Facilitating Email Thread Access by Extractive Summary Generation

Ani Nenkova
Columbia University
Computer Science Department
New York, NY 10027, USA
ani@cs.columbia.edu

Amit Bagga
Avaya Labs Research
233 Mt Airy Road
Basking Ridge, NJ 07920, USA
bagga@avaya.com

Abstract

Email threads are the most common way to represent (archived) discussion groups or mailing lists. Due to the large volumes of such archives, better representation of the threads is required in order to allow the users to find topics of interest and to decide which threads to read. This paper discusses our initial approach to generating thread overviews that serve as indicative summaries for the thread. The overviews give the user a better idea of what is discussed in a related set of email exchanges than the currently used conventions can provide. A relatively large user study on fifty email threads was performed and it confirmed the utility of the proposed approach.

1 Introduction

Mailing lists and discussion groups are becoming increasingly popular. They contain a lot of potentially useful or interesting information (Millen 00) but finding relevant information might be a daunting task. The main reason for this is that whether reading current postings or past archives, one has few clues what exactly is discussed in an email thread before actually reading all the postings in it. One of the most common representations of mailing list archives is a sequence of the threads in the archive, where each thread is shown as an indented list with subject lines and sender's name, time of posting and number of follow-ups for the message. Even when the subject of the initial posting (root) of the thread is well chosen and informative, the indented representation is not very helpful since the subjects of all follow-up messages are simply "Re: Original Subject". Hypermail[1] and MHonArc[2] are the two most commonly used programs for mailing list archiving and they create browsable representations of the kind just described.

A somewhat better representation is a more hierarchical listing, where the first level contains the subject of the first email in each thread (the root) and the number of postings in the thread and then when a thread/subject is chosen, the indented representation of just this thread is shown in a new window where the messages can be opened and viewed one at a time. The introduction of this extra level gives the user the ability to quickly skim through the initial subject lines and choose if there is something interesting to read. LISTSERV[3] provides archives of this kind. But, again, even when the subject for the initial posting is well-chosen, it can rarely give the user a good description of what is being discussed.

Google[4] groups further improve on that by using a two-level representation. The first level is similar to the first level in LISTSERV archives and contains date of posting, subject of *thread initial* email, name of most recent poster and number of messages in the thread. The second level consists of a two frame page showing the indented structure and a digest of the messages on one screen. The digest consists of the concatenated bodies of the first ten messages posted in the thread. The Google groups representation therefore allows a user to find topics of interest and to read the threads corresponding to each topic without having to click on every message.

In this paper we describe a more efficient representation that allows a user to decide which threads to read without browsing the actual content of the thread. We call this a *thread overview* and it consists of an extractive summary for the documents at the first two levels of the discussion thread tree. The overviews are relatively short and the user can skim through them in order to find threads of interest.

2 Related Work

The need for better and easier access to email, discussion groups, and mailing lists has given rise to several directions of research. A substantial body

[1] http://www.hypermail.org/
[2] http://www.oac.uci.edu/indiv/ehood/mhonarc.html
[3] http://www.lsoft.com/products/
[4] http://groups.google.com/

of work addresses the problem of visualization and how it can facilitate access and navigation. Conversation map (Sack 00), for example, is an interface for discussion archive browsing. The foci in its development are social, semantic and pragmatic aspects of the discussion and how these can be visualized. The system computes links of interaction – who responds to whom, finds "authorities" in the discussion and also creates semantic networks representing similarities and connections between topics discussed in the list. (Smith & Fiore 01) and (Donath *et al.* 99) also discuss issues with visualizing the huge amount of useful metadata that can be gathered from discussion lists. One of the problems common to all three of these approaches is that representations often involve diagrams, graphs and semantic networks whose interpretation is not intuitive for users.

Work by (Muresan *et al.* 01) shows an approach to summarizing single emails (not threads) by key phrase extraction. A corpus of email was manually labeled and machine learning techniques based on linguistic features were used to train a system that outputs noun phrases that represent the gist of an email.

Newman works on a project that attempts to provide overall archive characterizations plus a better representation of the individual threads (Newman 02b; Newman 02a). A portion of the essential text of each email is presented in the first level of overview. She also deals with visualization issues and the approach seems promising but no user studies have been conducted so far to evaluate the system.

All the work in the field of multi-document summarization (Goldstein *et al.* 00; Lin & Hovy 02; McKeown *et al.* 99; Radev *et al.* 00; Schiffman *et al.* 02) is very related to the problem of creating an overview for an email thread. A lot of research has been done on informative summarization for newswire, but it seems that indicative summarization is what is more needed in the scenario that we discuss. Informative summaries are intended to serve as surrogates for the original document(s), while indicative summaries aim at providing an idea about what is discussed in the document(s) and rather than substituting the original(s) they are supposed to help the user decide if the document(s) are worth retrieving and reading. (Kan *et al.* 01) discusses how domain specific indicative and informative summaries can be generated and how they compliment each other.

3 Task and Corpus

Taking into account all the previous discussion on the problem, the current practices and related research, we felt that an indicative summary representation of an email thread can facilitate the retrieval of relevant information from mailing list and mail-based discussion groups. Therefore, we wanted to generate indicative summary representations or overviews of threads that can provide a better idea of the problem discussed than the original subject line alone could give.

Since automated subject/headline generation (Banko *et al.* 00) is a hard problem and has been currently addressed only in single document setting, extractive summary generation seemed like a good way to proceed. In other words, extracting one informative sentence per email and using that as the subject would provide a preview of the content of an email. Such an approach complements the approach taken by (Newman 02b) where parts of messages are shown as part of an overview page. An obvious starting point would be to use the first sentence of an email as its subject. However, unlike newswire, first sentences in email can often be greetings, quotes from a previous message, header information, etc., and therefore they, in general, may not be informative. Nevertheless, we used a variant of first sentence extraction as a baseline for comparing the performance of our approach (described further in Section 5).

3.1 Corpus

We chose to work with the Pine-Info mailing list which contains emails regarding "features, bugs and workarounds, usage, installation, customization and more pertaining to the Pine software"[5]. The choice of this list was deliberate as the discussion there is very focused and usually problem-solving oriented and our approach is targeted toward discussion lists of this kind. In comparison, the discussion on general topics can be much more loosely related and far more difficult to process successfully.

The length of discussion paths (a single branch of the discussion tree) and their branching factor (the maximum number of answers a message gets)

[5] http://www.washington.edu/pine/pine-info/

depth	2	3	4–7	8-21
branches	9999	1930	1586	189

Table 1: Length of paths found in threads. The second row shows the number of branches (paths) with the specified length.

successors	1	2	3	4-7
messages	12058	2908	577	169

Table 2: Branching factors in threads. Leaf messages with no succesors are not shown.

can vary significantly from one discussion group to another. An analysis of the corpus showed that neither the depth nor the branching factor of the threads were very large. Table 1 shows that paths in threads from the list tend to be short, with the majority including just a root and a follow-up to it (depth 2) and that chains of replies longer than 4 are not typical. Threads in the list are also not very bushy, with most messages receiving just one or two replies. The exact figures can be seen in Table 2.

The shallow, thinly branched threads in this corpus are the result of the focused nature of the list. The initial emails are usually requests for help and they often receive direct answers. This makes the top level of the discussion tree (the root plus the first level of answers) very useful since they often contain statements of problems and their solutions. We should however note that not all threads start with a problem question followed by a series of answers. There are numerous cases where the initial email starts a general discussion, solicits opinions, reports bugs, suggests new features, etc. The archive provided us with 2389 threads to work with.

4 Generating Thread Overviews

As described earlier, our goal was to generate an indicative summary representation of a thread by extracting, from each email in the thread, a sentence which substitutes its subject line. Because of the problem solving nature of the discussion list, we noticed that a further reduction can be made by extracting a sentence only from the thread root message and its immediate follow-ups. These extracts ideally contain a statement of the problem and a suggestion for its solution, they are easy to read and also they give the user sufficient information on the topic of the thread so that the user can decide if he needs to read the entire thread. Thus, we do not use a fixed compression rate in terms of words for the produced summaries, but rather aim at a more flexible stratagy in which one sentence per message is chosen.

Figure 1 shows an example of an indicative summary of a thread. The first line, in bold, is the original subject line of the root message. Below this line is the summary subject line from the root followed by two summary subject lines from the two replies.

The algorithms used to generate such an indicative summary are described below.

4.1 Extracting sentences from roots

Often initial postings have well-chosen subjects. In order to find the sentence in the root email that contains the heart of the problem, rather than background or introduction information, we find the shortest sentence in the email that has the largest overlap of nouns, verbs, adjectives or adverbs with the subject of the message. There are four stages in the algorithm:

1. Clean any existing quotation and signature blocks from the root message.

2. The message and its subject are processed with LT POS (Mikheev 96). Sentence boundaries and parts of speech are assigned.

3. Every noun or verb is substituted by its noninflected lexical form, obtained from WordNet (Miller *et al.* 90).

4. Each sentence is assigned a score equal to $overlap_{subj}/length_{sent}$, where $overlap_{subj}$ is the overlap of noninflected forms of verbs, nouns, adjectives and adverbs in the subject and the sentences, and $length_{sent}$ is the number of words with such parts of speech in the scored sentence. In the case of ties, the sentence with highest score that appears first in the body of the message is chosen.

The sentence with highest score is extracted. The normalization to sentence length was aimed at picking shorter sentences. Very long sentences are not easy to skim and also for such sentences the probability of larger overlap is naturally higher. We will return to the issue of normalizing the sentence length for the root in Section 5.

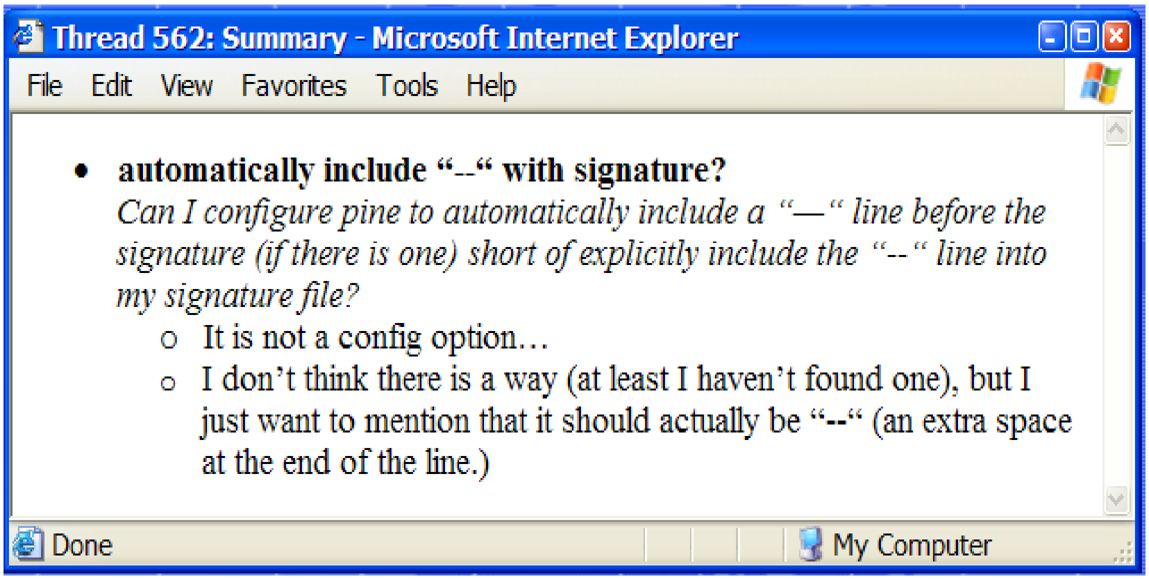


Figure 1: Thread Summary Example

4.2 Extracting sentences from follow-ups

In this stage, the overlap in terms of verbs and nouns between the root message and each of the sentences of the follow-up message is computed.

1. Clean any existing quotation and signature blocks in both the root and the follow-up messages.

2. Both the root mail and its follow-up are processed with LT POS to get part-of-speech tags and sentence boundaries. Using WordNet, nouns and verbs are converted to their noninflected forms.

3. For each sentence in the follow-up, a weight is computed equal to the overlap of nouns and verbs between the root email and the sentence.

4. The sentence with highest score is extracted as a "main topic" for the follow-up message. Again, ties are decided by giving preference to the sentence with the highest score that appears earliest in the message.

Thus, the root message is taken as background and the sentence most relevant to the background in the follow-up is extracted. This approach helps make the overviews more cohesive as it ensures that the subjects of the follow-ups are related to the subject of the root.

	GOOD	BAD
roots	58%	42%
follow-ups	64%	36%

Table 3: Rating of the appropriateness of the extracted sentence per email. GOOD means the evaluator could form an expectation about the subject matter of the message and BAD means the summary sentence did not help at all.

	GOOD	BAD
before reading messages	74%	26%
after reading messages	68%	32%

Table 4: Rating of how indicative the overviews were both before the messages were actually read and afterwards. GOOD and BAD have the same interpretations as in 3

An example of a thread overview and the messages it has been extracted for can be seen on Figure 2 and Figure 3.

4.3 Generalizing for emails further down the tree

Generalizing the approach for messages deeper in the discussion tree is straightforward. In this case, the background is taken to be the concatenation of all the messages preceding the currently processed mail up to the root. Once again, the sentence with highest score is the sentence with highest overlap of nouns and verbs.

Figure 2: Thread Overview Example

5 Evaluation and Results

Evaluating machine generated summaries in an automated way is currently an unsolved problem. Therefore, we used human subjects to judge the indicative summaries produced by our system. We used fifteen human subjects to evaluate fifty randomly selected threads containing 161 messages total. Each thread was evaluated by three subjects and the majority decision was taken as final.

We asked the users to evaluate the summaries in three dimensions:

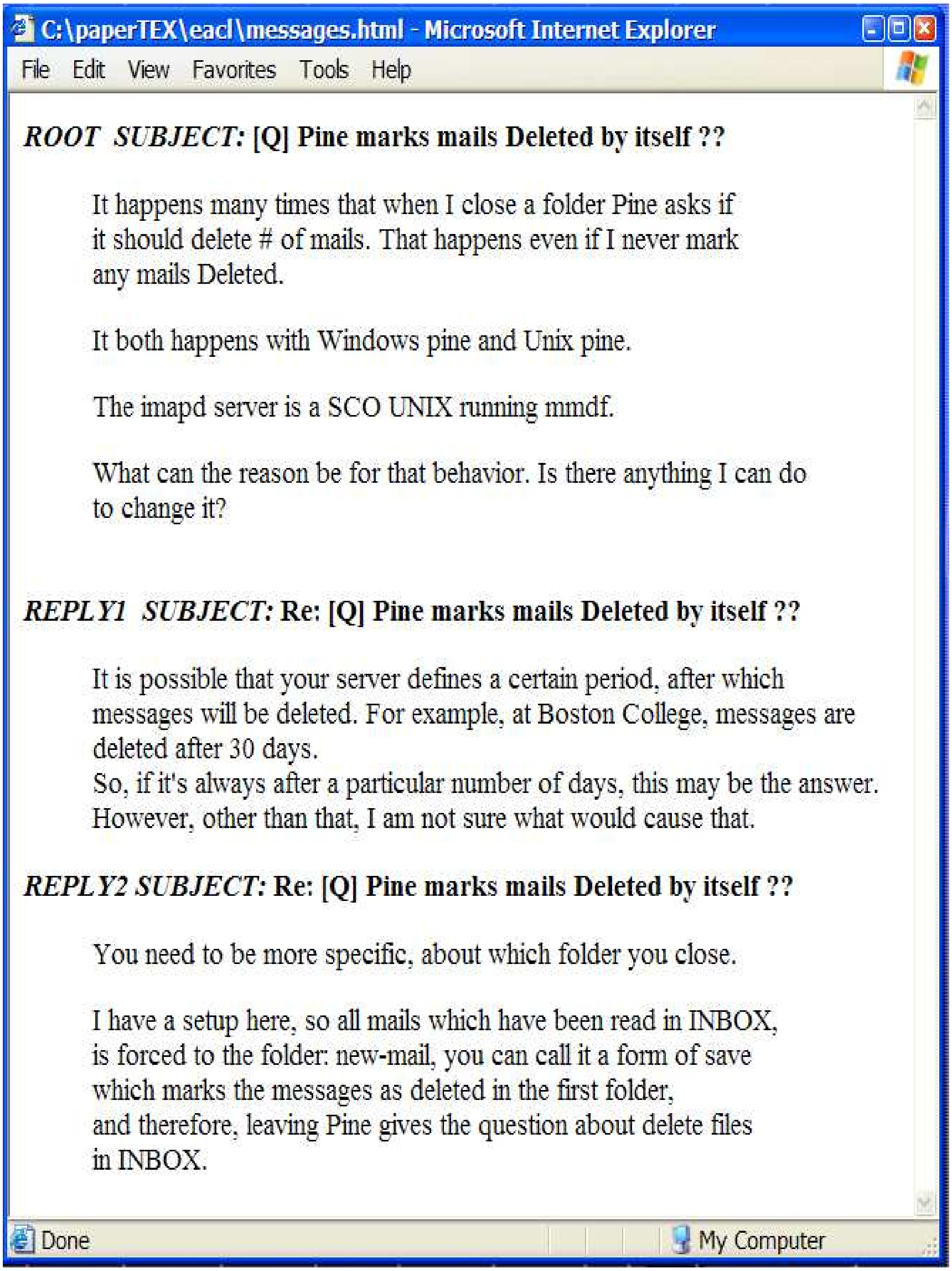
1. How informative was the overview? How good idea does it give you about what is being discussed in the thread?

2. Was the summary subject line of the root message appropriate? Can you get from it an idea of what the message is about?

3. Were the summary subject lines of each of the responses appropriate?

The first question is definitely different from the other two because it is possible that well chosen subject sentences do not make up a good overview, or alternatively, the overview can be informative even when the most appropriate sentences are not chosen as the subject for either the root or one or more of the follow-ups. In order to capture this variation we asked our subjects to first judge threads as a whole and then separately judge the appropriateness of each of the subject sentences in the overview with respect to the email it is extracted from. In addition, since it is possible that overviews may suggest a misleading topic of discussion, we asked the human judges to give their opinion on the informativeness of the thread in two scenarios. First, they were asked, before reading any of the messages in the thread, to read the overview and decide if they can form an idea of what will be discussed in the thread. Then, after reading the messages, they gave their opinion again, now acquainted with the actual content of the message. Opinions did change – out of the 150 total judgments of the overviews, there were 22 changes of opinion where an overview first seen as informative was judged misleading after reading the whole thread, and there were 7 changes in the opposite direction.

Table 3 shows the ratings on the appropriateness of the extracted sentences for each of the messages in the fifty threads. A closer look at the numbers showed that the performance for root messages was lower than that of the follow-ups. A subsequent analysis showed that normalizing the sentence length resulted in concise but not necessarily the most appropriate sentences being picked as the subject. Table 4 shows the ratings on how indicative the overviews really were. The first observation here is that the numbers for the indicativeness of the overviews are much higher



C:\paperTEX\eacl\messages.html - Microsoft Internet Explorer

File Edit View Favorites Tools Help

ROOT SUBJECT: **[Q] Pine marks mails Deleted by itself ??**

It happens many times that when I close a folder Pine asks if
it should delete # of mails. That happens even if I never mark
any mails Deleted.

It both happens with Windows pine and Unix pine.

The imapd server is a SCO UNIX running mmdf.

What can the reason be for that behavior. Is there anything I can do
to change it?

REPLY1 SUBJECT: **Re: [Q] Pine marks mails Deleted by itself ??**

It is possible that your server defines a certain period, after which
messages will be deleted. For example, at Boston College, messages are
deleted after 30 days.
So, if it's always after a particular number of days, this may be the answer.
However, other than that, I am not sure what would cause that.

REPLY2 SUBJECT: **Re: [Q] Pine marks mails Deleted by itself ??**

You need to be more specific, about which folder you close.

I have a setup here, so all mails which have been read in INBOX,
is forced to the folder: new-mail, you can call it a form of save
which marks the messages as deleted in the first folder,
and therefore, leaving Pine gives the question about delete files
in INBOX.

Done

My Computer

Figure 3: Messages for which the overview in Figure 2 was generated

overview	baseline	no pref	no majority
54%	16%	20%	10%

Figure 4: Preference of the overview or the baseline based on their informativeness.

than those for the appropriateness of the sentences. The figure also shows that there was a net 6% change of opinion in the negative direction. While the final numbers in this figure are not in the desired 80% plus range, we feel that when compared to the existing schemes of repeating the subject line of the root messages, our representation provides a huge improvement with 68% of the overviews actually being judged as good.

The judges were also asked to compare the thread overview generated by sentence extraction and a baseline overview generated by using the first sentence in a message. However, as noted earlier in the paper, the first sentence of a message can consist of quotes from previous messages in the thread, greetings, etc. Therefore, in order to make the comparison realistic, we manually chose the first "useful" sentence of the message. Figure 4 shows that the overview generated by our system was better than or equal to the baseline 74% of the time. The three judges could not form a majority opinion in 10% of the cases while in 16% of the cases they preferred the baseline overview.

6 Discussion

There are several aspects of email that need to be taken it account. Good "clean-up" of messages, e.g. getting rid of signature blocks, quotes from previous messages etc is needed in order to ensure accurate performance of the overview generation. For the experiments described in this paper, we developed our own simple procedures for clean-up, but any further work will require more serious effort in this direction. Also, conventional methods for end of sentence detection do not work for discussion group/mailing list corpora because of the numerous uses of periods in file extensions, email addresses, and URLs, as well as the erratic usage of punctuation and capitalization in email messages. Thus, special retraining or fine-tuning of already available tools needs to be done when they are to be used for email processing.

7 Conclusions and Future work

The user study evaluation shows that extractive techniques can help enhance access to discussion group archives. A natural next step will be to try multi-document summarization approaches to the messages at the same level of the discussion tree. In order to get maximum benefit from the thread overviews, the extraction of sentences needs to be combined with visualization techniques.

References

(Banko *et al.* 00) M. Banko, V. Mittal, and M. Witbrock. Headline generation based on statistical translation. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, 2000.

(Donath *et al.* 99) Judith Donath, Karrie Karahalios, and Fernanda Viegas. Visualizing conversations. In *Proceedings of the Hawaii International Conference on System Sciences 32*, 1999.

(Goldstein *et al.* 00) J. Goldstein, V. Mittal, J. Carbonell, and M. Kantrowiz. Multi-document summarization by sentence extraction. In *Proceedings of ANLP/NAACL Workshop on Summarization*, 2000.

(Kan *et al.* 01) Min-Yen Kan, Kathleen R. McKeown, and Judith L. Klavans. Domain-specific informative and indicative summarization for information retrieval. In *Proceedings of the Document Understanding Workshop (DUC 2001)*, pages 19–26, 2001.

(Lin & Hovy 02) Chin-Yew Lin and E.H. Hovy. Automated multi-document summarization in neats. In *Proceedings of the Human Language Technology Conference (HLT2002)*, 2002.

(McKeown *et al.* 99) Kathleen McKeown, Judith Klavans, Vasilis Hatzivassiloglou, Regina Barzilay, and Eleazar Eskin. Towards multidocument summarization by reformulation: Progress and prospects. In *Proc. of AAAI, 1999*, 1999.

(Mikheev 96) A. Mikheev. Learning part-of-speech guessing rules from lexicon. In *Proceedings of COLING'96*, pages 770–775, 1996.

(Millen 00) David Millen. Community portals and collective goods: Conversation archives as an information resource. In *Proceedings of HICSS - 33rd Annual Hawaii International conference on Systems Sciences*, 2000.

(Miller *et al.* 90) G.A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller. Introduction to wordnet: An on-line lexical database. *International Journal of Lexicography (special issue)*, 3(4):235–312, 1990.

(Muresan *et al.* 01) Smaranda Muresan, Evelyne Tzoukerman, and Judith Klavans. Combining linguistic and machine learning techniques for email summarization. In *Proceedings of CoNLL Workshop at ACL-EACL 2001*, 2001.

(Newman 02a) Paula S. Newman. Email archive overviews using subject indexes. In *Proceedings of Conference on Human Factors and Computing Systems (CHI2002)*, pages 652–653, 2002.

(Newman 02b) Paula S. Newman. Exploring discussion lists: Steps and directions. In *Proceedings of Joint Conference of Digital Libraries*, 2002.

(Radev *et al.* 00) Dragomir R. Radev, Hongyan Jing, and Malgorzata Budzikowska. Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *ANLP/NAACL Workshop on Summarization*, 2000.

(Sack 00) Warren Sack. Conversation map: an interface for very-large-scale conversations. *Journal of Management Information Systems*, 17(3), 2000.

(Schiffman *et al.* 02) Barry Schiffman, Ani Nenkova, and Kathleen McKeown. Experiments in multi-document summarization. In *Proceedings of the Human Language Technology Conference (HLT2002)*, 2002.

(Smith & Fiore 01) Mark Smith and Andrew Fiore. Visualization components for persistent conversations. In *ACM SIG CHI*, 2001.