# Picking Phrases, Picking Sentences

Terry COPECK, Stan SZPAKOWICZ
School of Information Technology and Engineering
University of Ottawa
800 King Edward Avenue, P.O. Box 450 Stn. A
Ottawa, Ontario, Canada K1N 6N5
{terry, szpak}@site.uottawa.ca

## Abstract

Our summarizer system has been redesigned in response to previous experience from the conduct of research in this area and in order to allow us to carry out the tasks defined for DUC 2003 with some chance of good output. While major elements of the new design have not yet been fully implemented, the results produced by its current interim version on the three tasks for which we submitted summaries in 2003 were rated from slightly below average to fairly good.

## 1 Introduction

This is the third year in which the Document Understanding Conference has been held. From the perspective of participants, the focus of the conference has evolved over the 24 months since DUC 2001. Initially NIST set participants the two tasks[1] of producing a 100-word summary of a single document, and summaries of multiple documents in four sizes (NIST 2001). These tasks were reworked in 2002 to recognize the problem of variation in system sentence boundary recognition (NIST 2002). Adopting the typology recapitulated in Kan, Klavans and McKeown (2002), summaries of texts broken into sentences by a participant's system were called *abstracts*. For the second task, NIST provided decompositions of documents into 'standard' sentences in addition to plain text versions. Summaries composed of standard sentences were labeled *extracts*. While abstracts cannot be guaranteed to match across systems, extracts are commensurable.

In 2003 DUC turned its attention to a new issue. Insofar as only partial document decompositions were provided in 2003 for one task, the extraction/abstraction distinction was decided in favor of abstraction, and the conference began to consider the role of *bias* in summarization (NIST 2003). The two tasks of previous years were reworked to become four. The first and only single document task now asked for 10-word summaries—phrases, a sentence, a *headline*. The other three 2003 tasks each involved the use of one or more statements external to the document set to influence the construction of its multi-document summary. These were: a single *viewpoint* statement, a multi-faceted *topic* specification, and finally a *question* to answer.

This is a fundamental shift. It is not that four tasks involve more work—indeed, the two 2002 tasks required the submission of seven different summaries—but rather that in the first two competitions, a general-purpose summarizer could be applied to the tasks with only limited modification. This is no longer the case in 2003. Whether the sticking point is peculiarities in input or output, a summarization system aimed at general use can no longer easily be harnessed to address the DUC tasks. Satisfying the requirements of the 2003 tasks requires changes to the internals of system design.

The potential DUC participant is thus faced with a choice between not participating, or making fundamental changes to their software so it can carry out the specified tasks with some likelihood of adequate performance. The cost of doing so, however, is to develop a system that is tailored to one particular application. The fact that we are presenting this paper indicates which choice we made.

Repeat participants in DUC can probably identify an evolution in how they have grappled with the problems of summarization which parallels the evolution evident in the definition of the DUC tasks. In our case, this involved a shift away from the Machine Learning approach used in our system for the first two

[1] This account ignores the tasks called 'exploratory' in 2001 and 'pilot projects' in 2002.

conferences (Copeck, Japkowicz and Szpakowicz 2002; Copeck, Szpakowicz and Japkowicz 2002) in an attempt to determine which features in documents and factors in techniques had the greatest impact on the quality of the summaries produced. Our findings were taken into account in the design of a new system for DUC 2003, described in the following section.

Limited resources and the short, six-month period between the DUC workshop in July 2002 and the mid-February 2003 deadline to submit summaries for the current conference conspired to keep us from fully realizing our new design. In particular most of the development of its 'picking sentences' function that is described in the following section remains to be done.

## 2 System Design

We perform text summarization by *extracting* the most representative sentences from a *single* document, treating multiple documents as one by concatenating them in no specific order. A quick scan over the submissions to Tasks 2 and 3 posted on the DUC Web site suggests that most participants in DUC 2003—14 of 16 for Task 2 and 10 of 11 for Task 3—also compose summaries from whole or slightly edited sentences. Systems 15 and 19 are the exceptions. While this aspect of our system has remained unchanged in all versions, DUC 2003 required the development of somewhat different summarizers for meet the requirements of each of the three tasks we chose to compete in. Though each is unique, these three designs are however fairly simple modifications of a common system architecture that can be described as 'picking phrases, picking sentences' from a document reduced to *NLP-normal* form.

Let us elaborate on this quick description. 'NLP-normal' is our term for a document formatted one sentence per line, with a blank line between paragraphs. Its need arose from early work on columnar text which contained many spurious line endings. Less correction is necessary now (will all text one day be digital?), but the normalization module remains in our system because it serves to identify sentence boundaries.

### *2.1 General System Architecture*

The general architecture of our summarization system appears in Figure 1, with thin lines indicating the particular processing flows for the three DUC tasks. As the figure indicates, our software is composed of three modules.

The first module, normalization, was inherited from the previous version of the system. In addition to detecting sentence and paragraph boundaries,

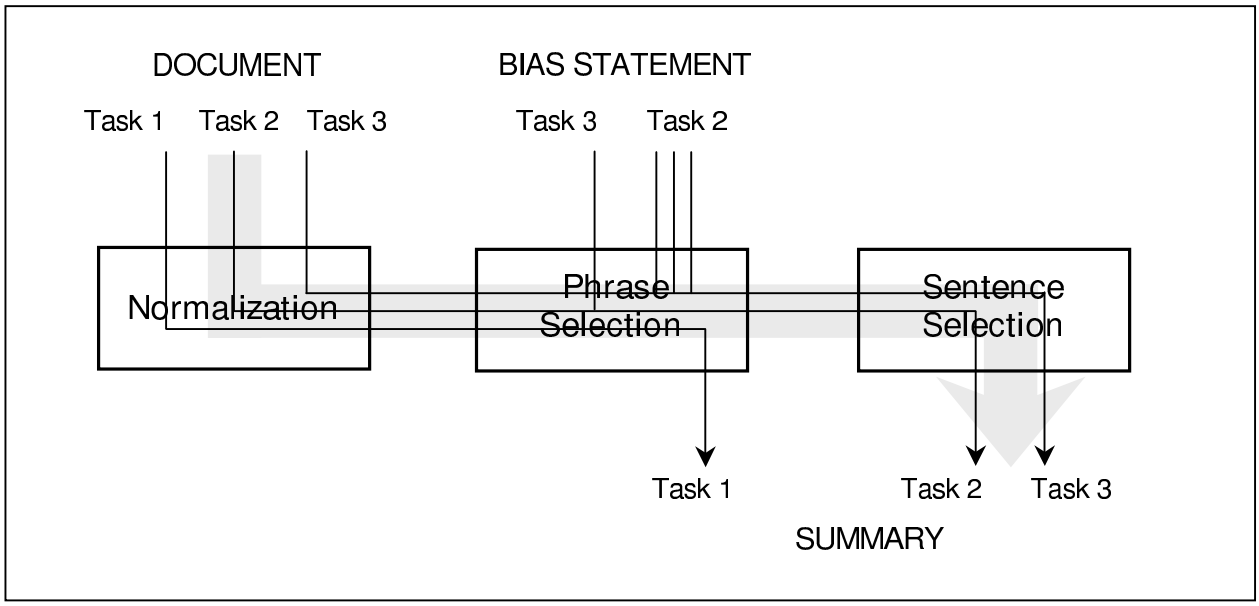


Figure 1. System and Task Architecture

| FEATURE | DESCRIPTION |
|---|---|
| chars | # of characters in the document |
| words | # of words in the document |
| sents | # of sentences in the document |
| paras | # of paragraphs in the document |
| kpiacnt | # of KPiA instances |
| conncnt | # of (Marcu) connectives |
| pncnt | # of PNs, acronyms |
| contcnt | # of content phrases |
| cphr, cphr2, cphr3, cphr4 | # of 1-, 2-, 3- and 3+ instance content phrases |
| cbig, cbig2, cbig3, cbig4 | # of 1-, 2-, 3- and 3+ instance content bigrams |
| abig, abig2, abig3, big4 | # of 1-, 2-, 3- and 3+ instance bigrams |

Table 1. Document features

normalization makes a number of passes over the document to remove tables, abstract, reference list, and page headers and footers (the latter in a fairly rigid way). What is left is called the document *body*. The module also tries to ensure that the document is encoded in 7-bit ASCII for Unix. Most of these operations are unnecessary for DUC documents but transparent to them.

Construction of a *data model* of the document is an operation separate from normalization but performed in conjunction with it. The model in question is a table containing for each sentence in the body, counts of the twenty lexical, syntactic and semantic features listed in Table 1. Two of these features require explanation. *KPiA* stands for 'Key Phrases in Abstract' and is an artefact of earlier work. *Marcu connectives* are items in the list of discourse cue phrases compiled by Daniel Marcu in his PhD dissertation (1997), slightly augmented.

Normalization is thus a clean-up procedure carried out prior to and in preparation for summarization. That actual operation is performed by the next two modules. The first extracts keyphrases from the document body, noting the sentences in which each occurs. The second picks from amongst these sentences ones to include in the summary, depending on which keyphrases are judged most pertinent. Identification of keyphrases can be accomplished in a number of ways of varying sophistication. While our current approach favors comprehensiveness, the tendency in the community is towards developing techniques of increasing sophistication, distinguishing not just phrases appearing literally in the text but the concepts and relations they mark (Hovy, Lin and Marcu 2003; Radev 2000).

Although cataloguing markers and rating them imposes an organization on the text, it does not produce a summary. To perform this final step a further process must determine which sentences best express the chosen markers, be they keyphrases in the syntax, or the constructs that these keyphrases mark. This is a difficult task because the selected sentences necessarily interplay with each other when read together in the summary, and interplay in the text with other sentences not selected according to the author's intention. Anaphoric reference is only the most obvious example of the latter. The particular approaches we took in these two modules will be discussed in connection with each DUC task in the following subsections.

## 2.2 *Task 1:* Headlines

The task here was to produce 10-word summaries of single documents. We submitted results in the easiest form to produce that was acceptable: a list of comma-delimited phrases. Our original intention was to approximate the character of headlines with clauses or clause-like constructions composed of noun phrases and a verb. To this end we integrated the LT-POS chunking parser from the University of Edinburgh into our system to provide access to all the phrasal chunks in a document.

At present these chunks are not used on the basis of their part of speech; initial trials showed that the verbs appearing in, at least, our test documents were too weak to invite direct use: *get*, *set*, *take* and the like. This does not mean that such verbs will not prove suitable when linked with concrete NPs, it simply means that we were not immediately encouraged to compose a ten-word summary in the way we had originally thought to. Our summaries were instead produced by linking chunked phrases together in a comma-delimited list. To construct this list, each member of the set of phrases is ranked according to the count of instances of its tokens in the text. The most highly-rated phrases are added to the summary according to the accumulation method presented in detail in the following paragraph.

WHAT: Chinese officials renew ban on formation of opposition parties in China

WHO: Li Peng, former Premier; Jiang Zemin, President; Wang Youcai, Qin Yongmin, Xu Wenli, opposition party leaders

WHERE: China

WHEN: November 23, 1998: Li Peng makes statement; December 20, 1998: Jiang Zemin makes statement

TOPIC EXPLICATION: In late November 1998, China's former Premier Li Peng renewed a pledge that China will not allow opposition parties. President Jiang Zemin also stated in December that China would never adopt a Western-style multi-party system. In the wake of Li Peng's announcement, opposition party organizers Xu Wenli and Qin Yongmin were arrested and sentenced, along with Wang Yongcai.

ON TOPIC: Stories covering Li Peng or Zemin's announcements concerning opposition parties, reaction within China and elsewhere, any direct consequences of the announcement (such as the arrest of opposition party leaders Xu, Qin and Wang).

Figure 2. Task 2 Topic Specification

Weighting pools inflected forms: *dog* and *dogs*, *receive* and *receiving*[2], when accumulating the instance count. An additional proviso is that no token in a phrase which is to be added to the summary can already appear there. This policy can be criticized but it does ensure that the summary has the broadest possible semantic span.

A close reading of the DUC task specifications led us to judge that a summary can approximate the target size (10-, 100- words) from either direction. That is, it can be larger or smaller. Our approach is to add units (phrases or sentences) until the next unit would bring the count of words to a sum which is farther from the target figure than the current total. When producing 100-word summaries, a 12-word sentence would thus not be added to a summary which already totaled 95 words.

[2] A weakness is that pooling occurs only if the root form of the word is present: 'dog' and 'dogs', 'dog' and 'dogging' will be pooled, but not 'dogs' and 'dogging'. We initially deemed it too computationally expensive to search for a common root for inflected forms, and then ran out of time.

### 2.3 *Task 2:* Topic *and Task 3:* Viewpoint

The objective of each of these tasks was to produce 100-word summaries of multiple documents according to a single viewpoint statement (Task 3) or as indicated by the specification of a topic (Task 2). The two tasks will be considered together because of their high degree of similarity. Both require that the selection of sentences (we continue to build large summaries by picking sentences from the text body) be guided by the content of a statement provided, which can be viewed as an approximation to a user query. Both were performed on a single text body constructed by simply concatenating the individual document bodies in the set to be summarized. Part-of-speech discrimination was not used in these tasks.

Topics and viewpoints differ as follows. A viewpoint is a single statement of fact or perspective, often a sentence, sometimes a list of phrases. Here are examples of viewpoints provided in DUC2003: 1) *Forty years after poor parenting was thought to be the cause of schizophrenia, researchers are working in many diverse areas to refine the causes and treatments of this disease and enable early diagnosis,* 2) *Growth and safety of internet auctions,* 3) *Hubble Space Telescope Service Missions* and 4) *Y2K dangers: threatened services.*

A topic, on the other hand, is communicated by a corpus specification borrowed from the Topic Detection and Tracking (TDT) project at NIST. A TDT specification is a structured record composed of eight top-level fields such as the title, an explication of the topic, the seed story etc. Of these, the Seminal Event and Topic Explication fields are most important. An example appears in Figure 2. A Seminal Event is composed of up to four statements specifying WHO, WHAT, WHERE and WHEN play a role in the issue discussed in articles that compose the topic cluster. We did not make use of the TDT Topic Explication.

Viewpoint summaries were produced by parsing each statement into its content phrases. The concatenated text body was then put into 'NLP-normal' format, and a data model was constructed of its contents (Copeck *et al* 2002a) Initially sentences in the body were ranked according to their exact matches of viewpoint phrases, but early testing showed that this

produced insufficient candidates for a summary — because they recapitulate a number of documents, many viewpoints contain words which generalize related concepts in a number of documents, with the generalized terms themselves appearing in none.

In response to this situation we created additional databases ranking sentences on other forms of matching. One database recorded phrases viewpoint matching substrings in text body sentences. A second recorded the reverse: sentence phrases which matched substrings of viewpoint phrases. Though the string sought was a sequence of one or more tokens, the matching operation did not respect token boundaries on the periphery of the target. As a result, unlike the case with exact matching, d*og* at the end of a phrase would now match *dogs* and *dogged*. A third 'short' database recorded the longest subsequence of tokens in a viewpoint phrase which are matched in one or more text body sentence. An example will clarify this. The viewpoint phrase *recognized opposition parties* would have its first token *recognized* removed and the substring *opposition parties* sought among text body sentences. If no match was found, *opposition* would be stripped off and the search performed again until success or until the viewpoint phrase was exhausted.

With these resources available the program began to construct the summary. It first handled two special cases. Any sentence containing two or more viewpoint phrases was immediately put in the summary. If space remained, any sentence which was the sole example of a viewpoint phrase not already represented was added to the summary.

Dealing with these two special cases never produced a hundred-word summary, and the program then moved on to add sentences exactly matching an unrepresented viewpoint phrase until the supply of candidates was exhausted. It next proceeded to add unrepresented matches from the 'short' database until none remained. After that it returned to the exact and short databases and added matches of already represented viewpoint phrases. If the summary was still of insufficient length, the summarizer turned to the poorer-quality substring databases, first adding unrepresented and then duplicate matches of viewpoint phrase substrings, and then repeating the process with text body substrings.

Examples may help illustrate this processing. Typical rather than ideal instances have been used. Figure 3 shows the summarization of document cluster D123H in terms of its viewpoint statement. The full statement and its phrasal decomposition appear at the top of the figure. The real world intrudes here: one of the words in the viewpoint statement is misspelled. *Sponsers* will never match any sentence in a document body.

The PHRASES table in the middle of the figure shows the accumulated result of phrase matching. *B-in-v* stands for 'body-[matched]-in-viewpoint', and *v-in-b* for 'viewpoint-in-body'. Neither of the two multiword viewpoint phrases, *flying schools* and the flawed *attends training offered*, is matched exactly or as a substring in the body sentences. The program has more

Viewpoint

| | |
|---|---|
| statement | Flying schools; who sponsers, who attends, training offered? |
| phrases | attends training offered, flying schools, sponsers |

Phrases

| TYPE | MATCH | SENTENCE NUMBERS |
|---|---|---|
| exact | | |
| v-in-b | | |
| b-in-v | attends | 94 |
| b-in-v | train | 25 45 281 |
| b-in-v | training | 93 193 198 250 |
| b-in-v | fly | 39 40 57 57 102 104 145 149 153 161 166 180 193 296 307 |
| b-in-v | school | 20 23 33 47 50 52 106 114 |
| b-in-v | flying | 79 98 |

Sentences

| TYPE | PHRASE | W | SENTENCES |
|---|---|---|---|
| 2, 2+ view phrases | | 0 | |
| sole exemplar | | 0 | |
| unique exact matches | | 0 | |
| unique shortened | | 0 | |
| duplicate exact | | 0 | |
| shortened exact | | 0 | |
| unique view in body | | 0 | |
| dup view in body | | 0 | |
| unique body in view | attends training offered | 10 | 93 |
| unique body in view | attends training offered, flying schools | 31 | 93 79 |
| dup body in view | " | 49 | 93 79 193 |
| dup body in view | " | 64 | 93 79 193 98 |
| dup body in view | " | 90 | 93 79 193 98 198 |

Figure 3. Viewpoint Processing for D123H

success with its *v-in-b* matching technique, which finds six phrases in sentences which match some part of a viewpoint phrase. Two of these are themselves substrings of other sentence phrases (fly→flying, train→training). Altogether 31 sentences were found from which to compose the summary.

Rows in the SENTENCES table at the bottom of Figure 3 show the results of one of the variety of passes over the databases of candidate sentences. Those following the discussion to this point will not be surprised to see none of the passes handling the two special cases, nor exact, shortened, or body-in-viewpoint matches provides any sentences for the summary (rows 1-8). The databases of these sorts of matches are all empty and only the viewpoint-in-body data offers hits. When the program finally arrives at this populated database, it first tries to construct a summary from sentences uniquely representing a viewpoint phrase (rows 9-10). The two sentences collected amount to only 31 words, so three additional passes must be made to bring the total count in words to 90 over five sentences (rows 11-13). From the perspective of our selection strategy, we have produced a summary, but using the weakest, most relaxed, strategies.

The actual construction of a summary from the sentences whose line numbers in the text body appear in the bag of matches proceeded in this way. Duplicate line numbers were removed from the bag to convert it to a set and, when more than one sentence remained, the first was used that offered less than the number of words remaining to the summary limit of 100 words. This is clearly an area where improvement can be made; the data model of each such sentence provides multiple criteria on which to base selection among candidates. The chosen sentences were then placed in text body order and output. Note that the sequencing this produces is entirely arbitrary, based as it is on the order in which individual texts in the document cluster were concatenated into one.

Processing for topic summaries was similar to that for viewpoints, with the complication that up to four statements or lists of phrases could be present for each 'WH-' *aspect* (our term) in a Seminal Event. The program also had to take into account that a phrase could appear under more than one aspect and also could be repeated in any aspect. A third special case was

Topic

| | |
|---|---|
| WHEN | September 23, 1998 |
| WHERE | Phillipines |
| WHAT | Phillipine Airlines (PAL) closes |
| WHO | Phillipine Airlines (PAL) |

Phrases

| TYPE | MATCH | SENTENCE NUMBERS |
|---|---|---|
| exact | September | 184 |
| exact | PAL | 42 44 48 71 73 78 80 82 92 109 119 135 149 150 152 155 165 167 171 173 179 181 184 186 186 187 193 195 195 209 213 221 229 243 249 279 |
| exact | PAL | 42 44 48 71 73 78 80 82 92 109 119 135 149 150 152 155 165 167 171 173 179 181 184 186 186 187 193 195 195 209 213 221 229 243 249 279 |

Sentences

| TYPE | PHRASE | W | SENTENCES | WHEN | WHAT | WHO |
|---|---|---|---|---|---|---|
| 2 or more | PAL, September | 24 | 184 | 1 | 1 | 1 |
| exact two | PAL, September | 24 | 184 | 1 | 1 | 1 |
| sole | PAL, September | 24 | 184 | 1 | 1 | 1 |
| least | PAL, September | 45 | 184 42 | 1 | 2 | 2 |
| least | PAL, September | 70 | 184 42 44 | 1 | 3 | 3 |
| least | PAL, September | 89 | 184 42 44 71 | 1 | 4 | 4 |

Figure 4. Topic Processing for D31041T

therefore added to the sequence of passes performed in sentence selection: a sentence should appear in the summary when a topic phrase represents two or more aspects. Because topics contain more phrases than viewpoints, it was possible to eliminate the more relaxed search strategies and limit selection to exact matches. In doing so, preference was given to sentences expressing whichever aspect(s) were least represented in the topic.

Figure 4 shows how this processing scheme was applied to produce a summary for document cluster D31041T. Its top table shows phrases extracted from the four aspects represented in the Seminal Event. The middle PHRASES table shows the exact matches found for two of the phrases; *PAL* is particularly well represented in the text body. The SENTENCES table shows that the first three passes (rows 1-3) made over the database of matches qualify #184 on three grounds: it contains two or more topic phrases, at least one of these phrases expresses two or more aspects, and one of them is the sole occurrence of the phase in the text body (*September*). However many times the program may add #184 in the summary, it still only amounts to one

sentence totaling 24 words. Consequently the program must duplicate one of the two topic phrases to accumulate a number of words approaching 100. It does this by adding sentences from the beginning of the list of hits for *PAL* (rows 4-5) according to the method presented in Section 2.2; that is, until the next sentence would increase the absolute difference between the current number of words and the target of 100. At that point (row 6), the program proceeds down the list of sentence matches until it finds a sentence, #71, whose addition will lower that absolute difference. Since no other sentence is added, we can be confident that none of the sentences remaining on *PAL*'s matches are less than 11 words (100 – 89). The rightmost three columns in SENTENCES show the number of times a phrase expressing an aspect is represented in the summary

## 3 Results

Limited resources prevent us from undertaking an analysis of the evaluation materials circulated by NIST prior to the workshop that would be more substantial than a survey of the basic statistics provided in the 'short results' table. Table 1 assembles this data for the 21 organizations who submitted summaries for the first three tasks in DUC 2003. This includes all participants; of these, six organizations other than ourselves submitted to all three tasks. We believe we are system #21, boldface in the table.

Assessing our results was easiest on Task 1, where the results on different measures tended to correlate consistently with each another. Taking length-adjusted coverage (n = 1 ⊃ mean = median) as the most significant measure, and with higher values indicating better performance, we came 8th out of 13 participants.

It is more difficult to judge results on Tasks 2 and 3 because the outcomes on different measures did not correlate as coherently. We therefore chose to make a more general judgment of results as 1) better than our own, 2) approximately on a par with ours, 3) worse than ours. Task 2 had 16 participants. Seven organizations appear to have performed distinctly better than us in terms of the SEE metrics, five did worse, and three were on a par (#10, #18, #26). This is approximately

| I | Related | # PUs | # M Pus | # U PUs | # Mus | Mean | Median | Mean L-A | Median |
|---|---|---|---|---|---|---|---|---|---|
| 6 | | | | | | | | | |
| 7 | 60 | 624 | 496 | 128 | 624 | 187 | 187 | 129 | 129 |
| 8 | 67 | 624 | 536 | 88 | 624 | 321 | 321 | 214 | 214 |
| 9 | 67 | 624 | 415 | 209 | 624 | 171 | 171 | 155 | 155 |
| 10 | 100 | 612 | 310 | 302 | 618 | 95 | 95 | 94 | 94 |
| 11 | | | | | | | | | |
| 12 | | | | | | | | | |
| 13 | 59 | 624 | 469 | 155 | 624 | 154 | 154 | 165 | 165 |
| 14 | | | | | | | | | |
| 15 | 145 | 624 | 306 | 318 | 624 | 104 | 104 | 101 | 101 |
| 16 | | | | | | | | | |
| 17 | 77 | 624 | 493 | 131 | 624 | 251 | 251 | 172 | 172 |
| 18 | 45 | 624 | 530 | 94 | 624 | 223 | 223 | 154 | 154 |
| 19 | | | | | | | | | |
| 20 | | | | | | | | | |
| 21 | 57 | 623 | 525 | 98 | 624 | 198 | 198 | 133 | 133 |
| 22 | 62 | 620 | 478 | 142 | 624 | 191 | 191 | 157 | 157 |
| 23 | | | | | | | | | |
| 24 | 54 | 564 | 353 | 211 | 564 | 133 | 133 | 131 | 131 |
| 25 | 61 | 624 | 518 | 106 | 624 | 182 | 182 | 121 | 121 |
| 26 | 23 | 624 | 571 | 53 | 624 | 236 | 236 | 161 | 161 |

| II | Related | # PUs | # M Pus | # U PUs | # Mus | Mean | Median | Sample | Mean L-A | Median |
|---|---|---|---|---|---|---|---|---|---|---|
| 6 | | 113 | 90 | 23 | 306 | 8 | 4 | 10 | 6 | 3 |
| 7 | | | | | | | | | | |
| 8 | | | | | | | | | | |
| 9 | | | | | | | | | | |
| 10 | 8 | 74 | 62 | 12 | 295 | 5 | 1 | 8 | 6 | 3 |
| 11 | 9 | 71 | 56 | 15 | 295 | 6 | 2 | 8 | 5 | 3 |
| 12 | 17 | 121 | 87 | 34 | 295 | 7 | 3 | 9 | 5 | 2 |
| 13 | 11 | 96 | 76 | 20 | 295 | 8 | 4 | 10 | 6 | 2 |
| 14 | 12 | 93 | 70 | 23 | 295 | 7 | 2 | 10 | 5 | 2 |
| 15 | 12 | 237 | 64 | 173 | 295 | 2 | 0 | 4 | 2 | 1 |
| 16 | 11 | 102 | 80 | 22 | 295 | 8 | 3 | 10 | 6 | 3 |
| 17 | 14 | 94 | 62 | 32 | 295 | 5 | 1 | 7 | 3 | 0 |
| 18 | 14 | 113 | 77 | 36 | 295 | 6 | 2 | 9 | 5 | 2 |
| 19 | 10 | 92 | 62 | 30 | 295 | 4 | 2 | 7 | 4 | 3 |
| 20 | 6 | 87 | 76 | 11 | 295 | 7 | 5 | 9 | 5 | 4 |
| 21 | 11 | 96 | 65 | 31 | 295 | 5 | 1 | 8 | 4 | 1 |
| 22 | 8 | 59 | 49 | 10 | 295 | 6 | 2 | 9 | 7 | 5 |
| 23 | 8 | 86 | 68 | 18 | 295 | 7 | 2 | 10 | 6 | 2 |
| 24 | | | | | | | | | | |
| 25 | | | | | | | | | | |
| 26 | 21 | 148 | 96 | 52 | 295 | 6 | 2 | 8 | 4 | 1 |

| III | Related | # PUs | # M Pus | # U PUs | # Mus | Mean | Median | Sample | Mean L-A | Median |
|---|---|---|---|---|---|---|---|---|---|---|
| 6 | | | | | | | | | | |
| 7 | | | | | | | | | | |
| 8 | | | | | | | | | | |
| 9 | | | | | | | | | | |
| 10 | 13 | 74 | 54 | 20 | 298 | 5 | 1 | 8 | 5 | 3 |
| 11 | 12 | 64 | 46 | 18 | 298 | 4 | 1 | 8 | 5 | 2 |
| 12 | | | | | | | | | | |
| 13 | 17 | 104 | 68 | 36 | 298 | 4 | 1 | 7 | 3 | 1 |
| 14 | | | | | | | | | | |
| 15 | 14 | 235 | 63 | 172 | 298 | 3 | 1 | 6 | 3 | 1 |
| 16 | 13 | 106 | 76 | 30 | 298 | 6 | 1 | 9 | 4 | 1 |
| 17 | 16 | 91 | 57 | 34 | 298 | 6 | 2 | 8 | 4 | 1 |
| 18 | 20 | 107 | 59 | 48 | 298 | 5 | 1 | 8 | 5 | 2 |
| 19 | | | | | | | | | | |
| 20 | 15 | 93 | 66 | 27 | 298 | 6 | 3 | 8 | 4 | 2 |
| 21 | 16 | 107 | 62 | 45 | 298 | 4 | 2 | 7 | 3 | 2 |
| 22 | 14 | 79 | 60 | 19 | 298 | 5 | 1 | 8 | 5 | 2 |
| 23 | 11 | 92 | 61 | 31 | 298 | 5 | 1 | 8 | 4 | 1 |
| 24 | | | | | | | | | | |
| 25 | | | | | | | | | | |
| 26 | | | | | | | | | | |

Table 1. Short Results for Tasks 1 to 3

equivalent to our slightly worse-than-average performance on the first task. Eleven participants submitted summaries for Task 3. Three (#13, #16, #20) outperformed us. No other organization seems truly to be on a par, while seven did worse. Although the range of results on Task 3 was not great, the results there seem to be our best performance in DUC 2003. Because the revised design of our current system has yet to be fully implemented we draw no conclusions from these outcomes.

## 4 Conclusions and Future Work

Our first future task must be to improve the grounds on which sentences are selected for a summary. 'First come, with a nod to best packing' can be nothing more than an interim strategy. A resource that will assist us in making a more informed choice is the document data model. Its record of sentence feature counts will provide us with likely raw material for experimentation to improve sentence selection. In order to perform such experiments, we will need to conduct a similar analysis of the manual abstracts provided for each DUC task. If these human-written abstracts are taken as 'gold standard', there is the basis for an application of ML akin to that of Turney (2000).

The first step in moving to an understanding of document content deeper than surface syntax would be to recognize the identity of synonyms. While WordNet is the obvious candidate to use for this purpose, locally-developed software (Jarmasz and Szpakowicz 2001) provides access to a machine-readable version of the 1987 edition of Roget's thesaurus, the other main English asset of this kind. Use of the two resources together may help disambiguate polysemous words.

## Acknowledgements

This work has been supported by the Natural Sciences and Engineering Research Council of Canada.

## References

Copeck, T., N. Japkowicz and S. Szpakowicz. 2002. Text Summarization as Controlled Search. Proc 15th Conf of the CSCSI (AI 2002) (LNAI 2338), Calgary, 268-280.

Copeck, T., S. Szpakowicz and N. Japkowicz. 2002. Learning How Best to Summarize. Workshop on Text Summarization, Proc 3rd Document Understanding Conference (DUC 2002). Philadelphia, July, 44-47, 47a-d.

Finch, S. and A. Mikheev. 1997. A Workbench for Finding Structure in Texts. Proc. 5th Applied Natural Language Processing (ANLP), Washington D.C., April, 372-379.

Hovy, E., C.-Y. Lin and D. Marcu. 2003. SUMMARIST: Automatic Text Summarization. www.isi.edu/natural-language/projects/ SUMMARIST.html

Jarmasz, M. and S. Szpakowicz. 2001. The Design and Implementation of an Electronic Lexical Knowledge Base. Proc 14th Conf of the CSCSI (AI 2001). Ottawa, June, 325 - 334.

Kan, M.-Y., J. Klavans and K. McKeown. 2002. Using the Annotated Bibliography as a Resource for Indicative Summarization. Proc LREC 2002, Las Palmas, Spain, 1746-1752.

Marcu, D. 1997. The Rhetorical Parsing, Summarization, and Generation of Natural Language Texts. PhD Thesis, Department of Computer Science, University of Toronto, December 1997.

NIST. 2001. DUC 2001 Guidelines. www.nlpir.nist.gov/projects/duc/guidelines/2001.html.

NIST. 2002. DUC 2002 Guidelines. www.nlpir.nist.gov/projects/duc/guidelines/2002.html.

NIST. 2003. DUC 2003 Guidelines. duc.nist.gov/duc2003/tasks.html.

Radev, D. 2000. A Common Theory Of Information Fusion From Multiple Text Sources, Step One: Cross-Document Structure. Proc 1st ACL SIGDIAL Workshop on Discourse and Dialogue. Hong Kong, October.

Turney, P. 2000. Learning algorithms for keyphrase extraction. Information Retrieval, 2 (4), 303-336.