Selecting Sentences for Multidocument Summaries using Randomized Local Search

Michael White
CoGenTex, Inc.
840 Hanshaw Road
Ithaca, NY 14850, USA
mike@cogentex.com

Claire Cardie
Dept. of Computer Science
Cornell University
Ithaca, NY 14850, USA
cardie@cs.cornell.edu

Paper ID: Pxxxx

Keywords: multidocument summarization, evaluation, sentence extraction, intelligibility

Contact Author: Michael White

Under consideration for other conferences (specify)? no

Abstract

We present and evaluate a randomized local search procedure for selecting sentences to include in a multidocument summary. The search favors the inclusion of adjacent sentences while penalizing the selection of repetitive material, in order to improve intelligibility without unduly affecting informativeness. Sentence similarity is determined using both surface-oriented measures and semantic groups obtained from merging the output templates of an information extraction subsystem. In a comparative evaluation against two DUC-like baselines and three simpler versions of our system, we found that our randomized local search method provided substantial improvements in both content and intelligibility, while the use of the IE groups also appeared to contribute a small further improvement in content.

Selecting Sentences for Multidocument Summaries using Randomized Local Search

Paper ID: Pxxxx

Abstract

We present and evaluate a randomized local search procedure for selecting sentences to include in a multidocument summary. The search favors the inclusion of adjacent sentences while penalizing the selection of repetitive material, in order to improve intelligibility without unduly affecting informativeness. Sentence similarity is determined using both surface-oriented measures and semantic groups obtained from merging the output templates of an information extraction subsystem. In a comparative evaluation against two DUC-like baselines and three simpler versions of our system, we found that our randomized local search method provided substantial improvements in both content and intelligibility, while the use of the IE groups also appeared to contribute a small further improvement in content.

1 Introduction

Improving the intellibility of multidocument summaries remains a significant challenge. While most previous approaches to multidocument summarization have addressed the problem of reducing repetition, less attention has been paid to problems of coherence and cohesion. In a typical extractive system (e.g. (Goldstein et al., 2000)), sentences are selected for inclusion in the summary one at a time, with later choices sensitive to their similarity to earlier ones; the selected sentences are then ordered either chronologically or by relevance. The resulting summaries often jump incoherently from topic to topic, and contain broken cohesive links, such as dangling anaphors or unmet presuppositions.

Barzilay et al. (2001) present an improved method of ordering sentences in the context of MultiGen, a multidocument summarizer that identifies sets of similar sentences, termed *themes*, and reformulates their common phrases as new text. In their approach, topically related themes are identified and kept together in the resulting summary, in order to help improve cohesion and reduce topic switching.

In this paper, we pursue a related but simpler idea in an extractive context, namely to favor the selection of blocks of adjacent sentences in constructing a multidocument summary. Here, the challenge is to improve intelligibility without unduly sacrificing informativeness; for example, selecting the beginning of the last article in a document set will usually produce a highly intelligible text, but one that is not very representative of the document set as a whole.

To manage this tradeoff, we have developed a randomized local search procedure (cf. (Selman and Kautz, 1994)) to select the highest ranking set of sentences for the summary, where the inclusion of adjacent sentences is favored and the selection of repetitive material is penalized. The method involves greedily searching for the best combination of sentences to swap in and out of the current summary until no more improvements are possible; noise strategies include occasionally adding a sentence to the current summary, regardless of its score, and restarting the local search from random starting points for a fixed number of iterations. In determining sentence similarity, we have used surface-oriented similarity measures obtained from Columbia's SimFinder tool (Hatzivassiloglou et al., 2001), as well as semantic groups obtained from merging the output templates of an information ex-

traction (IE) subsystem.

In related work, Marcu (2001) describes an approach to balancing informativeness and intelligibility that also involves searching through sets of sentences to select. In contrast to our approach, Marcu employs a beam search through possible summaries of progressively greater length, which seems less amenable to an anytime formulation; this may be an important practical consideration, since Marcu reports search times in hours, whereas we have found that less than a minute of searching is usually effective.

In order to evaluate our approach, we compared 200-word summaries generated by our system against those of two baselines, similar to those used in DUC 2001 (Harman, 2001), and three simpler versions of the system, where a simple marginal relevance selection procedure was used instead of the selection search, and/or the IE groups were ignored. In general, we found that our randomized local search method provided substantial improvements in both content and intelligibility over the DUC-like baselines and the simplest variant of our system, the one using marginal relevance selection and no IE groups (with the exception that the last article baseline was always ranked first in intelligibility). The use of the IE groups also appeared to contribute a small further improvement in content when used with our selection search.

2 System Description

We have implemented our randomized local search method for sentence selection as part of the RIPTIDES system. RIPTIDES combines information extraction (IE) in the domain of natural disasters and multidocument summarization to produce hypertext summaries. The hypertext summaries include a high-level textual overview; tables of all *comparable* numeric estimates, organized to highlight discrepancies; and targeted access to supporting information from the original articles. In (White et al., 2002), we showed that the hypertext summaries can help to identify disrepancies in numeric estimates, and provide a significantly more complete picture of the available information than the latest article. The next subsection walks through a sample hypertext summary; it is followed by descriptions of the IE and Summarizer system components.

2.1 Example

Figure 1 shows a textual overview of the first dozen or so articles in a corpus of news articles gathered from the web during the first week after the January 2001 earthquake in Central America. Clicking on the magnifying glass icon brings up the original article in the right frame, with the extracted sentences highlighted.

The index to the hypertext summary appears in the left frame of figure 1. Links to the overview and lead sentence of each article are followed by links to summary information organized according to the base level extraction slots for the main event (here, an earthquake) including its description, date, location, epicenter and magnitude. Access to overall damage estimates appear next, with separate tables for types of human effects (e.g. dead, missing) and for object types (e.g. villages, bridges, houses) with physical effects.

Figure 2 shows the extracted estimates of the overall death toll. In order to help identify discrepancies, the high and low current estimates are shown at the top, followed by other current estimates and then all extracted estimates. Heuristics are used to determine which estimates to consider current, taking into account the source (either news source or attributed source), specificity (e.g. *hundreds* vs. *at least 200*) and confidence level, as indicated by the presence of hedge words such as *perhaps* or *assumed*. The tables also provide links to the original articles, allowing the user to quickly and directly determine the accuracy of any estimate in the table.

2.2 IE System

The IE system combines existing language technology components (Bikel et al., 1997; Charniak, 1999; Day et al., 1997; Fellbaum, 1998) in a traditional system architecture (Cardie, 1997; Grishman, 1996). Unique features of the system

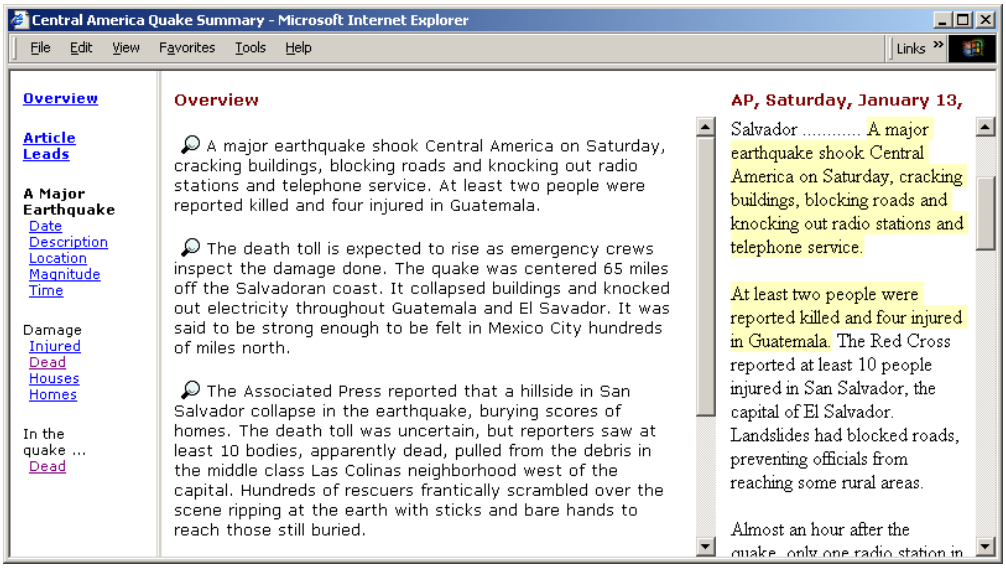

Figure 1: Example Multidocument Hypertext Summary Overview

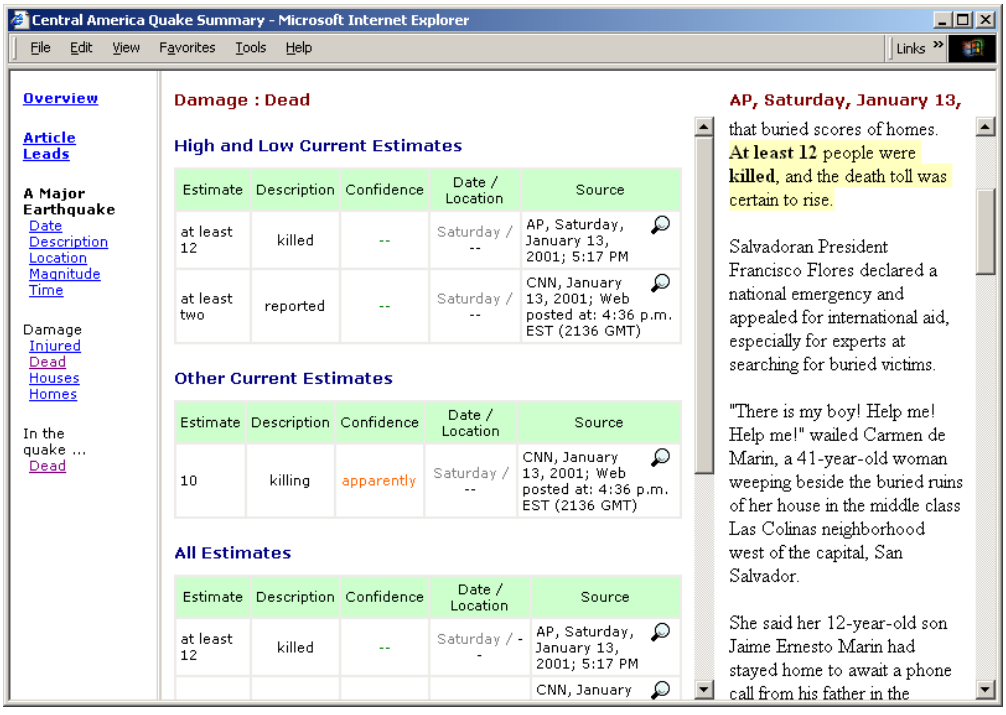

Figure 2: Example Tables of Death Toll Estimates

include a weakly supervised extraction pattern-learning component, Autoslog-XML, which is based on Autoslog-TS (Riloff, 1996), but operates in an XML framework and acquires patterns for extracting text elements beyond noun phrases, e.g. verb groups, adjectives, adverbs, and single-noun modifiers. In addition, a heuristic-based clustering algorithm organizes the extracted concepts into output templates specifically designed to support multi-document summarization (White et al., 2001): the IE system, for example, distinguishes different reports or views of the same event from multiple sources.

Output templates from the IE system for each text to be covered in the multi-document summary are provided as input to the summarization component along with all linguistic annotations accrued in the IE phase.

2.3 Summarizer

The Summarizer operates in three main stages. In the first stage, the IE output templates are merged into an event-oriented structure where comparable facts are semantically grouped. Towards the same objective, surface-oriented clustering is used to group sentences from different documents into clusters that are likely to report similar content. In the second stage, importance scores are assigned to the sentences based on the following indicators: position in document, document recency, presence of quotes, average sentence overlap, headline overlap, size of cluster (if any), size of semantic groups (if any), specificity of numeric estimates, and whether these estimates are deemed current. In the third and final stage, the hypertext summary is generated from the resulting content pool. Further details on each stage follow in the paragraphs below; see (White et al., 2002) for a more complete description.

In the analysis stage, we use Columbia's SimFinder tool (Hatzivassiloglou et al., 2001) to obtain surface-oriented similarity measures and clusters for the sentences in the input articles. To obtain potentially more accurate partitions of the IE output, we semantically merge the extracted slots into *comparable* groups, i.e. ones whose members can be examined for discrepancies. This requires distinguishing (i) different types of damage; (ii) overall damage estimates vs. those that pertain to a specific locale; and (iii) damage due to related events, such as previous quakes in the same area. During this stage, we also analyze the numeric estimates for specificity and confidence level, and determine which estimates to consider current.

In the scoring stage, SimFinder's similarity measures and clusters are combined with the semantic groupings obtained from merging the IE templates in order to score the input sentences. The scoring of the clusters and semantic groups is based on their size, and the scores are combined at the sentence level by including the score of all semantic groups that contain a phrase extracted from a given sentence. More precisely, the scores are assigned in two phases, according to a set of hand-tuned parameter weights. First, a base score is assigned to each sentence according to a weighted sum of the position in document, document recency, presence of quotes, average sentence overlap, and headline overlap. The average sentence overlap is the average of all pairwise sentence similarity measures; we have found this measure to be a useful counterpart to sentence position in reliably identifying salient sentences, with the other factors playing a lesser role. In the second scoring phase, the clusters and semantic groups are assigned a score according to the sum of the base sentence scores. After normalization, the weighted cluster and group scores are used to boost the base scores, thereby favoring sentences from the more important clusters and semantic groups. Finally, a small boost is applied for current and more specific numeric estimates.

In the generation stage, the overview is constructed by selecting a set of sentences in a context-sensitive fashion, then ordering the blocks of adjacent sentences according to their importance scores. The scoring model begins with the sum of the scores for the candidate sentences, which is then adjusted to penalize the inclusion of multiple sentences from the same cluster or semantic group, or sentences whose similarity measure is above a certain threshold,

and to favor the inclusion of adjacent sentences from the same article, in order to boost intelligibility. A larger bonus is applied when including a sentence that begins with an initial pronoun as well as the previous one, and an even bigger bonus is added when including a sentence that begins with a strong rhetorical marker (e.g. *however*) as well as its predecessor; corresponding penalties are also used when the preceding sentence is missing, or when a short sentence appears without an adjacent one.

To select the sentences for the overview according to this scoring model, we use an iterative randomized local search procedure inspired by (Selman and Kautz, 1994). For the first iteration, we begin with the highest scoring sentences up to the word limit. For subsequent iterations, we begin with randomly selected sentences, weighted according to their scores, up to the word limit. During each iteration, a random step or a greedy step is repeatedly performed until a greedy step fails to improve upon the current set of sentences. In each random step, a randomly selected sentence is added to collection. In each greedy step, one sentence is chosen to add to the summary, and zero or more (typically one) sentences are chosen to remove from the summary, such that the word limit is still met, and this combination of sentences represents the best swap available according to the scoring model. The search continues for a predetermined number of iterations, keeping track of the best combination of sentences found so far; it could easily be formulated in an anytime fashion as well. From a practical perspective, we have found that 10 iterations often suffices to find a reasonable collection of sentences, taking well under a minute on a desktop PC.

Once the overview sentences have been selected, the hypertext summary is generated as a collection of HTML files, using a series of XSLT transformations.

2.4 Training and Tuning

For the evaluation below, the IE system was trained on 12 of 25 texts from topic 89 of the TDT2 corpus, a set of newswires that describe the May 1998 earthquake in Afganistan. It achieves 42% recall and 61% precision when evaluated on the remaining 13 topic 89 texts. The parameters of the Summarizer were chosen by hand using the complete TDT2 topic 89 document set as input.

3 Evaluation Method and Results

To select the inputs for the evaluation, we took subsets of the articles from TDT2 topic 89, where the subsets consisted of all the articles up to the end of days 1 through 5 after the quake. We chose to use TDT2 topic 89 so that we could assess the impact of the IE quality on the results, given that we had previously created manual IE annotations for these articles (White et al., 2001).[1]

For each input document set, we ran the RIPTIDES system to produce overview summaries of 200 words or less. For comparison purposes, we also ran two baselines, similar to those used in DUC 2001 (Harman, 2001), and three simpler versions of the system, for a total of six summary types:

Last The first N sentences of the latest article in the document set, up to the word limit.

Leads The lead sentences from the latest articles in the document set, up to the word limit, listed in chronological order.

MR The top ranking sentences selected according to their thresholded marginal relevance, up to the word limit, listed in chronological order, using RIPTIDES to score the sentences, except with the IE groups zeroed out.

MR+IE The top ranking sentences selected according to their thresholded marginal relevance, up to the word limit, listed in chronological order, using RIPTIDES

[1] Although our decision to use subsets of TDT2 topic 89 as inputs meant that our training/tuning and test data overlapped, we do not believe that this choice overly compromises our results, since — as will be discussed in this section and the next — the impact of the IE groups turned out to be small, while with our selection search, we ran into a couple of problems on the test data that did not show up in tuning the parameter settings.

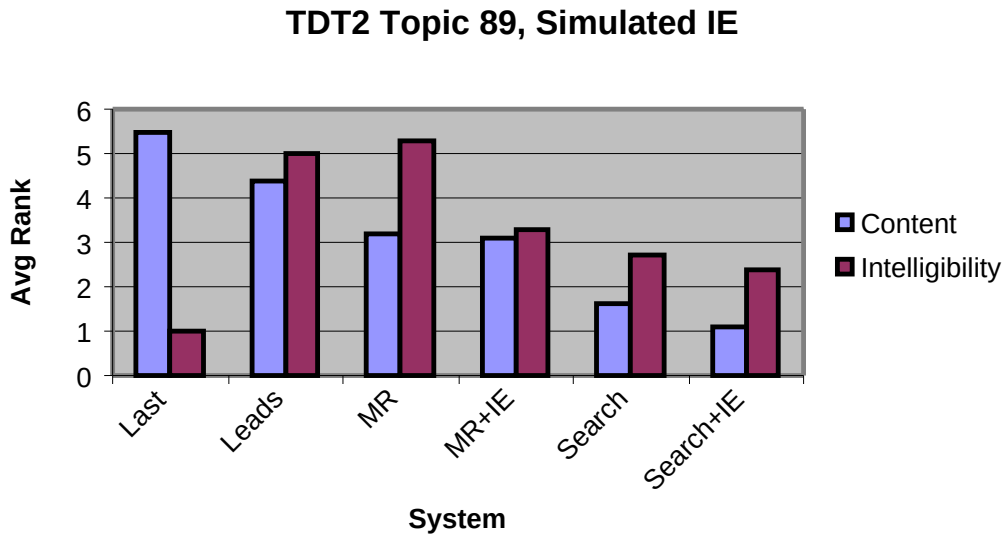

Figure 3: Average System Rank for Content and Intelligibility on TDT2 Topic 89, using Simulated IE. The ranks are averaged across two judges and five time points; manual IE annotations were used with the MR+IE and Search+IE systems.

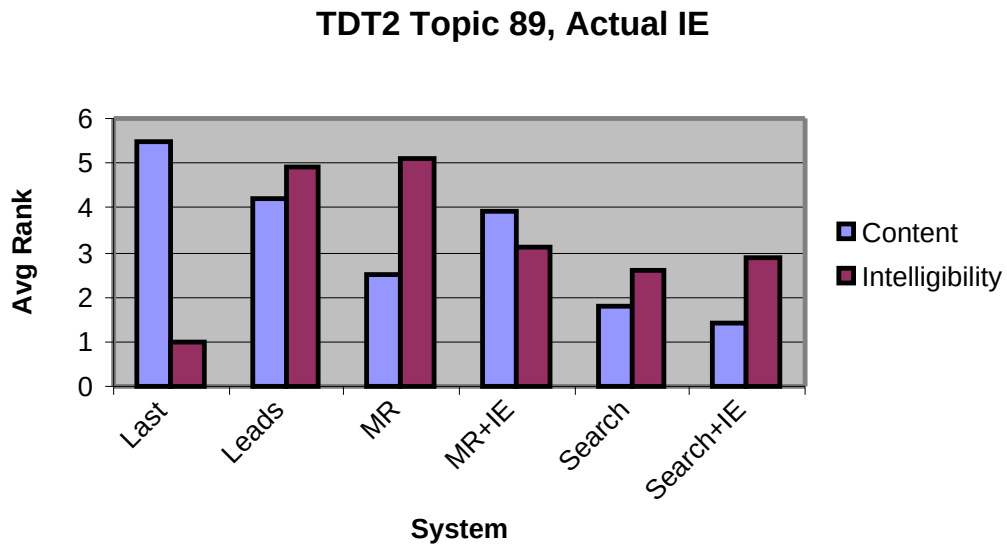

Figure 4: Average System Rank for Content and Intelligibility on TDT2 Topic 89, using Actual IE. The ranks are averaged across two judges and five time points; actual IE annotations were used with the MR+IE and Search+IE systems, making all systems fully automatic.

Content Rank, Simulated IE						
	Last	Leads	MR	MR+IE	Search	Search+IE
Day 1	5, 6	5, 5	4, 4	3, 3	1, 1	1, 1
Day 2	5, 4	6, 6	1, 4	4, 3	2, 1	1, 2
Day 3	6, 6	3, 3	3, 3	5, 5	1, 1	1, 1
Day 4	6, 5	3, 6	3, 4	3, 2	2, 2	1, 1
Day 5	6, 6	2, 5	2, 4	2, 2	2, 2	1, 1
Average	5.5 ±0.7	4.4 ±1.5	3.2 ±1.0	3.1 ±1.2	1.6 ±0.7	1.1 ±0.3

Table 1: Content Rankings on TDT2 Topic 89, using Simulated IE. The scores for the two judges at each time point are separated by commas.

Intelligibility Rank, Simulated IE						
	Last	Leads	MR	MR+IE	Search	Search+IE
Day 1	1, 1	5, 2	6, 6	3, 2	2, 4	3, 5
Day 2	1, 1	5, 5	6, 5	4, 4	2, 2	2, 3
Day 3	1, 1	6, 5	5, 6	3, 4	3, 1	2, 1
Day 4	1, 1	6, 6	5, 5	3, 3	3, 4	2, 2
Day 5	1, 1	4, 6	4, 5	4, 3	2, 4	2, 2
Average	1 ±0	5 ±1.2	5.3 ±0.7	3.3 ±0.7	2.7 ±1.1	2.4 ±1.1

Table 2: Intelligibility Rankings on TDT2 Topic 89, using Simulated IE.

Content Rank, Actual IE						
	Last	Leads	MR	MR+IE	Search	Search+IE
Day 1	5, 6	5, 5	4, 4	3, 3	2, 1	1, 1
Day 2	5, 4	6, 6	1, 4	3, 3	3, 2	1, 1
Day 3	6, 6	1, 2	1, 2	5, 5	3, 1	3, 2
Day 4	6, 5	4, 6	1, 2	5, 4	1, 1	1, 2
Day 5	6, 6	2, 5	4, 2	4, 4	2, 2	1, 1
Average	5.5 ±0.7	4.2 ±1.9	2.5 ±1.4	3.9 ±0.9	1.8 ±0.8	1.4 ±0.7

Table 3: Content Rankings on TDT2 Topic 89, using Actual IE.

Intelligibility Rank, Actual IE						
	Last	Leads	MR	MR+IE	Search	Search+IE
Day 1	1, 1	5, 2	6, 6	3, 2	2, 4	3, 5
Day 2	1, 1	5, 5	6, 5	4, 4	2, 2	2, 2
Day 3	1, 1	6, 4	5, 6	2, 3	2, 2	2, 4
Day 4	1, 1	6, 6	4, 4	4, 3	3, 2	2, 4
Day 5	1, 1	4, 6	4, 5	4, 2	3, 4	2, 3
Average	1 ±0	4.9 ±1.3	5.1 ±0.9	3.1 ±0.9	2.6 ±0.8	2.9 ±1.1

Table 4: Intelligibility Rankings on TDT2 Topic 89, using Actual IE.

P-Values for Content, Simulated IE						
	Last	Leads	MR	MR+IE	Search	Search+IE
Last	-	0.0570	**0.0001**	**0.0001**	**0.0001**	**0.0001**
Leads		-	0.0542	0.0473	**0.0001**	**0.0001**
MR			-	0.8438	**0.0009**	**0.0001**
MR+IE				-	0.0040	**0.0004**
Search					-	0.0607

Table 5: P-Values from Pairwise t-Tests for Difference Between Mean Content Rank on TDT2 Topic 89, using Simulated IE. Significant differences are shown in bold; differences below 0.0001 are rounded up.

P-Values for Intelligibility, Simulated IE						
	Last	Leads	MR	MR+IE	Search	Search+IE
Last	-	**0.0001**	**0.0001**	**0.0001**	**0.0007**	**0.0026**
Leads		-	0.5145	**0.0020**	**0.0003**	**0.0001**
MR			-	**0.0001**	**0.0001**	**0.0001**
MR+IE				-	0.1513	0.0403
Search					-	0.5375

Table 6: P-Values from Pairwise t-Tests for Difference Between Mean Intelligibility Rank on TDT2 Topic 89, using Simulated IE.

P-Values for Content, Actual IE						
	Last	Leads	MR	MR+IE	Search	Search+IE
Last	-	0.0636	**0.0001**	**0.0003**	**0.0001**	**0.0001**
Leads		-	0.0332	0.6542	**0.0028**	**0.0009**
MR			-	0.0147	0.1789	0.0393
MR+IE				-	**0.0001**	**0.0001**
Search					-	0.2459

Table 7: P-Values from Pairwise t-Tests for Difference Between Mean Content Rank on TDT2 Topic 89, using Actual IE.

P-Values for Intelligibility, Actual IE						
	Last	Leads	MR	MR+IE	Search	Search+IE
Last	-	**0.0001**	**0.0001**	**0.0001**	**0.0002**	**0.0004**
Leads		-	0.6899	**0.0022**	**0.0002**	**0.0016**
MR			-	**0.0001**	**0.0001**	**0.0001**
MR+IE				-	0.2098	0.6586
Search					-	0.5031

Table 8: P-Values from Pairwise t-Tests for Difference Between Mean Intelligibility Rank on TDT2 Topic 89, using Actual IE.

to score the sentences, including the IE groups.

Search The RIPTIDES overview, except with the IE groups zeroed out in the scoring.

Search+IE The RIPTIDES overview.

The marginal relevance systems (MR and MR+IE) used a simple selection mechanism which does not involve search, inspired by the maximal marginal relevance (MMR) approach (Goldstein et al., 2000). This selection mechanism begins by selecting the top ranking sentence for inclusion, then determines whether to include the second ranking sentence depending on whether it is sufficiently dissimilar from the first one, based on comparing the SimFinder similarity measure against a threshold, and likewise for lower ranked sentences, comparing them against all sentences included so far, up to the word limit. The selected sentences were then gathered into blocks of adjacent sentences, and ordered chronologically.[2]

We ran the six systems on two versions of the input document sets for each of the five time points, one with the manual IE annotations (simulated IE) and one with the automatic IE annotations (actual IE). Note that the first three systems produced essentially identical output for both versions, since they did not depend on the IE annotations and did not involve randomized search. Next, for each document set, we had two judges[3] rank the summaries from best to worst, with ties allowed, in two categories, content and intelligibility. In the case of ties, the tied systems shared the appropriate ranking; for example, if two summaries tied for the best content, each received a rank of 1, with the next best summary receiving a rank of 3.

The charts in figures 3 and 4 show the system rank for content and intelligibility for the simulated IE and actual IE versions of the document sets, respectively, averaged across the two judges and five time points. Tables 1 through 4 list all the judgements together with their means and standard deviations.

In general, we found that Search and Search+IE provided substantial improvements in both content and intelligibility over Last, Leads and MR, with the exception that Last was always ranked first in intelligibility. Search+IE also appeared to show a small further improvement in content.

Determining the significance of the improvements is somewhat complex, due to the small number of data points and the use of multiple comparisons. To judge the significance levels, we calculated pairwise t-tests for all the means listed in tables 1 through 4, and applied the Bonferroni adjustment, which is a conservative way to perform multiple comparisons where the total chance of error is spread across all comparisons. With the total α equal to 0.05, the Bonferroni adjustment provides a 95% confidence level that all the pairwise judgements are correct. In our case, a total α of 0.05 corresponds to an individual α of 0.0033, which is difficult to exceed with a small number of data points.

The p-values for the t-tests appear in tables 5 through 8. Turning first to the content rankings, with the simulated IE (table 5), we found that both Search and Search+IE scored significantly higher than Last, Leads and MR. While the difference between Search and Search+IE was not significant, only Search+IE achieved a significantly higher average rank than MR+IE. With the actual IE (table 7), Search and Search+IE again scored significantly higher than Last and Leads; and, although these two systems did not show a significant improvement over MR, both systems did improve significantly over Leads and MR+IE, in contrast to MR.

Turning now to the intelligibility rankings, with both the simulated and actual IE (tables 6 and 8), we found that Search and Search+IE improved significantly over Leads and MR. The difference between Search and Search+IE was not significant. Surprisingly, MR+IE scored significantly higher than MR, and not significantly worse than Search and Search+IE.

[2] In trying out the MR systems on all the articles in TDT2 topic 89, we found chronological ordering to usually be more coherent than importance ordering.

[3] The authors were the judges.

4 Discussion

We were pleased with the substantial improvements in both content and intelligibility that our randomized local search method provided over the DUC-like baselines and the simplest variant of our system, the one using marginal relevance selection and no IE groups (with the exception that the last article baseline was always ranked first in intelligibility). We did not expect to find that the selection search would yield substantial improvements over marginal relevance selection in the content rankings, since the search method was designed to improve intelligibility without unduly affecting content. At the same time though, we were somewhat disappointed that the use of the IE groups appeared to only contribute a small further improvement in content when used with our selection search.

It is not entirely clear why our selection search method led to improvements in the content rankings when compared to the marginal relevance variants. One possibility is that the randomized local search was able to find sentences with greater information density. Another possibility is that the use of hard thresholds by the marginal relevance variants led to some poor sentence selections fairly far down on the list of ranked sentences.

It is also not clear why the IE groups did not help more with content selection. It may well be that a more elaborate evaluation, involving more systems and judgements, would indeed show that the IE groups yielded significant improvements in content rankings.

On the intellibility side, we were surprised to find that the IE groups led to improvements in intelligibility when used with marginal relevance selection. One likely explanation for this improvement is that this system variant jumped around less from topic to topic than its counterpart that did not make use of the IE info.

Another question is why the selection search did not yield further improvements in intellibility. One reason is that the search method always selected sentences up to the word limit, even when this yielded highly repetitive summaries — as was the case with the first two test sets, which only contained a handful of articles. Another reason is that the search routine was prone to selecting a couple of sentences from an article that was largely off topic, only containing a brief mention of the quake.

These deficiencies point to possible improvements in the search method: informativeness could perhaps be balanced with conciseness by deselecting sentences that do not improve the overall score; and off-topic sentence could perhaps be avoided by taking into account the centrality of the document in the sentence scores. More speculatively, it would be interesting to extend the approach to work with sub-sentential units, and to make use of a greater variety of inter-sentential cohesive links.

References

Regina Barzilay, Noemie Elhadad, and Kathleen McKeown. 2001. Sentence Ordering in Multidocument Summarization. In *Proceedings of the First International Conference on Human Language Technology Research*, San Diego, CA.

D. Bikel, S. Miller, R. Schwartz, and R. Weischedel. 1997. Nymble: A High-Performance Learning Name-Finder. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, pages 194–201, San Francisco, CA. Morgan Kaufmann.

C. Cardie. 1997. Empirical Methods in Information Extraction. *AI Magazine*, 18(4):65–79.

Eugene Charniak. 1999. A maximum-entropy-inspired parser. Technical Report CS99-12, Brown University.

D. Day, J. Aberdeen, L. Hirschman, R. Kozierok, P. Robinson, and M. Vilain. 1997. Mixed-Initiative Development of Language Processing Systems. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*. Association for Computational Linguistics.

C. Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA.

J. Goldstein, V. Mittal, J. Carbonell, and M. Kantrowitz. 2000. Multi-document summarization by sentence extraction. In *Proceedings of the ANLP/NAACL Workshop on Automatic Summarization*, Seattle, WA.

R. Grishman. 1996. TIPSTER Architecture Design Document Version 2.2. Technical report, DARPA. Available at `http://www.tipster.org/`.

Donna Harman. 2001. *Proceedings of the 2001 Document Understanding Conference (DUC-2001)*. NIST.

Vasileios Hatzivassiloglou, Judith L. Klavans, Melissa L. Holcombe, Regina Barzilay, Min-Yen Kan, and Kathleen R. McKeown. 2001. Simfinder: A flexible clustering tool for summarization. In *Proceedings of the NAACL 2001 Workshop on Automatic Summarization*, Pittsburgh, PA.

Daniel Marcu. 2001. Discourse-Based Summarization in DUC-2001. In *Proceedings of the 2001 Document Understanding Conference (DUC-2001)*.

E. Riloff. 1996. Automatically Generating Extraction Patterns from Untagged Text. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pages 1044–1049, Portland, OR. AAAI Press / MIT Press.

Bart Selman and Henry Kautz. 1994. Noise Strategies for Improving Local Search. In *Proceedings of AAAI-94*.

Michael White, Tanya Korelsky, Claire Cardie, Vincent Ng, David Pierce, and Kiri Wagstaff. 2001. Multidocument Summarization via Information Extraction. In *Proceedings of the First International Conference on Human Language Technology Research*, San Diego, CA.

Michael White, Claire Cardie, Vincent Ng, and Daryl McCullough. 2002. Detecting Discrepancies in Numeric Estimates Using Multidocument Hypertext Summaries. In *Proceedings of the Second International Conference on Human Language Technology Research*, San Diego, CA. To appear.