IIRG-UCD at DUC 2005

William Doran, Eamonn Newman, †Nicola Stokes, John Dunnion, Joe Carthy.

Intelligent Information Retrieval Group,
School of Computer Science and Informatics,
College of Mathematical and Physical Sciences,
UCD Dublin, Ireland

†NICTA Victoria Research Laboratory,
Department of Computer Science and Software Engineering,
University of Melbourne,
Victoria 3010,Australia

{William.Doran, Eamonn.Newman, John.Dunnion, Joe.Carthy}@ucd.ie,
nstokes@cs.mu.oz.au

Abstract

This paper describes a user-centric multi-document summarisation system. This system creates 250 word summaries of a collection of documents that satisfy a set of questions. This involves a linguistic approach to question reformulation and analysis. These questions are passed to a question answering system that uses word overlap, cosine similarity and grammatical similarity to extract answers. We also present the results of the official DUC evaluation of our system.

1 Introduction

The task of this year's Document Understanding Conference (DUC) is to provide a 250-word summary of a collection of related documents that answers a set of supplied questions to a specified level of granularity. There are fifty such collections of these documents called topics and each topic contains approximately twenty documents.

This task has evolved over the past number of years from a merging of several research areas and a changing information need of the user. In previous DUC's[3] we had single document summaries of varying lengths, then summaries of several similar documents and now we have summaries of several documents that answer specific and often related questions. The reason for this evolution is the constantly changing information need of the user, stemming from the explosion of information sources and diversity of topics available. It no longer suffices for a user to read a document to find the answer to their information need. We now want to be able to find specific details about our desired need taken from a wide range of documents. This year's task is a step in that direction.

Question Answering (QA) researchers have for years worked on trying to find important information in documents in the guise of answers to questions, while summarisation researchers have struggled for years to find/construct sentences that express the meaning of the document. Both have similar and related problems, that spawn the basis of this task; How do we find the answer to a particular question and how do we present it in a concise meaningful way within the context of the document?

The already difficult task of question answering is compounded in this year's task by having a set of questions that not only look for specific facts, but also facts related to the answers and even facts that relate to the answers of previous questions. These facts must then be presented in a coherent, structured and concise summary. This adds problems with anaphora resolution, inference and elaboration of facts, among others.

Our research has focussed on the breaking down of these sets of questions into independent questions that can then be used as input to a simple QA system. The output of this QA system is a ranked list of sentences that best match the questions posed. The processing of the questions is based on linguistic techniques that identify surface patterns and clues that can then be used to separated compound questions and reformulate them if necessary. The QA system uses a very simple matching strategy based on word overlap, Cosine Similarity[11] and Grammatical Relation (GR)[1] overlap. A ranked list of sentences is then returned based on the combination of the scores assigned to the sentence based on the three matching functions. The summary then comprises of a set of highly-ranked sentences for each question.

The next section of this paper provides more details on our system implementation, section

3 explains the evaluation of the summaries and presents the official evaluated results of our system. In section 4, we discuss our results and finally we make our conclusions and propose further work to be undertaken.

2 System Overview

In this section we present the approach we used to tackle the user focussed multi-document summarisation task in this years DUC. To fulfil this task one must provide a summary that answers a set of given questions to a required level of granularity. These questions are in a narrative form and are often inter-related and interdependent on each other. To address this problem we chose a modular approach that would break this problem into smaller manageable chunks. We firstly describe the pre-processing that was required, then we will discuss the procedure of breaking the narrative into manageable questions and finally describe the answer selection mechanism.

2.1 Pre-processing:

There are two main types of document involved in this years task; there are the SGML formatted news articles and a question file that includes the narrative-style questions for each topic. Each of these files goes through the pre-processing stage with the question file undergoing a further pre-processing stage laid out in the next section.

Several steps are required to convert the documents from their raw SGML format to the format to be used in our system. The first step is to remove the tags and meta-information and to convert the text into a single-sentence-per-line format. This standard format is necessary for the modular design of the system ensuring that all modules receive the documents in the same manner. We input the documents to a part of speech (POS) tagger which removes the SGML tags and gives part of speech information to the words in the document. Following this we use the POS information to place one sentence per line on the document.

The next step is to resolve any anaphoric references that occur through the document. Although anaphoric reference is in itself a difficult task, we felt that is was a useful addition to the system, given that one of the evaluation criteria is referential clarity. We resolve the references using the GUITAR[9] system, this is a probabilistic based system that reports 67% accuracy, and it performs reasonably well for this task. Using some additional scripting, we replace all referents with their antecedents.

Further to all this we carry out syntactic analysis of the sentences in the documents by using the RASP system [2].This system parses the sentence and give us a syntactic representation of the sentence which is useful in the reformulation of questions and in the selection of candidate answers. A full explanation of this process is given below.

2.2 Question Reformulation:

The narrative style of the questions in this task make it very difficult to apply normal question answering heuristics without some modification. We propose that breaking the narrative questions into separate questions would simplify the task and make it easier to deal with in a question answering context.

Each narrative contains several questions; some of these questions look for direct answers (e.g. who, where, when etc.) while others require more detail and inference (eg. Identify, explain etc.). There are also grammatical issues that arise with conjunctions, comparisons, clauses and elaborating sentences. Other issues that can arise are that of anaphora, and references in questions to the answers of previous questions. All these issues must be dealt with in order to successfully translate the narrative into a set of useful questions.

We use some simple metrics to classify the questions and then some simple grammatical rules to reformulate them if necessary. We classify the sentences into different categories so that we can maintain our modular architecture and use different strategies to answer different types of questions. The broad class of questions are broken down below with a brief explanation and example.

- "wh-"-These are the standard QA type questions of who, what, where etc.. E.g. "In what countries are MAGLEV rail systems being proposed?"
- Imperative- A question in the imperative that usually asks for particular details or specific information about something. E.g. "Explain the industrial espionage case involving VW and GM."
- Elaboration – These questions have a lead-in sentence or a subsequent elaborating sentence to give more information about the question. E.g. " Nobel prizes are awarded each year for achievements in the sciences (physics, chemistry, physiology and medicine) and economics. Who are the Nobel prize winners in the sciences and in economics and what are their prize-winning achievements?"

- Boolean- These questions look for a yes/no answer and can also ask to choose between several cases. E.g. “Are journalists specifically targeted?”
- Conjunctions- These questions contain conjunctions of items that need to be split up into separate questions. E.g. “Have diplomatic, economic, and military relations been restored?”
- Clausal- These questions contain several clauses that need to be refined into separated questions. E.g. “Where have poachers endangered wildlife, what wildlife has been endangered and what steps have been taken to prevent poaching?”
- Other- There are other types of question that can occur for example; including questions with prepositions and conjunction, and comparisons with previous answers. E.g. “What other factors affect the disputes?”,” What rules have been imposed regarding food labelling and by whom?”

Due to time constraints at the time of submission we had just one method for answering all types of question but we have investigate other techniques. We will briefly explain how we intend to implement these as well as giving an in-depth explanation of the method we used for the submission.

Firstly, we will describe the algorithm for splitting the sentences. The algorithm recurses several times to ensure that questions are split and reformulated correctly and also to ensure that any newly formed questions are also split and reformulated if required.

We begin with a queue of questions in single-line format with any anaphora resolved. The first step is to look for surface grammatical clues of conjunction of question clauses. These are usually two or more “wh-“ questions joined with a conjunction. We split the sentence as long as we have the following regular expression“,? and|or|but wh[o|ere|en]|how”. The next step is to look for questions that do not contain any of the common “wh” type question words. These types of sentences are generally either an imperative style question or an elaborating sentence. If the sentence begins with an imperative key word (explain, name, describe, identify, define etc.), then it is deemed to be an imperative question. If the sentence contains no such keywords at the start then it is deemed to an elaborating sentence. At present we haven’t linked these sentences to their elaborated question, but intend to use our previous work on lexical cohesion analysis[4] and anaphoric resolution to create a useful question from the elaborating sentence and question.

The next type of question to look for is the Boolean question. These questions usually have some form of the verb “to be” at the beginning of the question. E.g. is there, is, are, was etc.. The answer for these types of question is often a yes/no answer but there are also cases where you are asked to decide which case is true, e.g. is A, B or C true. We decided to reformulate these types of questions into statements. We did this by swapping the subject and the auxiliary of the verb. The premise behind this was to try to find grammatically similar sentences in the documents to the reformulated statements. We are currently working on a method of answering Boolean questions using textual entailment[5] to check the veracity of sentences in the text using the reformulated statement as a hypothesis, we feel that it will also be possible to use this to decide between several cases as above with A,B and C.

The next type of question are those that contain conjunctions. There are several different sub-class of these; the first being of the kind “Provide information on A, B and C”, this splits into three separate questions as the conjunction is used in an enumerative context. To separate these questions we generally look for a proposition followed by a conjunction of items. We then match the individual conjuncts to stem of the question to form separate questions. This stem usually occurs before the conjunction but it can also happen afterwards and this must be taken into account.

The second instance of this type of question occurs in the following form, ”Are the proposals for short_A or long_B haul_C”, where the conjunction joins two modifiers (A and B) of the head noun (C). In this case we must split the modifiers and rejoin them separately with the head noun to form two new questions. This case is greatly simplified when the modifiers are antonyms of each other. The final case we looked at was when we have a question followed by another short question, for example “where was Kennedy killed and by whom?”. In this case we need to split the sentence at the conjunction of clauses as before, but we also need to check that the resulting sentences are formed correctly. If the second sentence is too short (one or two words) E.g. “by whom”, then we create a new question by swapping the subject of the verb with the subject in the second. E.g. “by whom was Kennedy killed?”.

Following are some examples of the reformulation and breaking up of conjunctions of items.

Are the proposals for short or long haul? ?
The proposals are for long haul?
The proposals are for short haul?

Have diplomatic, economic, and military relations been restored? ?
Diplomatic relations have been restored?
Economic relations have been restored?
Military relations have been restored?

We continue to process the questions on the queue until there are no new questions formed. This then leaves us with a list of independent questions that can be used as input to a stand-alone question answering system. In the following section we will explain the method of how we extract candidate answers to the processed questions from the document collection.

2.3 Candidate Answer selection.

The selection of suitable candidate answers is crucial to the success of any question answering system. This difficulty of this stage is further compounded in this task by not only looking for direct answers to questions but also to other pieces of information that are related to the answer. It is for this reason that one strategy will not work for all types of question and thus we require several, often very different, methods for providing the sought answer in correct context.. In this section we will describe the strategy we implemented using Grammatical Relations and cosine similarity.

Above we explained how we spilt the narrative questions and classified the resulting new questions depending on the type of answer that is required. This classification is the basis for the expansion of our system using different modules to solve different classes of questions. For the submission we used a combination of modules that relied on simple word overlap statistics, grammatical similarity and cosine similarity.

The questions are pre-processed in the same manner as the documents in the collection. Thus we have a suitable representation of the questions and the sentences in the documents to make comparisons and find candidates that answer the questions. We perform a pair-wise comparison of all questions and document sentences. We compare them using three metrics; firstly, by measuring the amount of word overlap between the candidate sentence and the question. We perform stemming on both the sentence and question using the Porter algorithm[10]. We also use the widely used cosine similarity metric to measure similarity between the sentences. Finally, we use a similar method to the AnswerFinder system[6]. We use Grammatical Relation overlap, one of the metrics implemented by AnswerFinder. These relations were originally designed to compare the output of different types sentence parsers. These relations link works together in a more general way than in specific grammars and allow us to compare the syntactic and grammatical structure of sentences more easily.

These three similarity metrics each contributes a weighted score to the overall similarity of the sentence and question. These weights were hand-crafted based on empirical observation, ideally some regression technique could be used to automatically determine these weights. We then selected the three highest scoring sentences across the document collection for each question. This then formed the basis of our summary. In the next section we present and discuss our official results from DUC.

3 DUC evaluation

The evaluation procedure of this year's task comprehensively covered the intrinsic qualities required by a good summary i.e. linguistic quality and information content. Several tools, both automatic and semi-automatic were used to evaluate the information content and linguistic quality of the submitted summaries. The linguistic quality is determined by marks given out of five for various linguistic properties; 1) grammaticality, 2) non-redundancy, 3) referential clarity, 4) focus, 5) structure and coherence. The average scores for the linguistic quality questions are presented in table 1.

Peer	**1**	**2**	**3**	**4**	**5**
23	3.74	3.96	2.54	2.38	1.68

Quality Score (1 = very poor...5 = very good)

Table 1: Average Linguistic Quality Marks.

The information content is measured using several different methods. The first being responsiveness; this is a human-assigned ranking system that reflects the information content of a summary with the respect to the information need expressed in the topic. These scores must be scaled to allow for the differing number of human summaries. The score for our system is presented below in its raw and scaled forms in table 2. The "system only" score doesn't include scores for the number of human summaries, these are include for completeness in the column marked all.

Peer	**Raw**	**System only**	**All**
23	1.38	6.04	6.11

Table 2: Average Responsiveness Scores.

The second information content measuring metric is the fully automatic ROUGE evaluation system[5]. ROUGE scores systems based on the number of N-gram matches between the summary and several model summaries. The official ROUGE scores for this year are the ROUGE-2 metric (OR2), which is a measure of bi-gram overlap, and ROUGE-SU4 (ORSU4) which allows for matches that bridge a gap of four or less words. The official scores have been averaged to offset the bias given to topics that are evaluated with a greater number of model summaries. These scores are presented in table 3. We have also tabulated the raw averaged scores for the remaining rouge n-gram metrics, Rouge1 (rR1), Rouge2(rR2), RougeLCS, (rRLCS), RougeW1.2 (rRW) and finally RougeSU4 (rRSU4). These ROUGE results are laid out below in table 3.

OR2	ORSU4	rR1	rR2	rRLCS	rRW	rRSU4
0.026	0.056	0.179	0.026	0.416	0.12	0.16

Table 3: Raw Macro-averaged ROUGE

The final information content metric is that of the Pyramid Model Evaluation method[7]. This is the first time this method has been used on such a large scale. The Pyramid Model relies on two things; the first being the identification of SCU's (sub-sentence content units), and secondly the organisation of these SCU's into a weighted Pyramid structure. This Pyramid structure places higher importance on SCU's that occur in many model summaries than on SCU's that occur less frequently. Using this hierarchical technique we can compare a candidate summary to the Pyramid and highlight which SCU's occur in summary and the Pyramid. The score for the summary then depends on how many of its SCU's occur in the Pyramid, what level they occur at, the amount of repetition that occurs and the number of non-pyramid SCU's that occur in the summary. The identification of SCU's and building of Pyramids is done by hand. The annotation of summaries is also done by hand and the scoring is done automatically based on the annotations. The time constraints involved in the pyramid evaluation meant that only a subset of the topics were used in the evaluation. The official Pyramid scores for this are tabulated in table 4.

Peer 23	Pyramid Score	Modified Score	No. SCU/Summary
Average	0.115292	0.059588	2.653846
St. Dev	0.117111	0.071861	2.279339

Table 4: Official Pyramid Evaluation scores.

4 Discussion

Above we have presented the official results from our participation in this year's DUC. Overall some results are positive and some are disappointing. We performed well on the grammaticality and non-redundancy fields, this was slightly surprising as we had not implemented any of our previous multi-doc redundancy techniques. The grammaticality was no real surprise as we used an extractive sentence selection method. We felt the grammaticality may have suffered due to anaphora resolution but this doesn't seem to be the case. In light of this the referential clarity scores are good reflecting that anaphora resolution step is a worthwhile task.

The ROUGE results leave little to be interpreted. Across the spectrum of metrics, we didn't perform as well as we had expected. We only managed to gain on average 18% of the content of the model summaries (rR1), as a result of this the other metrics suffer. One probable reason for this was the sentence selection metric. We decided to pick only highly ranking sentences for each question posed. This works fine when a lot of questions are asked, or generated in our case, but when relatively few questions are present it reduces the number sentences added to the summary. In many cases this was below the threshold of 250 words thus reducing the chances of positive matching with the model summaries. We are currently investigating ways to pad out these summaries with other candidate sentences to try to boost the performance of our system.

The scores for the pyramid evaluation were not as good as we had hoped. Our system is a little behind the leading systems. Again we feel the brevity of the summaries is the primary reason behind this. On swift analysis of our results and performance, we failed to score at all on a number of topics. Looking at other systems scores for these topics, it seems that many, if not all systems struggled on these; posting lower scores, as well as a few joining us in failing to score at all. This would possibly indicate the difficulty of the topic and questions for these particular cases.

It will be interesting over the coming months to investigate the pro's and cons of each evaluation methodology, and to measure the level of correlation between them. It also presents an opportunity to analyse the summaries to identify reasons for poor/good performance.

5 Conclusion and Future work

Overall this was a very challenging task this year and called for a diverse and multi-faceted approach. The questions themselves were very complex and thus needed a complex system to perform well. We feel that this task will continue to challenge for many years to come.

The evaluation using the pyramid model was a very useful exercise and hopefully it will become easier to use and more beneficial once the community converges on a concrete set of regulations for its use.

Our results were disappointing but we feel we are on right track in trying to break the complex task into smaller manageable chunks. This is a very worthwhile task for us as it incorporates a lot of other work our group carries out. We intend to carry on with the question reformulation and will try to improve and develop upon what we have already achieved. We hope to incorporate simple discourse relations and try to relate the questions to each other and the documents in a more structured manner. We also plan to continue working on methods to find answers to the questions including textual entailment. As well as this we want to implement a plan-b option that will rely on previous work we have done in generic multi-doc summarisation. This will be used when the questions are very general and we hope it will boost our performance in subsequent years.

References:

[1] Ted Briscoe and John Carroll. 2000. Grammatical relation annotation. On-line document. http://www.cogs.susx.ac.uk/lab/nlp/carroll/grdescription/index.html.

[2] E. Briscoe and J. Carroll Robust accurate statistical annotation of general text. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, Las Palmas, Gran Canaria. 1499-1504. 2002

[3] Document Understanding Conference, http://www-nlpir.nist.gov/projects/duc/index.html

[4] William P. Doran, Nicola Stokes, John Dunnion, Joe Carthy. *Assessing the Impact of Lexical Chain Scoring Methods and Sentence Extraction Schemes on Summarization.* In the Proceedings of the 5th International conference on Intelligent Text Processing and Computational Linguistics CICLing-2004, 2004.

[5] Lin G-Y, Hovy E.. *Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics.* In Proceedings of HLT-NAACL-2003. 2003

[6] D. Mollá and M. Gardiner. AnswerFinder at TREC 2004 (2005)**.** *The Thirteenth Text REtrieval Conference (TREC 2004).2004*

[7] Ani Nenkova and Rebecca Passonneau, *Evaluating Content Selection in Summarization: the Pyramid Method.* In Procedings of Human Language Technology conference / North American chapter of the Association for Computational Linguistics. (NAACL-HLT),Boston, USA. 2004.

[8] Eamonn Newman, Nicola Stokes, Joe Carthy, John Dunnion. *UCD IIRG Approach to the Textual Entailment Challenge*. In the Proceedings of the PASCAL Recognising Textual Entailment Challenge, April 2005.

[9] Poesio, Massimo and Mijail A. Kabadjov. A General-Purpose, off-the-shelf Anaphora Resolution Module: Implementation and Preliminary Evaluation. In *Proc. of the 4th International Conference on Language Resources and Evaluation* (LREC), Lisbon, Portugal.2004

[10] Porter, M.F., "*An algorithm for suffix stripping*", Program; automated library and information systems, 14(3), 130-137, 1980.

[11] Salton, G., Singhal, A., Mitra, M., and Buckley, C. , Automatic text structuring and summarisation. *Information Processing and Management* 33(2):193–208. 1997