QASUM-TALP at DUC 2005 Automatically Evaluated with the Pyramid based Metric AutoPan

Maria Fuentes, Edgar Gonzàlez, Daniel Ferrés, Horacio Rodríguez
TALP Research Center, Software Departament
Universitat Politècnica de Catalunya
{*mfuentes, egonzalez, dferres, horacio*}*@lsi.upc.edu*

Abstract

This paper presents QASUM-TALP, a query-driven multidocument summarizer. We describe the techniques used to briefly synthesize a set of documents in 250 words. A first approach was evaluated in the context of our participation in the DUC 2005 task. The approach used is based on the creation of simpler questions for each topic title and topic narrative. These simpler questions are used by a QA system for extracting rellevant information. Finally, a Summary content selection process creates summaries using the relevant information and lexical chains. We analyze the results and report and fix problems in the execution process. Moreover, we present and use AutoPan a metric to automatically evaluate summarizers performance. This metric is based in Pyramid information manually created for DUC 2005 contest from human summaries at Columbia University. We show that AutoPan has a very good correlation to manual Pyramid scores.

1 Introduction

The QASUM-TALP system participated in the DUC 2005 task: to synthesize from a set of 25-50 documents a brief, well-organized, fluent answer to a need for information that cannot be met by just stating a name, date, quantity, etc. This task models real-world complex question answering. In DUC 2005, for each topic the following user information is given: title, narrative description, and information about the kind of expected summary (generic or specific). The size of the summary is fixed to 250 words.

We present a query-driven multidocument summarizer. Our approach uses extraction techniques to produce summaries, selecting and putting together several portions from the original set of documents. In brief, our approach is divided into four main steps. First, a set of queries is automatically generated from the complex description given by the user. Then TALP-QA, a multilingual open-domain Question Answering (QA) system, is used to detect those sentences with relevant information. The TALP-QA system is in continuous evolution, the last English prototype participated in TREC 2005 QA track (see [Ferrés et al, 05]). An earlier version took part in TREC 2004 QA track ([Ferrés et al, 04]). We have used this system without the specific Definitional QA subsystem. Moreover, only the initial steps of TALP-QA have been performed, as we are not interested in obtaining the factual answer of the questions but only to locate the sentences at where they is located.

The summary content is selected from the set of candidate sentences in the relevant passages. The semantic representation of the sentences is contrasted to avoid redundancy.

Automatic evaluation is helpful when checking the performance of several approaches or different versions of the same summarizer. We propose to use manually annotated pyramids to automatically evaluate new summaries, in the same direction as [Harnly et al, 05]. To apply our method it is necessary to have a pyramid (as defined in [Nenkova and Passonneau, 04]) from each set of documents to be summarized.

The rest of the paper is structured as follows. The next section presents the overall architecture of QASUM-TALP and briefly describes the main components. Section 3 presents an evaluation contrasting our participation at DUC (QASUM-UPC) with the new prototype (QASUM-TALP), where several failures were located and fixed. This new prototype is also automatically contrasted with two baselines. On the one hand, document summaries produced by the English version of the single document summarizer presented in [Fuentes and Rodríguez, 02] have been considered. On the other hand, the Definitional QA subsystem (see [Ferrés et al, 05]) is used as second baseline. We finish the paper with an analysis of the obtained results (Section 4), followed by an exposition of the extracted conclusions and pointers to future work (Section 5).

2 Process Overview

In our approach, to face the DUC 2005 task, summaries are produced from a complex question given by a user in four phases: Collection Pre-processing, Question Generation, Relevant Information Extraction and Summary Content Selection (see Figure 1). Each phase is briefly described below.

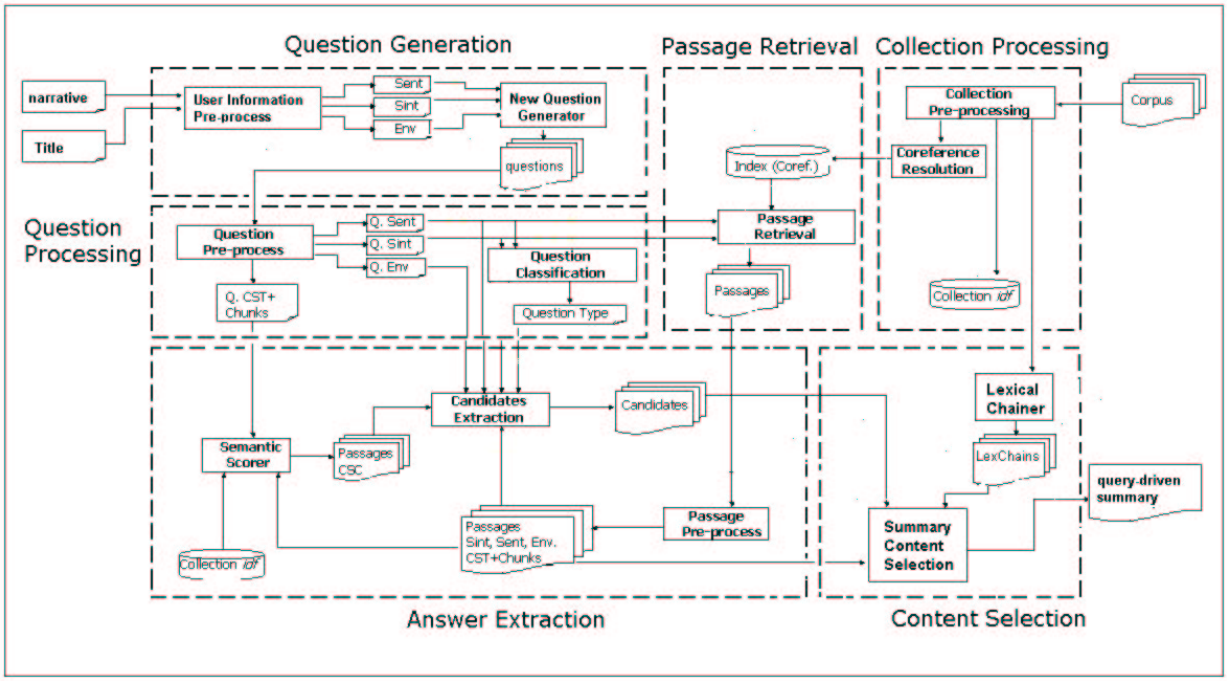

Figure 1: QASUM-TALP Architecture.

2.1 Collection Processing

Each set of documents is pre-processed with general purpose linguistic tools. Text is enriched with part-of-speech (POS) tag, lemma, Named Entity (NE) and syntactic information; and 3rd person pronoun co-reference is solved. These are the same tools used by the TALP-QA system.

This enriched collection has been indexed using *Lucene*[1] Information Retrieval engine. Lucene is used to create an index with two fields per document:

- the lemmatized text with NEs recognized and classified and syntactic information
- the original text (forms) with NEs recognized (not classified) and syntactic information.

As an additional knowledge an *idf* weight is computed at document level for the whole collection. This information is also used in the Relevant Information Detection phase.

2.2 Question Generation

The current TALP-QA system does not process complex questions. For that reason the original user information is previously transformed into a set of simpler factual questions. The Relevant Information Detection module (Section 2.3) processes those questions. Information from the corresponding candidate answers is used in the Summary Content Selection (Section 2.4).

To create a set of questions, the title and the narrative of a topic are pre-processed (as in Section 2.1). After that, each sentence from the narrative is considered as a question to be included in the set. Sentences containing locally conjoined elements with *and* or *or* are splitted into several ones, each containing an element of the conjunction. The title of the topic is used to create a question with the following pattern: "What is <title>?". Finally, a set of new questions is generated using semantic information extracted from the narrative. The title is attached at the end of each question to provide a context for them.

Following TALP-QA system format, pre-processed sentences from the narrative are represented by: *sent*, *sint*, and *environment*; roughly corresponding to the lexical, syntactic, and semantic levels:

- **Sent**, lexical information for each word: form, lemma, POS, semantic class of NE, list of WordNet synsets and, finally, whenever possible other complementary information, such as the verbs associated to the actor and the relations between locations and their gentile.
- **Sint**, syntactic constituent structure of the question (*head* specified for each constituent), and information about the relations between components (*subject*, *object*, and *indirect object*).
- **Environment**, semantic relations between different components identified in the question text. These relations are organized into an ontology of about 100 semantic classes and 25 relations (mostly binary), where classes and relations are connected by taxonomic links. The ontology tries to reflect what is needed for an appropriate representation of the semantic environment of the question (and

[1] http://jakarta.apache.org/lucene

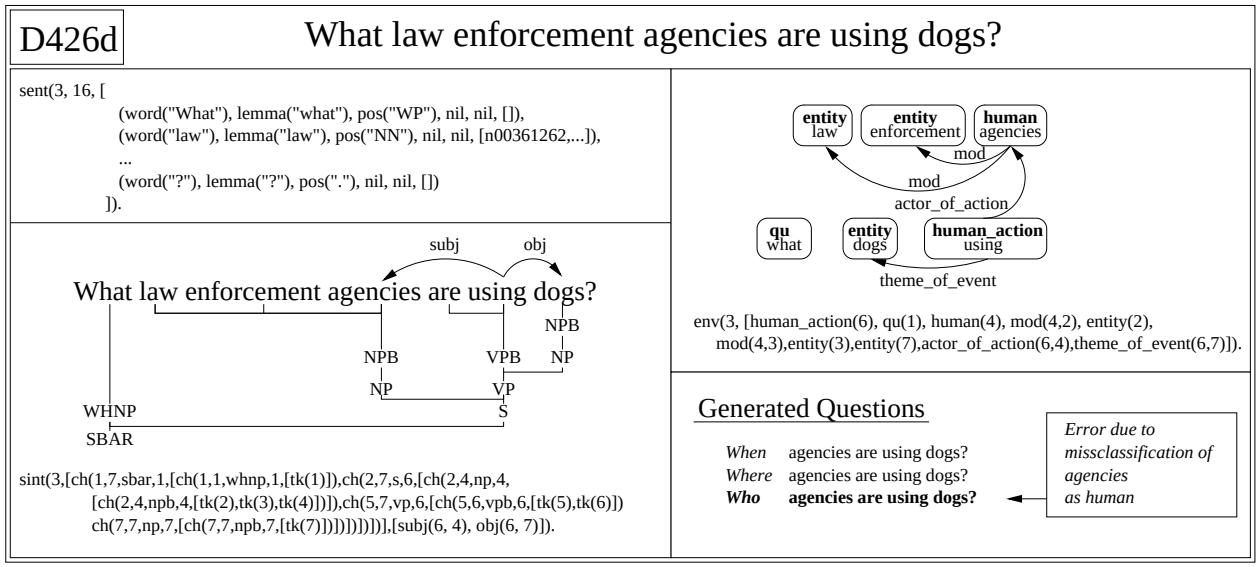

Figure 2: Sample of a pre-processed text.

the expected answer). For instance, *Action* is a class and *Human_action* is another class related to *Action* by means of an *is_a* relation. In the same way, *Human* is a subclass of *Entity*. *Actor_of_action* is a binary relation (between a *Human_action* and a *Human*). When a question is classified as *Who_action* an instance of the class *Human_action* has to be located in the question text and its referent is stored.

Questions are generated from NEs and Actions detected in the narrative. NE classification is taken into account to generate specific patterns (see Table 1). In Figure 2.2 we can see a sample of Questions generated from a part of the narrative. In this case *using* has been detected as an action. The new question is generated considering the chunk that contains both the subject *agencies* and the object of the action *dogs*.

LOCATION	PERSON
Where is <*NE*>?	Who is <*NE*>?
	When <*NE*> was born?
ORGANIZATION	Where <*NE*> was born?
Where is <*NE*>?	When <*NE*> died?
Where is <*NE*> located?	Where <*NE*> died?
When was <*NE*> founded?	When <*NE*> lived?
Who is <*NE*> director?	Where <*NE*> lived?
OTHERS	ACTION
Where is <*NE*>?	Where <*chunkAction*>?
When was <*NE*>?	When <*chunkAction*>?
Who did <*NE*>?	Who <*chunkAction*>?

Table 1: Question Generation patterns by NE type.

2.3 Relevant Information Detection

The relevant information detection is performed by the TALP-QA system ([Ferrés et al, 05]). The approach is based on in-depth NLP processing and a semantic information representation. The system has been adapted to deal with documents indexed with semantic information in order to improve system speed. On the other hand, its specific Definitional QA subsystem has not been used.

The TALP-QA system architecture has three subsystems that are executed sequentially without feedback:

First, questions are processed (QP); after that, for each question the passage retrieval (PR) component extracts pieces of text likely to content the answer; finally, candidate answers are extracted (AE). Below, each component is briefly described.

2.3.1 Question Processing

The main goal of this component is to detect the expected answer type and to generate the information needed in the following components. For PR, the information needed is basically lexical (POS and lemmas) and syntactic, and for AE, lexical, syntactic and semantic. To process questions and passages we use the tools and representation formalisms used to previously process complex questions (see Section 2.2).

2.3.2 Passage Retrieval

The main function of the passage retrieval component is to extract small text passages that are likely to contain the correct answer. Document retrieval is performed using the *Lucene* Information Retrieval system. Each keyword is assigned a priority using a series of heuristics. For example, a proper noun is assigned a priority higher than a common noun, the question focus word (e.g. "state" in the question "What state has the

most Indians?") is assigned the lowest priority, and stop words are removed.

The passage retrieval algorithm uses a data-driven query relaxation technique: if too few passages are retrieved, the query is relaxed first by increasing the accepted keyword proximity and then by discarding the keywords with the lowest priority. The contrary happens when too many passages are extracted.

2.3.3 Answer Extraction

After PR, two tasks are performed in sequence: Candidate Extraction (CE) and Answer Selection (AS). In the first component, all the candidate answers are extracted from the highest scoring sentences of the selected passages. In the second component the best answer is chosen.

2.4 Summary Content Selection

The sentences that will build the final summary are selected one at a time from the set of sentences retrieved in the Passage Retrieval phase. The sentences are selected in a greedy way, in the following order:

- Firstly, sentences containing answers to the generated questions are considered, sorted by their score.
- Next the other sentences are considered, at every step priorizing those that precede or follow a previously selected one.

At each step, the redundancy of the candidate sentence with respect to the previously selected ones is measured, taking into account the environments of all sentences. If this redundancy exceeds a threshold, the sentence is discarded. The redundancy measure currently used is the fraction of environment elements of the candidate sentence not present in the environments of the previously selected ones, with respect to the total size of the environment of the candidate. The threshold has been set to 0.5.

This redundancy measure may be modified by another factor, according to the specificity of the desired summary: in the case the summary is asked to be specific the similarity between NEs in the sentences is also taken into account. When the summary is required generic this similarity is not considered in the measure.

Sentences are added until the desired summary size is reached. If all retrieved sentences have been considered and the size has not been reached, sentences from a back-off summary, generated using lexical chain information, are incorporated into the summary to complete it. To create this back-off summary, previously lexical chains from each document are computed using the module described in [Fuentes et Rodríguez 02]. After that the first sentence of each document crossed by a lexical chain is taken, until the desired size is achieved.

3 Evaluation

In DUC 2001 to 2004, the manual evaluation was based on comparison with a single human-written model and a lot of the information of evaluated summaries (both human and automatic), was marked as "related to the topic, but not directly expressed in the model summary". The pyramid method ([Nenkova and Passonneau, 04]) addresses the problem by using multiple human summaries to create a gold-standard and by exploiting the frequency of information in the human summaries in order to assign importance to different facts.

However, the method of pyramids for evaluation requires a human annotator to match fragments of text in the system summaries to the Semantic Content Units (SCUs) in the pyramids. We have tried to automate this part of the process[2].

- The text in the SCU label and all its contributors is stemmed and stop words are removed, obtaining a set of stem vectors for each SCU. The system summary text is also stemmed and freed from stop words.
- A search for non-overlapping windows of text which can match SCUs is carried. A window and an SCU can match if a fraction higher than a threshold (experimentally set to 0.90) of the stems in the label or some of the contributors of the SCU are present in the window, without regarding order. Each match is scored taking into account the score of the SCU as well as the number of matching stems. The solution which globally maximizes the sum of scores of all matches is found using dynamic programming techniques.

The constituent annotations automatically produced are scored using the same metrics as for manual annotations, and it is found that there is statistical evidence supporting the hypothesis that the scores obtained by automatic annotations are correlated to the ones obtained by manual ones for the same system and summary. We apply a Spearman test to the scores obtained by every summary, including the human ones. In total, the data set consists of 540 samples. The test reports values of $r = 0.52$ for *Original pyramid score* and $r = 0.58$ for *Modified pyramid score*, which exceed the critical value for a confidence of 99%. If we repeat the Spearman test with only the automatic systems (500 samples) the values go down to $r = 0.49$ and $r = 0.52$, but they remain inside the 99% confidence level. Nevertheless, it should be noted that in both cases the scores of automatic annotations tend to be quite lower than those of manual ones.

We will refer hereforth to the *Original pyramid score* obtained from the automatic pyramids as **AutoPan1** and to the *Modified pyramid score* as **AutoPan2**. The scores obtained from manual pyramids will be referred to as **ManPan1** and **ManPan2**.

[2]Software is available at `http://www.lsi.upc.edu/~egonzalez/autopan.html`

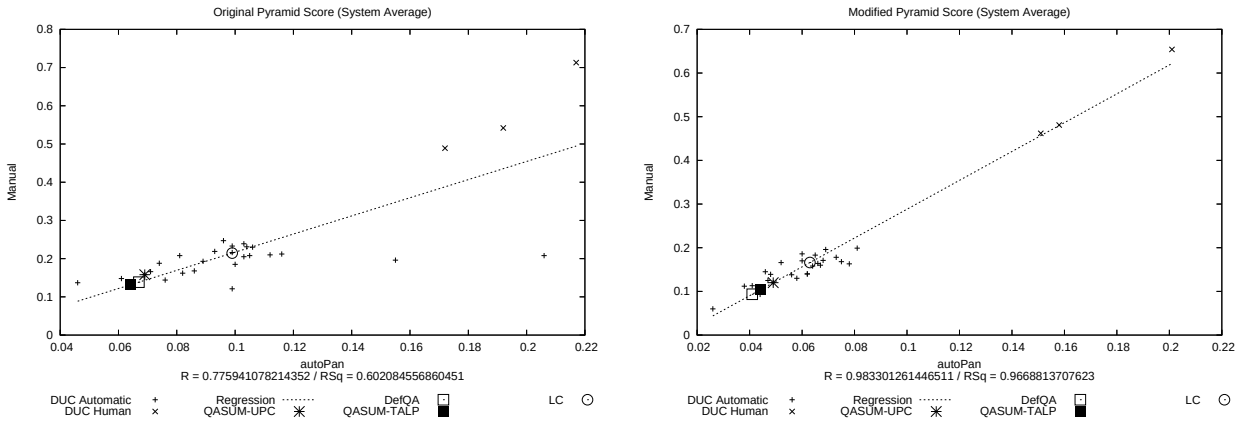

Figure 3: Scores obtained by manual and automatic constituent annotation of the DUC system summaries, averaged by system

If instead of considering the results summary by summary, we take the averages of the scores for each system, we obtain the plots depicted in Figure 3. We observe there seems to be linear dependency between the two variables. If we apply linear regression, we obtain a Pearson coefficient of 0.77 for **AutoPan1** and **ManPan1** and 0.98 for **AutoPan2** and **ManPan2**, with a data set size of 28 samples. In both cases, the highest scores correspond to human systems, which looks sensible. Only for the correlation between **AutoPan1** and **ManPan1** measures, the points on the right side of the plot diverge from the regression line.

The fact that, being a system which uses such shallow linguistic information (only stemming), the correlation with human judgment be significant is encouraging. However, we have observed that sometimes this system creates incorrect matches (some contributors require a context to convey the meaning of their associates SCU, and our system does not take context into account) and of course some matches are missed.

Once we found that the scores from the automatically constructed pyramids correlated with those from the manually constructed ones, we consider the correlation of these scores to the pyramid scores manually assigned, to the responsiveness measure, and to R2 and RSU4, the ROUGE metrics used to automatically evaluate systems at DUC 2005. We apply Spearman and Pearson tests to the average of the scores obtained in all clusters by every non-human system. The results are summarized in table 2.

On the one hand, the **AutoPan2** metric correlates well with all other metrics in both tests: the Pearson correlation test and the Spearman rank correlation. All values exceed the confidence level of 99%. On the other, **AutoPan1** correlates at 95% confidence level with the manual pyramid-based scores using both tests. The only exception is the 0.360 value for the Pearson correlation with **ManPan2**, which does not exceed the imposed 0.05 p-value. For the ROUGE measures and the Responsiveness, no correlation test achieves this confidence level.

One of the explanations to this non-correlation can be seen in figure 4, which describes the behavior of the four automatic measures when evaluating non-human systems. It can be seen that two systems are particularly wrongly scored by **AutoPan1**. They are systems 11 and 26, the same systems that are scored at the level of human summarizers in figure 3. In fact, system 11 is better scored than two humans. It seems that **AutoPan1** must have some sensitivity to the kind of automatic summary produced. However, the plots for **AutoPan2** and the ROUGE metrics have a similar shape, and this agrees with the correlation values seen in table 2.

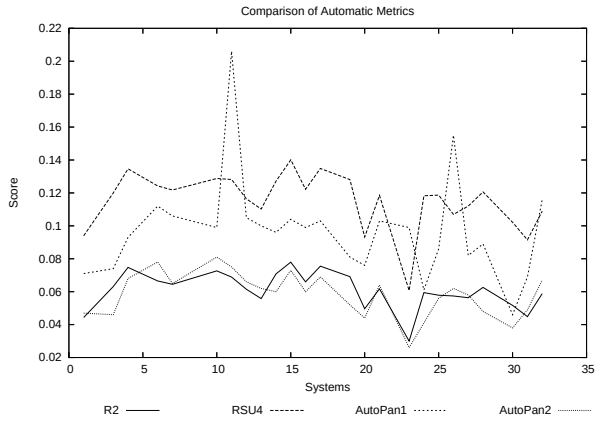

Figure 4: Comparison of Automatic Metrics for the DUC05 Submitted Systems

4 Analisys of the Results

After **QASUM-UPC** submission, we detected that in the preprocess step NEs were not correctly computed. NEs were not used in the index creation. A second run of the system, **QASUM-TALP**, has been carried out, considering also NEs in the in-

	Test	R2	RSU4	ManPan1	ManPan2	Resp
AutoPan1	Spearman	0.376	0.330	**0.582**	**0.548**	0.302
	Pearson	0.292	0.238	***0.432***	0.360	0.207
AutoPan2	Spearman	**0.683**	**0.665**	**0.802**	**0.802**	**0.649**
	Pearson	**0.755**	**0.725**	**0.820**	**0.851**	**0.699**

Table 2: Correlation values for several metrics, evaluating average scores of non-human systems. Values which exceed the p-value for $p = 0.01$ are shown in **bold**. Values which exceed the p-value for $p = 0.05$ are shown in ***bold italics***.

dexing process. Moreover, in the submitted system, NEs were badly segmented such as the case of **American_Tobacco_Companies_Overseas**. This was solved for the second run. Figure 3 shows how the best results are obtained by **LC** (the back-off summary), which reaches an average position with respect to the rest of systems in DUC. **DefQA** obtains slightly lower results than **QASUM-UPC**, the submitted system. **QASUM-TALP** is worse than the submitted system. The new systems have been evaluated using only the automatic procedure, and their scores are inferred from the found regression line.

As the approach presented is still a preliminary version, there can be multiple reasons for this decrease in the performance.

The previous section reports a black box evaluation. In order to explain why the first version performs better than the second one a glass box evaluation has been carried out. First of all it has been observed that 23 of the 50 submitted summaries were back-off summaries. As seen in Figure 3 the back-off summaries (see **LC**) are quite well scored, at least better than our own submitted system. Contrasting both approaches with the subset of 20 clusters with manual Pyramid information it has been seen that in **QASUM-UPC** 8 of the 20 summaries were from the **LC** while in **QASUM-TALP** was only 1. In order to check how well did the **TALP-QA** system extract relevant information, the summary sentence corpus proposed in [Copeck and Szpakowicz, 05] was used. This corpus was created taking into account information from manual evaluations done using the Pyramid method. This corpus has been aligned at our sentence segmentation level to check how well the Passage Retrieval module of the QA system extracts relevant sentences. Evaluated with data from TREC'05 this module obtains an accuracy performance of 62,60% (216/345) of questions for which an answer has been found in their set of passages (see [Ferrés et al, 05] for more details). In order to see how **TALP-QA** is affected by the fact that this system was dessigned to answer factual queation as defined in TREC'05 we study the Precission (P) and the Recall (R). Table 3 shows the results obtained

We detect that in the first approach titles were segmented as a single NE and were classified as a PERSON. In general much more questions were generated in the first approach than in the second one (see 4). For that reason more sentence were retrieved from each question in QASUM-UPC than in QASUM-TALP. Even though less sentences were extracted in the last version, their precision was better. That leads us to think that we are following a right direction and that future experimentation will bring better results.

Approach	P	R
QASUM-UPC	0.27	0.28
QASUM-TALP	0.37	0.24

Table 3: Precision and Recall obtained in selecting relevant sentences.

QASUM-UPC
Who is Robot Technology?
When Robot Technology was born?
Where Robot Technology was born?
When Robot Technology died?
Where Robot Technology died?
When Robot Technology lived?
Where Robot Technology lived?
QASUM-TALP
What new applications of robot technology are in current use successfully?
What is new successfully applications of robot technology?

Table 4: Questions to be answered by the QA system.

5 Conclusions and Future Work

We have presented a query-driven multidocument summarizer. This system generates a set of new queries from the complex one given by the user. Then a Question Answering system is used to detect relevant information. After that the semantic representation of all relevant sentence are contrasted in order to avoid redundancy when selecting the summary content. A first approach was used to participate in the DUC 2005 contest. We also have presented a method to automatically evaluate different versions of our approach. This method has a good correlation with human Pyramid scores.

Talking about the proposed system, some directions for future work include the adequation of the PR module to face this kind of complex questions, as well as further experimentation in techniques for relevant information detection.

About our method to automatically detect SCUs, we think that it cannot only be useful as a way to evaluate automatic systems in a faster and less costly way than by manual evaluation, but that it can also be interesting as a tool for human annotators. When evaluating sets of summaries, it is difficult for humans to keep constant criteria along the set. On the contrary, our tool can give a homogeneous starting point that only needs to be corrected by the annotator.

However, it is clear that there is still work to be done on our tool. An interesting possibility is adding deeper linguistic information. By now, we only consider stems when matching SCUs and fragments of text. However, breaking down each sentence into a set of minimal semantic units, such as the Basic Elements proposed in [Hovy et al, 05], may improve the quality of the alignment and of the resulting pyramid annotations. We believe that exploration of methods in that direction can lead to the production of a better tool.

Acknowledgments

This work has been partially supported by the European Commission (CHIL, IST-2004-506909), the Spanish Research dept. (ALIADO, TIC2002-04447-C02) the Department of Universities, Research and Society of Information (DURSI) of the Catalan Government and the European Social Fund. Daniel Ferrés is supported by a UPC-Recerca grant from Universitat Politècnica de Catalunya (UPC). Our research group, TALP Research Center, is recognized as a Quality Research Group (2001 SGR 00254) by DURSI. We would like to thank Ani Nenkova and René Witte for useful discussion, as well as Hoa Trang Dang, Terry Copeck, and Stan Szpakowicz.

References

[Copeck and Szpakowicz, 05] T. Copeck and S. Szpakowicz. **Leveraging Pyramids**, *Notebook Papers of Document Understanding Conference (DUC 2005). HLT-NAACL 2005 Workshop, Vancouver, Canada*, 2005.

[Ferrés et al, 04] D. Ferrés, S. Kanaan, E. González, A. Ageno, H. Rodríguez, M. Surdeanu, and J. Turmo. **TALP-QA System at TREC 2004: Structural and Hierarchical Relaxation Over Semantic Constraints**, *Proceedings of the Text Retrieval Conference (TREC-2004)*, 2004.

[Ferrés et al, 05] D. Ferrés, S. Kanaan, D. Dominguez-Sal, E. González, A. Ageno, M. Fuentes, H. Rodríguez, M. Surdeanu, and J. Turmo. **TALP-UPC at TREC 2005: Experiments Using a Voting Scheme Among Three Hetereogeneous QA Systems**, *Proceedings of the Fourteenth Text Retrieval Conference (TREC-2005). Gaithersburg, Maryland, United States, 2005. To appear*, 2005.

[Fuentes and Rodríguez, 02] M. Fuentes, H. Rodríguez. **Using cohesive properties of text for automatic summarization**, *Proceedings JOTRI'02.* Valencia, Spain, 2002.

[Harnly et al, 05] A. Harnly, A. Nenkova, R. Passonneau, and O. Rambow **Automation of Summary Evaluation by the Pyramid Method**, *Proceedings RANLP'05* Borovets, Bulgary, 2005.

[Hovy et al, 05] E. Hovy, C.-Y. Lin, and L. Zhou **Evaluating DUC 2005 using Basic Elements**, *Notebook Papers of Document Understanding Conference (DUC 2005). HLT-NAACL 2005 Workshop, Vancouver, Canada*, 2005.

[Nenkova and Passonneau, 04] A. Nenkova and R. Passonneau. **Evaluating content selection in summarization: The pyramid method**, *Proceedings HLT/NAACL 2004*, 2004.