**	**ill**

Table 3: Example of system input extracted from the DUC'05 development corpus (topic d383j)

Since a stop list is applied in order to keep only the content words, three patterns are extracted from the current topic.

Type	Patterns
Inflected forms	drugs.* treat.* mental.* illness.*
Lemmas	drug.* treat.* mental.* illness.*
Stems	drug.* treat.* mental.* ill.*

Table 4: Example of patterns

Each pattern is then added to the corresponding model: in this case, a 4-gram, 4-lemma and a 4-stem. More generally we obtain at the end of the extraction process, three different models: the n-gram (noted g), the n-lemma (noted l) and the n-stem (noted s) ones with $n \leq 6$. These three models have been combined with two other scores:

- a coverage rate (noted c) computed as the ratio of the topic vocabulary found at least once in a segment;
- a model (noted r) defined as the inverse of the segment position in the file, relying on the assumption that sentences at the beginning of a text are more likely to appear in the summary than the ones at the end.

For each segment, a score is computed as the weighted sum of the five scores. The coefficients of this linear combination have been manually optimized on the DUC'05 dataset. This approach happened to be quite robust since the results estimated with the Rouge measure were still better on the test data (i.e. DUC'06 dataset) as shown in figure 1.

2.4 System 4 (*S4*): the passage retrieval component of the LIA QA system

Question Answering systems aim at retrieving precise answers to questions expressed in natural language. Questions processed are mainly factual questions and answers are pieces of text extracted from a collection (such as newspaper articles compilation). They have been particularly studied since 1999 and the first large scale QA evaluation campaign held as a track of the Text REtrieval Conference (Voorhees and Harman, 2005).

A typical QA system architecture involves at least these main components (most often pipelined):

- Question Analysis, to extract semantic type(s) of the expected answer;
- Document Retrieval to restrict the amount of processed data by further components;
- Passage Retrieval to choose the best answering passages from documents;

- and final Answer Extraction Strategies to determine the best answer candidate(s) drawn from the previously selected passages.

The last two components have been used for DUC 2006: System 4 (*S4*) presented in this section is based on the Passage Retrieval module of the LIA QA system; System 5 (*S5*) is based on the Answer Extraction module of the same system.

Passage retrieval can be seen as a kind of summary processing by filtering document passages according to a topic. Applied to DUC 2006 data, the inputs are the DUC topics (title+description) and the sets of documents; the outputs are ordered lists of retrieved sentences.

Since our first TREC QA participation (Bellot et al., 2003), our passage retrieval approach changed from a cosine based similarity to a density measure. For QA, our passage retrieval component considers a question as a set of several kinds of items : words, lemmas, POS tags, Named Entity tags, and expected answer types. For DUC 2006, items are the lemmas of the topics (empty words are filtered according to their POS tags) and the maximum size of a retrieved passage is limited to one sentence.

First, a density score s is computed for each occurrence o_w of each topic lemma w in a given document d. This score measures how far are the words of the topic from the other words of the document. This process focuses on areas where the words of the topic are most frequent. It takes into account the number of different lemmas $|w|$ in the topic, the number of topic lemmas $|w, d|$ occurring in the document d and a distance $\mu(o_w)$ that represents the average number of words from o_w to the other topic lemmas in d (in case of multiple occurrences of a lemma, only the nearest occurrence to o_w is considered).

Let $s(o_w, d)$ be the density score of o_w in document d:

$$s(o_w, d) = \frac{\log\left[\mu(o_w) + (|w| - |w, d|)\,.p\right]}{|w|}$$

where p is an empirically fixed penalty. The score of each sentence S is the maximum density score of the topic lemmas it contains:

$$s(S, d) = \max_{o_w \in S} s(o_w, d)$$

Sentences from the topic document set are ranked according to their scores.

2.5 System 5 (*S5*): using a QA answer extraction metric for summarization

This component is built from the answer extraction method we developed for our Question-Answering System (QAS) (Gillard et al., 2005). It has been applied without tuning it to the summarization task or to DUC data.

For our DUC'06 experiments, a $QSet$ is defined from a topic seen as a bag-of-words, lemmatized and stop-listed. Each sentence of a document is considered as an interesting passage. Unlike the QA task no semantic answer type is associated to the sentences, therefore we consider each of the item inside the $QSet$ as a possible answer candidate, and compute a density measure (called *compacity*) of the other items drawn from the $QSet$ around it. The best *compacity* score give us a centered window of an interesting subpart of the sentence, and this score is extrapolated to the one of the sentence.

The assumption behind our *compacity* score is that the best candidate answer is closely surrounded by the important words of the question. Any word not seen in the question can disturb the relation between a candidate answer and its responsiveness to a question. In QA, term frequencies are not as useful as for Document Retrieval: an answer word can appear only once, and it is not guaranteed that words of the question will be repeated in the passage, particularly in the sentence containing the answer. A score improvement can come from using an Inverse Document Frequency to further take into account a variation of distance coming from a non QSet word.

For each $x_i \in QSet$, compacity score is computed as follow:

$$compacity(x_i) = \frac{\sum\limits_{\substack{y \in QSet \\ y \neq x_i}} p_{y_m, x_i}}{|QSet|}$$

with y_m being the occurrence of y in the sentence which maximize:

$$p_{y_m, x_i} = \frac{|W|}{2R + 1}$$

and where:

$$\begin{aligned} R &= distance(y_m, x_i) \\ W &= \{z | z \in QSet, \text{distance}\,(z, x_i) \leq R\} \end{aligned}$$

Also, based on DUC'05 data and automatic evaluation measures, we consider compound words found in Topics for inclusion in the $QSet$ rather than their constituents, however no noticeable improvement was found. Similarly, using stems was not significant. Therefore we use single word lemmas as an elementary unit of our $QSet$. The *compacity* measure is computed by searching for the one with the best contribution y_m rather than the nearest occurrence of y, as previously done in our QAS.

While this approach seems to be the simplest of the five systems, that it is not tuned nor to the task nor to the data and that it obtains the lowest results on the ROUGE

metrics, *compacity* seems better that more complex approaches: QAS is the component with the best agreement with the 4 other systems as it chooses the largest number of sentences finally selected during the voting mechanism (around 23%). It can thus be used as a baseline for selecting important sentences while other systems specialized for more difficult extractions.

3 Fusion strategy

Fusion processes have been shown to outperform the best system alone in several domains such as Automatic Speech Recognition (ASR) or Speaker Recognition. For example, the ROVER (Fiscus, 1997) method used in ASR consists in aligning the automatic transcriptions of several speech-to-text systems in order to perform a vote among the different hypotheses obtained. Similarly we wanted to develop an alignment and voting method dedicated to process the output of several sentence selection systems. The rational behind this work is the following: because of the availability of the DUC-2005 data, one can develop a summarization system by training it on this data in order to improve as much as possible the ROUGE scores. However, because of the limited size of this corpus, there is a high overfitting risk for the models. Therefore by using several systems with very different sentence selection algorithms, some heavily tuned on the DUC-2005 corpus and some taken *out of the shelf*, this overfitting risk is reduced and the robustness of our summarization system can be increased. The fusion strategy developed at the LIA is described in the next section.

3.1 Summary selection as a best-path search problem

First all DUC documents are preprocessed and split into sentences, each associated with a unique identifier. Secondly, for each topic to process, the 5 sentence selection systems described in the previous section return a list of sentence IDs ordered by relevance towards the topic. The maximum size of this list is limited to 30 candidates. These 5 ordered lists of sentences are compiled into a single a Finite State Transducer (FST) by means of the following process:

- The 5 lists are merged and only one occurrence of each sentence is kept.
- Each sentence is represented by an FST accepting words and outputting the sentence ID.
- All these FSTs are concatenated into a single FST with empty epsilon transitions allowing jumping over any sentence.
- In order to keep only paths leading to summaries of about 250 words, we build a FST of 251 states and 250 transitions, each transition accepting all the words of the DUC documents. Only the last 20 states are final states, therefore by performing an intersection process between this FST and the previous one, we obtain an FST made of paths leading to summaries made of 230 to 250 words.
- Finally this FST is weighted according to the following cost function.

Three different costs are defined, at the sentence level, at the word level, and on the final states. The cost function associated with a sentence is made of two features: the vote between systems (i.e. the number of systems that have selected the sentence in their top 30 hypotheses) and the best rank obtained by the sentence in the 5 hypothesis lists. Some weights are also given at the word level in order to penalize sentences containing some particular features. For example, because no anaphora resolution module is included in our system we chose to penalize sentences containing personal pronouns. Finally, a cost is associated to each final state of the FST: this cost is set to zero for the final states leading to summaries of exactly 250 words and it increases linearly as the size of the summaries gets shorter.

The last step in the fusion process is to obtain the lowest cost path on the resulting FST. This path corresponds to the best set of sentences, according to the cost functions, leading to a summary whose size is as close as possible to 250 words. All the weights of the different cost functions have been tuned on the DUC-2005 corpus in order to maximize the ROUGE scores. All the operations on the FST have been made thanks to the AT&T FSM library (Mohri et al., 1997).

Once the set of sentences is selected, an ordering and structuring process is performed. It is presented in the following section.

3.2 Sentence ordering and structuring

Three partial orders are used for sorting the set of sentences obtained from the FST:

1. sentence order within a document;
2. temporal order of the documents (all the sentences of a document are labeled with the date of the document);
3. geographical order of the documents (all the sentences of a document are labeled with the geographical origin of the document).

The first one is always applied for sorting sentences belonging to the same document. The last two can be considered as rough heuristics, as it is obvious that a sentence is not necessarily characteristic of the location and

the date of creation of the document it belongs to. However, these partial orders are used in the following way: firstly each topic is labeled with four tags: *General*, *Specific*, *Temporal*, *Geographical*, given by simple rules developed on the DUC-2005 corpus. Here is an example of such rules:
if the topic description does not contain any proper name, then set ***general*** *to* ***true***.
if there is an occurrence of a list of words such as 'world', 'country', 'nation',... then set ***Geographical*** *to* ***true***.

If the tag *temporal* is given, the temporal order is used first. If the tag *Geographical* is given it's the geographical order which is used first. If no tag is given, the temporal order is assigned by default.

A paragraph break (empty line) is added each time the year or the location, according to the partial order chosen, is modified. The remaining partial order is then applied to each paragraph.

The last process consists of adding an explicit reference to the year (e.g. *In 2002*) or the geographical location (e.g. *In Brazil*) of the document at the beginning of each paragraph, if the tags *temporal* and/or *Geographical* have been set to true. If both tags have been given, the *temporal* reference is preferred.

4 Linguistic post-processing

Our rule based linguistic post-processing targeted sentence length reduction and coherency maximisation. The process included the following steps and tried to minimize the linguistic risk of taking wrong decisions :

- Acronym rewriting: we replaced the first occurrence of an acronym by its definition and ensured that the acronym was used in the rest of the summary instead of the long form. The definitions were mined in the DUC corpus as parenthesized upper-case letters after an aligned capitalized word sequence. We tried to extend this strategy to the gigaword corpus and to use Google queries for unknown acronym resolution but the many erroneous resolutions motivated us not to use these extensions for the final run. Nevertheless the readability improvement by acronym rewriting seemed significant for acronym based topics.

- Person name rewriting: a similar approach was used to rewrite person names using only their family name except for their first occurrence. Person name mining in the DUC corpus involved confidence levels based on frequency and presence of job titles. Again, this proved useful to improve readability of person centered topics.

- We also implemented a reformatting of numbers and dates, removal of link words, person titles, say clauses and a few temporal references. For all of these, only the less risky rules were kept for the final run.

- Finally, duplicated sentences (bringing no new words to the summary) where skipped, punctuation cleaned (parenthesised content removal...) and glued to words.

corpus	pre-processing	post-processing
2005	260.50	249.26
2006	259.0	249.22

Table 5: Average summary length with and without reduction using linguistic processing

5 Results

Figure 1 shows the ROUGE scores obtained by our 5 systems on DUC 2005 and 2006 data. Two fusion results are also displayed: *F1* that corresponds to the fusion of the three systems that have been tuned on DUC 2005, *S1*, *S2* and *S3*, and *F2* that corresponds to the fusion of all the five systems. Two main comments can be made on these results:

- the improvements obtained by tuning the systems *S1*, *S2* and *S3* on DUC 2005 data apply also to DUC 2006: if, before tuning, all our five systems obtained comparable scores on DUC-2005 data, the three systems with tuning achieved much better ROUGE performance than the other two (*S4*, *S5*);

- the fusion process always improve the scores over the best system alone.

Although *F1* of the top 3 systems obtains better results on the 2005 data, *F2* is better on 2006. This validates our intuition that a fusion process with very different systems is a good strategy for preventing overfitting on the training corpus.

vote	2 syst.	3 syst.	4 syst.	5 syst
2005	100%	72.7%	35.6%	7.4%
2006	100%	79.4%	44.5%	12.4%

Table 6: % of sentences from the fusion summaries obtained on DUC 2005 and 2006 data that have been voted by 2, 3, 4 or 5 systems

Tables 6 and 7 describes the content of the summaries obtained with the fusion *F2*. More than 70% of the sentences part of the summaries produced have been previously chosen by at least 3 systems, and all of them have been chosen by at least 2 systems. When looking at the distribution of the sentences among the different systems,

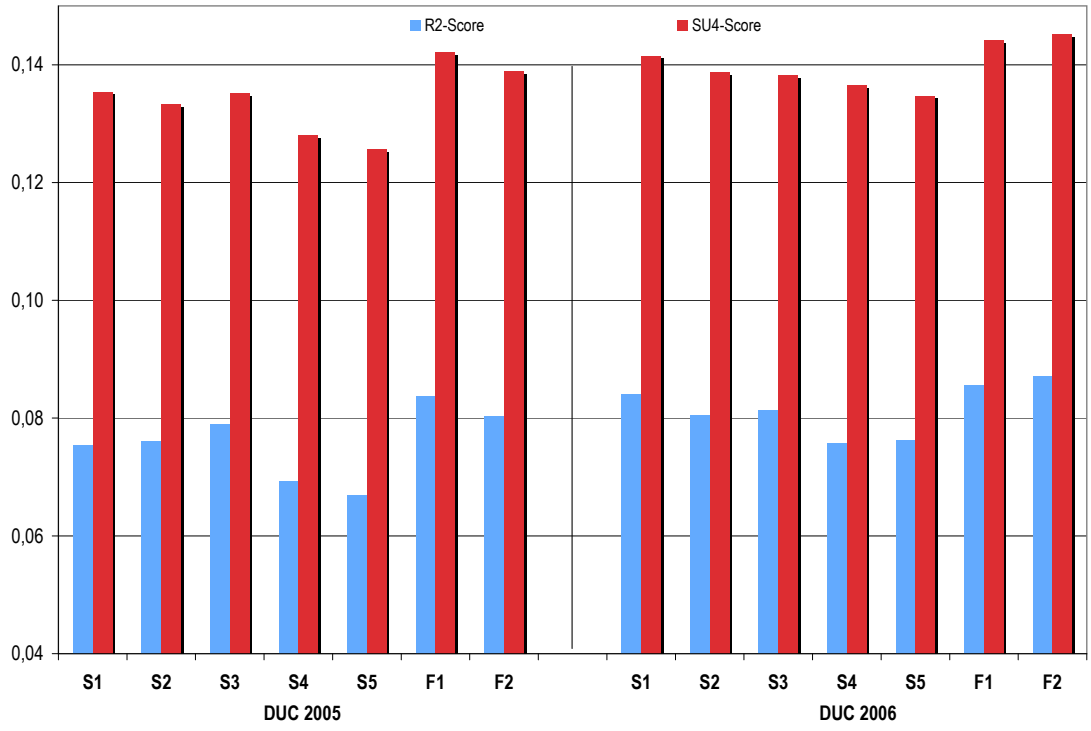

Figure 1: Recall ROUGE results, ROUGE2 and SU4, for the 5 systems, the two fusions (fusion 1 is the fusion of the systems 1,2 and 3; fusion 2 is the fusion of all the systems) on DUC 2005 and 2006 corpora

System	S1	S2	S3	S4	S5
2005	15%	20.7%	23%	17.4%	23.9%
2006	16.5%	21.2%	22.8%	16.8%	22.7%

Table 7: Importance of each system choice in the final summaries produced after the fusion process

one can see that each system participates to the final result, with proportions varying from 15 to 23%. Interestingly, the system that obtains the best results alone (*S1*) has the lowest contribution to the final summaries.

Another way to detail our results is to look at all the scores obtained according to the type of topic targeted. In table 8 the topics are automatically labeled, as presented in section 3.2, with the four tags *Specific*, *General*, *Temporal*, and *Geographical*. An *Unknown* tag is added if no *Temporal* or *Geographical* is given. As we can see, it seems that the best results are obtained on the *General* and *Geographical* topics. Adding explicit reference to locations at the beginning of each paragraph seems to have been useful for the *Geographical* topics. However, this didn't bring any improvement over the *Unknown* topics for the temporal reference added for the *Temporal* topics.

Finally tables 9 and 10 compares our results to the other systems participating to the DUC 2006 evaluation. As we can see, the LIA-Thales fusion system obtained very good results in the automatic evaluations (ranked in the top 6 systems, with only 2 systems significantly better). On the human evaluations our system achieved good performance (ranked 8 in *Resp-Overall* and *Resp-Content*). However, the poor *Structure and coherence* and *Non-redundancy* scores emphasize the need for more complex post-linguistic processes in order to compensate one of the weakness of the fusion strategy that tends to select the most *obvious* sentences, leading potentially to a high level of redundancy and a lack of structure in the summaries produced.

6 Conclusion

We have presented the LIA-Thales system based on the fusion process of five different sentence selection systems. Our experiments have shown that the fusion of the five systems always scores better on ROUGE and BE than each system alone. Moreover, the fusion with all

Manual scores					
Measure	**Specif.**	**Gen.**	**Temp.**	**Geo.**	**Unk.**
Gramm.	3.90	**4.30**	3.80	4.00	**4.25**
Non-redund.	3.52	**4.28**	3.81	**4.33**	3.75
Ref. clarity	**3.59**	3.19	**3.69**	3.33	3.28
Focus	**3.79**	3.66	3.37	**4.33**	3.82
Struct. coher.	2.14	**2.42**	2.00	**2.83**	2.28
Ling. Qual.	3.39	**3.58**	3.34	**3.77**	3.48
Resp Content	2.55	**3.09**	**3.06**	3.00	2.57
Resp Overall	2.38	**2.48**	2.37	**2.66**	2.39
Automatic scores					
Rouge-2 R	**0.090**	0.082	0.077	**0.095**	0.091
Rouge-SU4 R	**0.146**	0.144	0.136	**0.151**	0.149
BE	0.045	**0.051**	0.046	**0.056**	0.047

Table 8: Automatic and manual scores on DUC 2006 data according to the labels automatically given to the topics: *Specific* (Specif), *General* (Gen.), *Temporal*, *Geographical* (Geo) and *Unknown* (Unk)

Manual scores	rank
Grammaticality	7
Non-redundancy	31
Referential clarity	6
Focus	13
Structure and coher.	19
Ling Quality Mean	14
Resp-Content	8
Resp-Overall	8

Table 9: Rank of the LIA-Thales system at the DUC 2006 evaluation for the manual scores. For ROUGE and BE scores, the number of systems significantly better or worse is also precised

the five systems obtains better scores on DUC-2006 than the fusion with only the *best* three tuned systems, indicating that the fusion process is a good strategy for preventing overfitting on the training corpus. In the DUC-2006 evaluation, the LIA-Thales fusion system obtained very good results in the automatic evaluations (ranked 5th in SU4, 6th in ROUGE-2, 6th in BE and 6th in Pyramid) and achieved good performance in human evaluations (ranked 8th in the Resp-Overall).

Automatic scores	rank	# syst. better	# syst. worse
Pyramid-Score	6		
R2-Score	6	2	25
SU4-Score	5	1	26
BE-Score	6	2	26

Table 10: Rank of the LIA-Thales system at the DUC 2006 evaluation for all the automatic scores. For ROUGE and BE scores, the number of systems significantly better or worse is also precised

References

P. Bellot, E. Crestan, M. El-Bèze, L. Gillard, and C. de Loupy. 2003. Coupling named entity recognition, vector-space model and knowledge bases for trec-11 question-answering track. In *The Eleventh Text REtrieval Conference (TREC 2002), NIST Special Publication 500-251*.

J.G. Fiscus. 1997. A post-processing system to yield reduced word error rates: Rover. In *Proceedings of IEEE ASRU Workshop, Santa Barbara, USA*, pages 347–352.

L. Gillard, P. Bellot, and M. El-Bèze. 2005. *Le LIA à EQueR*. In *Actes de TALN-Recital 2005*, volume 2, pages 81–84.

J. Goldstein, V. Mittal, J. Carbonell, and M. Kantrowitz. 2000. Multi-document summarization by sentence extraction. In *ANLP/NAACL Workshop on Automatic Summarization*, page 4048.

Mehryar Mohri, Fernando Pereira, and Michael Riley. 1997. AT&T FSM Library - Finite State Machine Library. *AT&T Labs - Research*.

G. Murray, S. Renals, and J. Carletta. 2005. Extractive summarization of meeting recordings. In *Eurospeech 2005*.

G. Salton, 1989. *Automatic text processing*, chapter 9. Addison-Wesley Longman Publishing Co., Inc.

J.M. Torres-Moreno, P. Velazquez-Morales, and J.G. Meunier. 2002. *Condensés de textes par des méthodes numériques*. *JADT*, 2:723–734.

J.M. Torres-Moreno, P. Velazquez-Moralez, and J. Meunier. 2005. *CORTEX, un algorithme pour la condensation automatique de textes*. In *ARCo*, volume 2, page 365.

E.M. Voorhees and D. Harman, 2005. *TREC Experiment and Evaluation in Information Retrieval*, chapter 10, pages 233–257. MIT Press.