Automated Multi-document Summarization in NeATS

Chin-Yew Lin and Eduard Hovy
USC Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292-6695
Tel: +1-310-448-8711/8731
{lin,hovy}@isi.edu

ABSTRACT
This paper describes the multi-document text summarization system NeATS. Using a simple algorithm, NeATS was among the top two performers of the DUC-01 evaluation.

Keywords
Multi-document text summarization; NeATS

1. OVERVIEW
In this paper we describe work in our NeATS (Next Generation Automated Text Summarization) project on multi-document summarization. To select important content, we used techniques that proved effective in single document summarization such as sentence position [1], term frequency [13], topic signature [11,7], and term clustering. To remove redundancy, we used MMR [3]. To improve cohesion and coherence, we used stigma word filters [4] and time stamps. Although most of the individual techniques are not new, assembling them and applying them to multi-document summarization is new. Also, including lead sentences to ensure coherence is new, and turned out to be important.

For much of the system we re-used modules built in prior work, notable the SUMMARIST single-document summarizer [7] and the Webclopedia question answering system [8,9].

NeATS was evaluated in the Document Understanding Conference DUC-01 [15]. It consistently was among the top performers in the multi-document summarization track. In the aggregating peer-to-peer comparison suggested by [14], NeATS scored first in precision and second in recall among 12 participants. The system also achieved the best F1-measure score across all summary sizes.

2. NeATS
NeATS attempts to extract relevant or interesting portions from a set of documents about some topic and to present them in coherent order. It is tailored to the genre of newspaper news articles, and it works for English, but can be made multilingual without a great deal of effort. At present NeATS produces generic (author's point of view) summaries, but it could be made sensitive to desired focus topics, input by a user.

Given an input of a collection of sets of newspaper articles, NeATS applies the following 6 steps.

3. ALGORITHM

3.0 Input
The input is a set of topic groups. Each topic group is a set of approx. 10 newspaper articles selected by the evaluation organizers. A topic group may focus on a single natural disaster (earthquake, hurricane, etc.), a single event (election, car race, etc.), multiple instances of a type of event (many earthquakes, elections, etc.), or a single person (in which case the summary would be a biography).

3.1 Extract and Rank Passages
Given the input documents, form a query, extract sentences, and rank them, using modules of Webclopedia:

1.a identify key words for each topic group: compute unigram, bigram, and trigram topic signatures [11] for each group, using the likelihood ratio [2]. A topic signature is a list of words/phrases, each with strengths, that characterizes the group and differentiates it from others [7]

1.b to facilitate fallback (query generalization), remove from the signatures all words or phrases that occur in fewer than half the texts of the topic group

1.c save the signatures in a tree, organized by signature overlap. We use the format of Webclopedia's parser CONTEX [5,6]; see Figure 1

1.d use Webclopedia's ranking algorithm to rank sentences [8].

3.2 Filter for Content
Given the ranked list of sentences, re-rank or remove those according to the following conditions:

2.a remove all sentences with sentence position > 10. This is a simple version of SUMMARIST's Optimum Position Policy (OPP) [10], which records the relative importance of sentence positions

2.b decrease ranking score of all sentence containing stigma words [4] (day names; time expressions; sentences starting with conjunctions such as "but", "although"; sentences containing quotation marks; sentences containing the verb "say").

3.3 Enforce Cohesion and Coherence

Locate and include a suitable introductory sentence for each remaining sentence:

3.a pair each sentence with the first sentence (lead) of its document; but if the first sentence contains fewer than 5 words, then take the next one. For example (where *x.y* stands for *document number . sentence number*):

4.3, 6.6, 2.5, 5.2...

 4.1, 4.3, 6.1, 6.6, 2.1, 2.5, 5.1, 5.2...

3.4 Filter for Length

Select the required number of sentence pairs using a simplified version of CMU's MMR algorithm [3]:

4.a include first pair

4.b using a simplified version of MMR, find the sentence pair most different from the included ones, and include it too. (In the DUC-2001 implementation, NeATS did not consider the sentence pair, just the sentence. This caused some degradation.)

4.c repeat step 4.b until the summary length criterion is satisfied:

 4.1, 4.3, 2.1, 2.5

3.5 Ensure Chronological Coherence

Reorder the pairs in publication order, and disambiguate all time words with explicit dates:

5.a reorder pairs in publication order:

 2.1, 2.5, 4.1, 4.5

5.b for each time word ("today", "Monday", etc.) compute the actual date (from the dateline) and include it in the text in parentheses, in order to signal which day each "today" (etc.) is.

3.6 Format and Print Results

Format and output the final result.

4. DISCUSSION

This simple algorithm gives surprisingly reasonable results. We like the following aspects.

Typical current extractive summarization methods are essentially IR in miniature: from a set of sentences (instead of texts), select and rank the ones most relevant to the query. The major problems are **creating the query** and then **assembling the extracted sentences into a single coherent text** (a step that IR does not have).

For creating the query, we saved a great deal of development time by using existing modules from SUMMARIST [7] and Webclopedia [8,9]. SUMMARIST's topic signature creation techniques [11] allowed us directly to compute a ranked list of words (and bi- and trigrams) most characteristic of each document set. By placing these ngrams (and their sub-ngrams, which form a cluster) into the parse tree format we use for the retrieval stage of Webclopedia (Figure 1), we could directly form increasingly general queries, with which to extract the most relevant sentences from the document set, and rank them.

To assemble the extracted sentences into a single coherent text, we used the fact that a lead sentence, which introduces the article, is a powerful context-setter for each nearby (early) sentence in the article. We therefore paired each extracted sentence with its lead sentence, selected as appropriate.

One further factor interfering with coherence was misleading time words: "today" in articles written on different days means different dates. To disambiguate all time words we therefore computed the actual dates from the articles' datelines and included them after each time word. A typical summary is shown in Figure 2. Note that the sentences span 4 years; without the absolute time references, a very misleading picture of the documents would have been created.

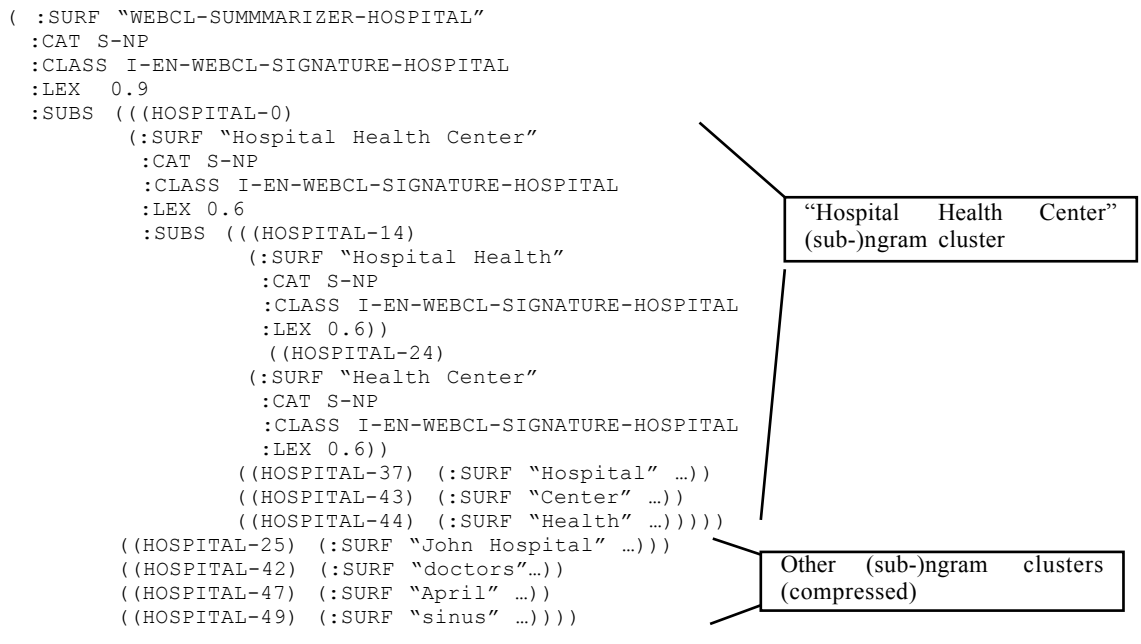

Figure 1. Portion of topic signature cluster tree for ngrams of "Hospital Health Center".

```
<multi size="100" docset="d45h">
(06/25/90) The republic of Slovenia plans to
  begin work on a constitution that will give
  it full sovereignty within a new Yugoslav
  confederation, the state Tanjug news agency
  reported Monday (06/25/90).
(06/28/91) On Wednesday (06/26/91), the
  Slovene soldiers manning this border post
  raised a new flag to mark Slovenia's
  independence from Yugoslavia.
(06/28/91) Less than two days after Slovenia
  and Croatia, two of Yugoslavia's six
  republics, unilaterally seceded from the
  nation, the federal government in Belgrade
  mobilized troops to regain control.
(02/09/94) In the view of Yugoslav diplomats,
  the normalization of relations between
  Slovenia and the Federal Republic of
  Yugoslavia will certainly be a strenuous and
  long-term project.
</multi>
```

Figure 2. Example 100-word summary (Slovenia's secession from Yugoslavia).

5. RESULTS

We were pleased by the content and readability of the results. Analyzing all systems' results for DUC-2001, we computed Recall, Precision, and F-Measure using the following formulas:

Recall = (# of model units marked with peer units) / (# of model units)

Precision = (# of unique peer units marked with model units) / (# of peer units)

F-Measure = 2*Recall*Precision / (Recall+Precision)

Here 'model unit' denotes an evaluation unit (usually, a sentence) contained in the human-produced model summary and 'peer unit' a unit produced by the system. The DUC organizers used the Summary Evaluation Environment SEE built by one of the authors [12] to rate the relevance of all units in the system summary by comparing them to each model unit.

According to this, NeATS (system N) did not fare badly (though its relative rank may change with different definitions of Recall and Precision). Systems' scores using these formulas are shown in the histogram in Figure 3. Humans did better than any system (both humans over 50%, human X is the average of human 1 and human 2), outscoring the nearest system by about 10%. Only 1 system (NeATS) scored in the mid-40s, with 45%. 5 systems scored between 35% and 41%, and 3 scored between 30% and 35%. Despite the low inter-human agreement (which we take to reflect the undefinedness of the 'generic summary' task), there is obviously still considerable room for systems to improve. We expect that systems that compress their output (unlike NeATS) will thereby gain more space to include additional important material.

NeATS tended to perform best on single-event stories and general topics, across the scale on biographies, and not so well on multi-instance events. A somewhat more targeted strategy is called for in topic groups with internal structure such as the latter two types.

It is interesting to note that systems are separated into two major groups by baseline 2. Baseline 2 forms its summaries by taking the first sentence in the first document, the first sentence in the second document, and so on until the number of sentences in the summary reaches the 50, 100, 200, or 400 word limits. Baseline 1 takes the first 50, 100, 200, 400 words in the last document (by date) of the collection. Almost all systems outperform baseline 1. This result indicates that on average it is necessary to cover most documents in a collection to generate good multi-document summaries.

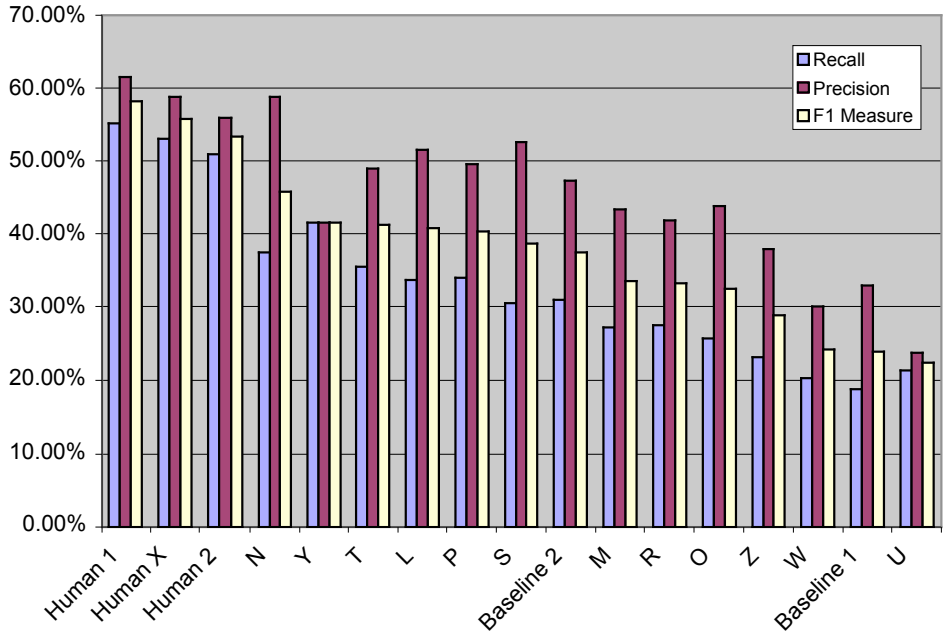

Figure 3. DUC-01 recall, precision, and F1 scores.

6. REFERENCES

[1] Baxendale, P.B. 1958. Machine-Made Index for Technical Literature—An Experiment. *IBM Journal* (October):3, 54–361.

[2] Dunning, T. 1993. Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics* 19, 61–74.

[3] Goldstein, J., M. Kantrowitz, V. Mittal, and J. Carbonell. 1999. Summarizing Text Documents: Sentence Selection and Evaluation Metrics. *Proceedings of the 22nd International ACM Conference on Research and Development in Information Retrieval (SIGIR-99)*, Berkeley, CA, 121–128.

[4] Edmundson, H.P. 1969. New Methods in Automatic Abstracting. *Journal of the Association for Computing Machinery* 16(2).

[5] Hermjakob, U. 1997. *Learning Parse and Translation Decisions from Examples with Rich Context*. Ph.D. dissertation, University of Texas at Austin. file://ftp.cs.utexas.edu/pub/~mooney/papers/hermjakob-dissertation-97.ps.gz.

[6] Hermjakob, U. 2000. Rapid Parser Development: A Machine Learning Approach for Korean. *Proceedings of the North American chapter of the Association for Computational Linguistics* (NAACL-2000). http://www.isi.edu/~ulf/papers/kor_naacl00.ps.gz.

[7] Hovy, E.H. and C.-Y. Lin. 1999. Automated Text Summarization in SUMMARIST. In M. Maybury and I. Mani (eds), *Advances in Automatic Text Summarization*. Cambridge, MA: MIT Press.

[8] Hovy, E.H., L. Gerber, U. Hermjakob, M. Junk, and C.-Y. Lin. 2000. Question Answering in Webclopedia. *Proceedings of the TREC-9 Conference.* NIST, Gaithersburg, MD. November 2000.

[9] Hovy, E.H., L. Gerber, U. Hermjakob, C.-Y. Lin, and D. Ravichandran. 2001. Toward Semantics-Based Answer Pinpointing. *Proceedings of the DARPA Human Language Technology Conference (HLT)*. San Diego, CA. March 2001.

[10] Lin, C.-Y. and E.H. Hovy. 1997. Identifying Topics by Position. *Proceedings of the 5th Conference on Applied Natural Language Processing* (ANLP). Washington, DC.

[11] Lin, C.-Y. and E.H. Hovy. 2000. The Automated Acquisition of Topic Signatures for Text Summarization. *Proceedings of the COLING Conference*. Saarbrücken, Germany. August, 2000.

[12] Lin, C.-Y. 2002. The SEE Summarization Evaluation Environment. In prep.

[13] Luhn, H.P. 1959. The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*: 159–165. Also in I. Mani and M. Maybury (eds), *Advances in Automated Text Summarization*. 1999. Cambridge, MA: MIT Press, 15–22.

[14] McKeown, K., R. Barzilay, D. Evans, V. Hatzivassiloglou, M-Y Kan, B, Schiffman, and S. Teufel 2001. Columbia Multi-Document Summarization: Approach and Evaluation. *Workshop on Text Summarization*. New Orleans, LA. September, 2001.

[15] Over, P. 2001. Introduction to DUC-2001: An Intrinsic Evaluation of Generic News Text Summarization Systems. *Workshop on Text Summarization* at the SIGIR Conference. New Orleans, LA. September, 2001.