Summarizing Spoken and Written Conversations

Gabriel Murray and **Giuseppe Carenini**
Department of Computer Science
University of British Columbia
Vancouver, BC V6T 1Z4 Canada

Abstract

In this paper we describe research on summarizing conversations in the meetings and emails domains. We introduce a conversation summarization system that works in multiple domains utilizing general conversational features, and compare our results with domain-dependent systems for meeting and email data. We find that by treating meetings and emails as conversations with general conversational features in common, we can achieve competitive results with state-of-the-art systems that rely on more domain-specific features.

1 Introduction

Our lives are increasingly comprised of multimodal conversations with others. We email for business and personal purposes, attend meetings in person and remotely, chat online, and participate in blog or forum discussions. It is clear that automatic summarization can be of benefit in dealing with this overwhelming amount of interactional information. Automatic meeting abstracts would allow us to prepare for an upcoming meeting or review the decisions of a previous group. Email summaries would aid corporate memory and provide efficient indices into large mail folders.

When summarizing in each of these domains, there will be potentially useful domain-specific features – e.g. prosodic features for meeting speech, subject headers for emails – but there are also underlying similarites between these domains. They are all multiparty conversations, and we hypothesize that effective summarization techniques can be designed that would lead to robust summarization performance on a wide array of such conversation types. Such a general conversation summarization system would make it possible to summarize a wide variety of conversational data without needing to develop unique summarizers in each domain and across modalities. While progress has been made in summarizing conversations in individual domains, as described below, little or no work has been done on summarizing unrestricted, multimodal conversations.

In this research we take an extractive approach to summarization, presenting a novel set of conversational features for locating the most salient sentences in meeting speech and emails. We demonstrate that using these conversational features in a machine-learning sentence classification framework yields performance that is competitive or superior to more restricted domain-specific systems, while having the advantage of being portable across conversational modalities. The robust performance of the conversation-based system is attested via several summarization evaluation techniques, and we give an in-depth analysis of the effectiveness of the individual features and feature subclasses used.

2 Related Work on Meetings and Emails

In this section we give a brief overview of previous research on meeting summarization and email summarization, respectively.

2.1 Meeting Summarization

Among early work on meeting summarization, Waibel et al. (1998) implemented a modified version of the Maximal Marginal Relevance algorithm (Carbonell and Goldstein, 1998) applied to speech transcripts, presenting the user with the *n* best sentences in a meeting browser interface. Zechner (2002) investigated summarizing several genres of speech, including spontaneous meeting speech. Though relevance detection in his work relied largely on *tf.idf* scores, Zechner also explored cross-speaker information linking and question/answer detection.

More recently, researchers have investigated the utility of employing speech-specific features for summarization, including prosodic information. Murray et al. (2005a; 2005b) compared purely textual summarization approaches with feature-based approaches incorporating prosodic features, with human judges favoring the feature-based approaches. In subsequent work (2006; 2007), they began to look at additional speech-specific characteristics such as speaker status, discourse markers and high-level meta comments in meetings, i.e. comments that refer to the meeting itself. Galley (2006) used skip-chain Conditional Random Fields to model pragmatic dependencies between paired meeting utterances (e.g. QUESTION-ANSWER relations), and used a combination of lexical, prosodic, structural and discourse features to rank utterances by importance. Galley found that while the most useful single feature class was *lexical* features, a combination of acoustic, durational and structural features exhibited comparable performance according to Pyramid evaluation.

2.2 Email Summarization

Work on email summarization can be divided into summarization of individual email messages and summarization of entire email threads. Muresan et al. (2001) took the approach of summarizing individual email messages, first using linguistic techniques to extract noun phrases and then employing machine learning methods to label the extracted noun phrases as salient or not. Corston-Oliver et al. (2004) focused on identifying speech acts within a given email, with a particular interest in task-related sentences.

Rambow et al. (2004) addressed the challenge of summarizing entire threads by treating it as a binary sentence classification task. They considered three types of features: basic features that simply treat the email as text (e.g. *tf.idf*, which scores words highly if they are frequent in the document but rare across all documents), features that consider the thread to be a sequence of turns (e.g. the position of the turn in the thread), and email-specific features such as number of recipients and subject line similarity.

Carenini et al. (2007) took an approach to thread summarization using the Enron corpus (described below) wherein the thread is represented as a fragment quotation graph. A single node in the graph represents an *email fragment*, a portion of the email that behaves as a unit in a fine-grain representation of the conversation structure. A fragment sometimes consists of an entire email and sometimes a portion of an email. For example, if a given email has the structure

A
$>$ B
C

where B is a quoted section in the middle of the email, then there are three email fragments in total: two new fragments A and C separated by one quoted fragment B. Sentences in a fragment are weighted according to the Clue Word Score (CWS) measure, a lexical cohesion metric based on the recurrence of words in parent and child nodes. In subsequent work, Carenini et al. (2008) determined that subjectivity detection (i.e., whether the sentence contains sentiments or opinions from the author) gave additional improvement for email thread summaries.

Also on the Enron corpus, Zajic et al. (2008) compared Collective Message Summarization (CMS) to Individual Message Summarization (IMS) and found the former to be a more effective technique for summarizing email data. CMS essentially treats thread summarization as a multi-document summarization problem, while IMS summarizes individual emails in the thread and then concatenates them to form a thread summary.

In our work described below we also address the task of thread summarization as opposed to summarization of individual email messages, following Carenini et al. and the CMS approach of Zajic et al.

3 Experimental Setup

In this section we describe the classifier employed for our machine learning experiments, the corpora used, the relevant summarization annotations for each corpus, and the evaluation methods employed.

3.1 Statistical Classifier

Our approach to extractive summarization views sentence extraction as a classification problem. For all machine learning experiments, we utilize logistic regression classifiers. This choice was partly motivated by our earlier summarization research, where logistic regression classifiers were compared alongside support vector machines (SVMs) (Cortes and Vapnik, 1995). The two classifier types yielded very similar results, with logistic regression classifiers being much faster to train and thus expediting further development.

The *liblinear* toolkit [1] implements simple feature subset selection based on the F statistic (Chen and Lin, 2006) .

3.2 Corpora Description

For these experiments we utilize two corpora, the Enron corpus for email summarization and the AMI corpus for meeting summarization.

3.2.1 The Enron Email Corpus

The Enron email corpus[2] is a collection of emails released as part of the investigation into the Enron corporation (Klimt and Yang, 2004). It has become a popular corpus for NLP research (e.g. (Bekkerman et al., 2004; Yeh and Harnly, 2006; Chapanond et al., 2005; Diesner et al., 2005)) due to being realistic, naturally-occurring data from a corporate environment, and moreover because privacy concerns mean that there is very low availability for other publicly available email data.

39 threads have been annotated for extractive summarization, with five annotators assigned to each thread. The annotators were asked to select 30% of the sentences in a thread, subsequently labeling each selected sentence as either ’essential’ or ’optional.’ Essential sentences are weighted three times as highly as optional sentences. A sentence score, or GSValue, can therefore range between 0 and 15, with the maximum GSValue achieved when all five annotators consider the sentence essential, and a score of 0 achieved when no annotator selects the given sentence. For the purpose of training a binary classifier, we rank the sentences in each email thread according to their GSValues, then extract sentences until our summary reaches 30% of the total thread word count. We label these sentences as positive instances and the remainder as the negative class. Approximately 19% of sentences are labeled as positive, extractive examples.

Because the amount of labeled data available for the Enron email corpus is fairly small, for our classification experiments we employ a leave-one-out proceedure for the 39 email threads. The labeled data as a whole total just under 1400 sentences.

3.2.2 The AMI Meetings Corpus

For our meeting summarization experiments, we use the *scenario* portion of the AMI corpus (Carletta et al., 2005). The corpus consists of about 100 hours of recorded and annotated meetings. In the scenario meetings, groups of four participants take part in a series of four meetings and play roles within a fictitious company. While the scenario given to them is artificial, the speech and the actions are completely spontaneous and natural. There are 96 meetings in the training set, 24 in the development set, and 20 meetings for the test set.

For this corpus, annotators wrote abstract summaries of each meeting and extracted transcript dialogue act segments (DAs) that best conveyed or supported the information in the abstracts. A many-to-many mapping between transcript DAs and sentences from the human abstract was obtained for each annotator, with three annotators assigned to each meeting. It is possible for a DA to be extracted by an annotator but not linked to the abstract, but for training our binary classifiers, we simply consider a dialogue act to be a positive example if it is linked to a given human summary, and a negative example otherwise. This is done to maximize the likelihood that a data point labeled as “extractive” is truly an informative example for training purposes. Approximately 13% of the total DAs are ultimately labeled

[1] http://www.csie.ntu.edu.tw/˜cjlin/liblinear/

[2] http://www.cs.cmu.edu/˜enron/

as positive, extractive examples.

The AMI corpus contains automatic speech recognition (ASR) output in addition to manual meeting transcripts, and we report results on both transcript types. The ASR output was provided by the AMI-ASR team (Hain et al., 2007), and the word error rate for the AMI corpus is 38.9%.

3.3 Summarization Evaluation

For evaluating our extractive summaries, we implement existing evaluation schemes from previous research, with somewhat similar methods for meetings versus emails. These are described and compared below. We also evaluate our extractive classifiers more generally by plotting the receiver operator characteristic (ROC) curve and calculating the area under the curve (AUROC). This allows us to gauge the true-positive/false-positive ratio as the posterior threshold is varied.

We use the differing evaluation metrics for emails versus meetings for two primary reasons. First, the differing summarization annotations in the AMI and Enron corpora naturally lend themselves to slightly divergent metrics, one based on extract-abstract links and the other based on the essential/option/uninformative distinction. Second, and more importantly, using these two metrics allow us to compare our results with state-of-the-art results in the two fields of speech summarization and email summarization. In future work we plan to use a single evaluation metric.

3.3.1 Evaluating Meeting Summaries

To evaluate meeting summaries we use the weighted f-measure metric (Murray et al., 2006). This evaluation scheme relies on the multiple human annotated summary links described in Section 3.2.2. Both weighted precision and recall share the same numerator

$$num = \sum_{i=1}^{M} \sum_{j=1}^{N} L(s_i, a_j) \tag{1}$$

where $L(s_i, a_j)$ is the number of links for a DA s_i in the machine extractive summary according to annotator a_i, M is the number of DAs in the machine summary, and N is the number of annotators.

Weighted precision is defined as:

$$precision = \frac{num}{N \cdot M} \tag{2}$$

and weighted recall is given by

$$recall = \frac{num}{\sum_{i=1}^{O} \sum_{j=1}^{N} L(s_i, a_j)} \tag{3}$$

where O is the total number of DAs in the meeting, N is the number of annotators, and the denominator represents the total number of links made between DAs and abstract sentences by all annotators. The weighted f-measure is calculated as the harmonic mean of weighted precision and recall. The intuition behind weighted f-score is that DAs that are linked multiple times by multiple annotators are the most informative.

3.3.2 Evaluating Email Summaries

For evaluating email thread summaries, we follow Carenini et al. (2008) by implementing their *pyramid precision* scheme, inspired by Nenkova's pyramid scheme (2004). In Section 3.2.1 we introduced the idea of a GSValue for each sentence in an email thread, based on multiple human annotations. We can evaluate a summary of a given length by comparing its total GSValues to the maximum possible total for that summary length. For instance, if in a thread the three top scoring sentences had GSValues of 15, 12 and 12, and the sentences selected by a given automatic summarization method had GSValues of 15, 10 and 8, the pyramid precision would be 0.85.

Pyramid precision and weighted f-score are similar evaluation schemes in that they are both sentence based (as opposed to, for example, n-gram based) and that they score sentences based on multiple human annotations. Pyramid precision is very similar to equation 3 normalized by the maximum score for the summary length. For now we use these two slightly different schemes in order to maintain consistency with prior art in each domain.

4 A Conversation Summarization System

In our conversation summarization approach, we treat emails and meetings as conversations comprised of turns between multiple participants. We follow Carenini et al. (2007) in working at the finer granularity of email fragments, so that for an email thread, a turn consists of a single email fragment in the exchange. For meetings, a turn is a sequence of dialogue acts by one speaker, with the turn boundaries delimited by dialogue acts from other meeting participants. The features we derive for summarization are based on this view of the conversational structure.

We calculate two **length** features. For each sentence, we derive a word-count feature normalized by the longest sentence in the conversation (*SLEN*) and a word-count feature normalized by the longest sentence in the turn (*SLEN2*). Sentence length has previously been found to be an effective feature in speech and text summarization (e.g. (Maskey and Hirschberg, 2005; Murray et al., 2005a; Galley, 2006)).

There are several **structural** features used, including position of the sentence in the turn (*TLOC*) and position of the sentence in the conversation (*CLOC*). We also include the time from the beginning of the conversation to the current turn (*TPOS1*) and from the current turn to the end of the conversation (*TPOS2*). Conversations in both modalities can be well-structured, with introductory turns, general discussion, and ultimate resolution or closure, and sentence informativeness might significantly correlate with this structure. We calculate two pause-style features: the time between the following turn and the current turn (*SPAU*), and the time between the current turn and previous turn (*PPAU*), both normalized by the overall length of the conversation. These features are based on the email and meeting transcript timestamps. We hypothesize that pause features may be useful if informative turns tend to elicit a large number of responses in a short period of time, or if they tend to quickly follow a preceding turn, to give two examples.

There are two features related to the conversation **participants** directly. One measures how dominant the current participant is in terms of words in the conversation (*DOM*), and the other is a binary feature indicating whether the current participant initiated the conversation (*BEGAUTH*), based simply on whether they were the first contributor. It is hypothesized that informative sentences may more often belong to participants who lead the conversation or have a good deal of dominance in the discussion.

There are several **lexical** features used in these experiments. For each unique word, we calculate two conditional probabilities. For each conversation participant, we calculate the probability of the participant given the word, estimating the probability from the actual term counts, and take the maximum of these conditional probabilities as our first term score, which we will call *Sprob*.

$$Sprob(t) = \max_S p(S|t)$$

where t is the word and S is a participant. For example, if the word *budget* is used ten times in total, with seven uses by participant A, three uses by participant B and no uses by the other participants, then the *Sprob* score for this term is 0.70. The intuition is that certain words will tend to be associated with one conversation participant more than the others, owing to varying interests and expertise between the people involved.

Using the same procedure, we calculate a score called *Tprob* based on the probability of each turn given the word.

$$Tprob(t) = \max_T p(T|t)$$

The motivating factor for this metric is that certain words will tend to cluster into a small number of turns, owing to shifting topics within a conversation.

Having derived *Sprob* and *Tprob*, we then calculate several sentence-level features based on these term scores. Each sentence has features related to max, $mean$ and sum of the term scores for the words in that sentence (*MXS*, *MNS* and *SMS* for *Sprob*, and *MXT*, *MNT* and *SMT* for *Tprob*). Using a vector representation, we calculate the cosine between the conversation preceding the given sentence and the conversation subsequent to the sentence, first using *Sprob* as the vector weights (*COS1*) and then using *Tprob* as the vector weights (*COS2*). This is motivated by the hypothesis that informative sentences might change the conversation in some fashion, leading to a low cosine between the preceding and subsequent portions. We similarly calculate two scores measuring the cosine between the current sentence and the rest of the converation, using each term-weight metric as vector weights (*CENT1* for *Sprob* and *CENT2* for *Tprob*). This measures

Feature ID	Description
MXS	max *Sprob* score
MNS	mean *Sprob* score
SMS	sum of *Sprob* scores
MXT	max *Tprob* score
MNT	mean *Tprob* score
SMT	sum of *Tprob* scores
TLOC	position in turn
CLOC	position in conv.
SLEN	word count, globally normalized
SLEN2	word count, locally normalized
TPOS1	time from beg. of conv. to turn
TPOS2	time from turn to end of conv.
DOM	participant dominance in words
COS1	cos. of conv. splits, w/ *Sprob*
COS2	cos. of conv. splits, w/ *Tprob*
PENT	entro. of conv. up to sentence
SENT	entro. of conv. after the sentence
THISENT	entropy of current sentence
PPAU	time btwn. current and prior turn
SPAU	time btwn. current and next turn
BEGAUTH	is first participant (0/1)
CWS	rough ClueWordScore
CENT1	cos. of sentence & conv., w/ *Sprob*
CENT2	cos. of sentence & conv., w/ *Tprob*

Table 1: Features Key

whether the candidate sentence is generally similar to the conversation overall.

There are three word entropy features, calculated using the formula

$$went(s) = \frac{\sum_{i=1}^{N} p(x_i) \cdot - \log(p(x_i))}{(\frac{1}{N} \cdot - \log(\frac{1}{N})) \cdot M}$$

where s is a string of words, x_i is a word type in that string, $p(x_i)$ is the probability of the word based on its normalized frequency in the string, N is the number of word types in the string, and M is the number of word tokens in the string.

Note that word entropy essentially captures information about type-token ratios. For example, if each word token in the string was a unique type then the word entropy score would be 1. We calculate the word entropy of the current sentence (*THISENT*), as well as the word entropy for the conversation up until the current sentence (*PENT*) and the word entropy for the conversation subsequent to the current sentence (*SENT*). We hypothesize that informative sentences themselves may have a diversity of word types, and that if they represent turning points in the conversation they may affect the entropy of the subsequent conversation.

Finally, we include a feature that is a rough approximation of the ClueWordScore (CWS) used by Carenini et al. (2007). For each sentence we remove stopwords and count the number of words that occur in other turns besides the current turn. The CWS is therefore a measure of conversation cohesion.

For ease of reference, we hereafter refer to this conversation features system as ConverSumm.

5 Comparison Summarization Systems

In order to compare the ConverSumm system with state-of-the-art systems for meeting and email summarization, respectively, we also present results using the features described by Murray and Renals (2008) for meetings and the features described by Rambow (2004) for email. Because the work by Murray and Renals used the same dataset, we can compare our scores directly. However, Rambow carried out summarization work on a different, unavailable email corpus, and so we re-implemented their summarization system for our current email data.

In their work on meeting summarization, Murray and Renals creating 700-word summaries of each meeting using several classes of features: prosodic, lexical, structural and speaker-related. While there are two features overlapping between our systems (word-count and speaker/participant dominance), their system is primarily domain-dependent in its use of prosodic features while our features represent a more general conversational view.

Rambow presented 14 features for the summarization task, including email-specific information such as the number of recipients, number of responses, and subject line overlap. There is again a slight overlap in features between our two systems, as we both include length and position of the sentence in the thread/conversation.

6 Results

Here we present, in turn, the summarization results for meeting and email data.

6.1 Meeting Summarization Results

Figure 1 shows the F statistics for each Conversumm feature in the meeting data, providing a measure of the usefulness of each feature in discriminating between the positive and negative classes. Some

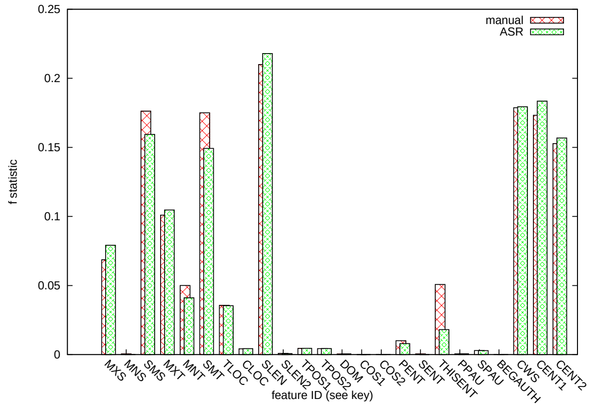


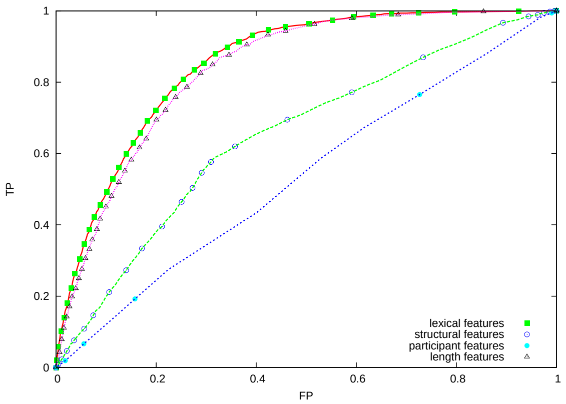
Figure 1: Feature F statistics for AMI meeting corpus

System	Weighted F-Score	AUROC
Speech - Man	0.23	0.855
Speech - ASR	0.24	0.850
Conv. - Man	0.23	0.852
Conv. - ASR	0.22	0.853

Table 2: Weighted F-Scores and AUROCs for Meeting Summaries

features such as participant dominance have very low F statistics because each sentence by a given participant will receive the same score; so while the feature itself may have a low score because it does not discriminate informative versus non-informative sentences on its own, it may well be useful in conjunction with the other features. The best individual ConverSumm features for meeting summarization are sentence length (SLEN), sum of $Sprob$ scores, sum of $Tprob$ scores, the simplified CWS score (CWS), and the two centroid measures (CENT1 and CENT2). The word entropy of the candidate sentence is very effective for manual transcripts but much less effective on ASR output. This is due to the fact that ASR errors can incorrectly lead to high entropy scores.

Table 2 provides the weighted f-scores for all summaries of the meeting data, as well as AUROC scores for the classifiers themselves. For our 700-word summaries, the Conversumm approach scores comparably to the speech-specific approach on both manual and ASR transcripts according to weighted f-score. There are no significant differences according to paired t-test. For the AUROC measures, there are again no significant differences between the conversation summarizers and speech-specific summarizers. The AUROC for the conversation system is slightly lower on manual transcripts and slightly higher when applied to ASR output.



Fea. Subset	AUROC
Structural	0.652
Participant	0.535
Length	0.837
Lexical	0.852

Figure 2: AUROC Values for Feature Subclasses, AMI Corpus

For all systems the weighted f-scores are somewhat low. This is partly owing to the fact that output summaries are very short, leading to high precision and low recall. The low f-scores are also indicative of the difficulty of the task. Human performance, gauged by comparing each annotator's summaries to the remaining annotators' summaries, exhibits an average weighted f-score of 0.47 on the same test set. The average kappa value on the test set is 0.48, showing the relatively low inter-annotator agreement that is typical of summarization annotation. There is no additional benefit to combining the conversational and speech-specific features. In that case, the weighted f-scores are 0.23 for both manual and ASR transcripts. The overall AUROC is 0.85 for manual transcripts and 0.86 for ASR.

We can expand the features analysis by considering the effectiveness of certain subclasses of features. Specifically, we group the summarization features into *lexical*, *structural*, *participant* and *length* features. Figure 2 shows the AUROCs for the feature subset classifiers, illustrating that the lexical subclass is very effective while the length features also constitute a challenging baseline. A weakness

System	Pyramid Precision	AUROC
Rambow	0.50	0.64
Conv.	0.46	0.75

Table 3: Pyramid Precision and AUROCs for Email Summaries

of systems that depend heavily on length features, however, is that recall scores tend to decrease because the extracted units are much longer - weighted recall scores for the 700 word summaries are significantly worse according to paired t-test (p<0.05) when using just length features compared to the full feature set.

6.2 Email Summarization Results

Figure 3 shows the F statistic for each ConverSumm feature in the email data.The two most useful features are sentence length and CWS. The *Sprob* and *Tprob* features rate very well according to the F statistic. The two centroid features incorporating *Sprob* and *Tprob* are comparable to one another and are very effective features as well.

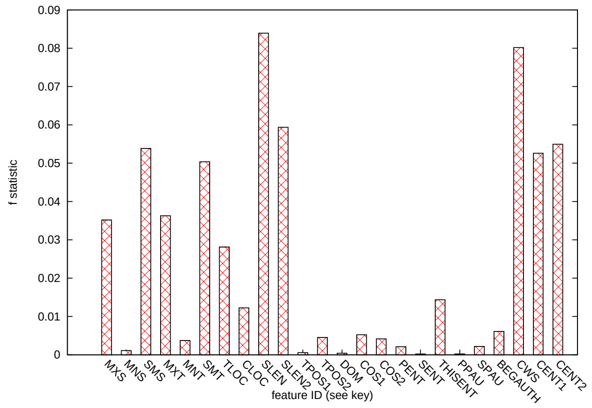
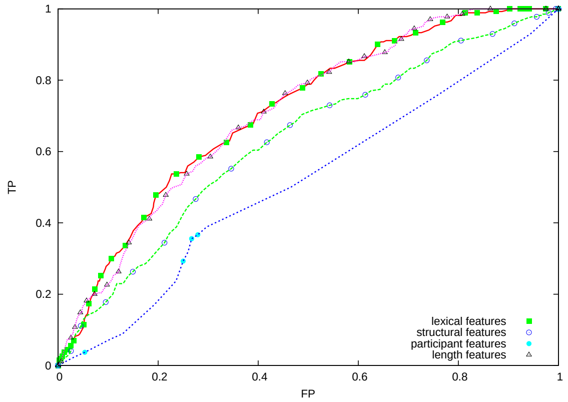


Figure 3: Feature F statistics for Enron email corpus

After creating 30% word compression summaries using both the ConverSumm and Rambow approaches, we score the 39 thread summaries using Pyramid Precision. The results are given in Table 3. On average, the Rambow system is slightly higher with a score of 0.50 compared with 0.46 for the conversational system, but there is no statistical difference according to paired t-test.

The average AUROC for the Rambow system is 0.64 compared with 0.75 for the ConverSumm system, with ConverSumm system significantly better according to paired t-test (p<0.05). Random classification performance would yield an AUROC of 0.5.



Fea. Subset	AUROC
Structural	0.63
Participant	0.51
Length	0.71
Lexical	0.71

Figure 4: AUROC Values for Feature Subclasses, Enron Corpus

Combining the Rambow and ConverSumm features does not yield any overall improvement. The Pyramid Precision score in that case is 0.47 while the AUROC is 0.74.

Figure 4 illustrates that the lexical and length features are the most effective feature subclasses, though the best results overall are derived from a combination of all feature classes.

7 Discussion

According to multiple evaluations, the ConverSumm features yield competitive summarization performance with the comparison systems. There is a clear set of features that are similarly effective in both domains, especially CWS, the centroid features, the $Sprob$ features, the $Tprob$ features, and sentence length. There are other features that are more effective in one domain than the other. For example, the BEGAUTH feature, indicating whether the current participant began the conversation, is more useful for emails. It seems that being the first person to speak in a meeting is not as significant as being the first person to email in a given thread. SLEN2, which normalizes sentence length by the longest sentence in the turn, also is much more effective for emails. The reason is that many meeting turns consist of a single, brief utterance such as "Okay, yeah."

The finding that the summary evaluations are not significantly worse on noisy ASR compared with manual transcripts has been previously attested (Valenza et al., 1999; Murray et al., 2005a), and it is encouraging that our ConverSumm features are similarly robust to this noisy data.

8 Conclusion

We have shown that a general conversation summarization approach can achieve results on par with state-of-the-art systems that rely on features specific to more focused domains. We have introduced a conversation feature set that is similarly effective in both the meetings and emails domains. The use of multiple summarization evaluation techniques confirms that the system is robust, even when applied to the noisy ASR output in the meetings domain. Such a general conversation summarization system is valuable in that it may save time and effort required to implement unique systems in a variety of conversational domains.

We are currently working on extending our system to other conversation domains such as chats, blogs and telephone speech. We are also investigating domain adaptation techniques; for example, we hypothesize that the relatively well-resourced domain of meetings can be leveraged to improve email results, and preliminary findings are encouraging.

References

R. Bekkerman, A. McCallum, and G. Huang. 2004. Automatic categorization of email into folders: Benchmark experiments on Enron and SRI corpora. Technical Report IR-418, Center of Intelligent Information Retrieval, UMass Amherst.

J. Carbonell and J. Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proc. of ACM SIGIR Conference on Research and Development in Information Retrieval 1998, Melbourne, Australia*, pages 335–336.

G. Carenini, R. Ng, and X. Zhou. 2007. Summarizing email conversations with clue words. In *Proc. of ACM WWW 07, Banff, Canada*.

G. Carenini, X. Zhou, and R. Ng. 2008. Summarizing emails with conversational cohesion and subjectivity. In *Proc. of ACL 2008, Columbus, Ohio, USA*.

J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, I. McCowan, W. Post, D. Reidsma, and P. Wellner. 2005. The AMI meeting corpus: A pre-announcement. In *Proc. of MLMI 2005, Edinburgh, UK*, pages 28–39.

A. Chapanond, M. Krishnamoorthy, and B. Yener. 2005. Graph theoretic and spectral analysis of enron email data. *Comput. Math. Organ. Theory*, 11(3):265–281.

Y-W. Chen and C-J. Lin. 2006. Combining SVMs with various feature selection strategies. In I. Guyon, S. Gunn, M. Nikravesh, and L. Zadeh, editors, *Feature extraction, foundations and applications*. Springer.

S. Corston-Oliver, E. Ringger, M. Gamon, and R. Campbell. 2004. Integration of email and task lists. In *Proc. of CEAS 2004, Mountain View, CA, USA*.

C. Cortes and V. Vapnik. 1995. Support-vector networks. *Machine Learning*, 20(3):273–297.

J. Diesner, T. Frantz, and K. Carley. 2005. Communication networks from the enron email corpus "it's always about the people. enron is no different". *Comput. Math. Organ. Theory*, 11(3):201–228.

M. Galley. 2006. A skip-chain conditional random field for ranking meeting utterances by importance. In *Proc. of EMNLP 2006, Sydney, Australia*, pages 364–372.

T. Hain, L. Burget, J. Dines, G. Garau, V. Wan, M. Karafiat, J. Vepa, and M. Lincoln. 2007. The AMI system for transcription of speech in meetings. In *Proc. of ICASSP 2007,*, pages 357–360.

B. Klimt and Y. Yang. 2004. Introducing the enron corpus. In *Proc. of CEAS 2004, Mountain View, CA, USA*.

S. Maskey and J. Hirschberg. 2005. Comparing lexial, acoustic/prosodic, discourse and structural features for speech summarization. In *Proc. of Interspeech 2005, Lisbon, Portugal*, pages 621–624.

S. Muresan, E. Tzoukermann, and J. Klavans. 2001. Combining linguistic and machine learning techniques for email summarization. In *Proc. of ConLL 2001, Toulouse, France*.

G. Murray and S. Renals. 2008. Meta comments for summarizing meeting speech. In *Proc. of MLMI 2008, Utrecht, Netherlands*.

G. Murray, S. Renals, and J. Carletta. 2005a. Extractive summarization of meeting recordings. In *Proc. of Interspeech 2005, Lisbon, Portugal*, pages 593–596.

G. Murray, S. Renals, J. Carletta, and J. Moore. 2005b. Evaluating automatic summaries of meeting recordings. In *Proc. of the ACL 2005 MTSE Workshop, Ann Arbor, MI, USA*, pages 33–40.

G. Murray, S. Renals, J. Moore, and J. Carletta. 2006. Incorporating speaker and discourse features into speech summarization. In *Proc. of the HLT-NAACL 2006, New York City, USA*, pages 367–374.

G. Murray. 2007. *Using Speech-Specific Features for Automatic Speech Summarization*. Ph.D. thesis, University of Edinburgh.

A. Nenkova and B. Passonneau. 2004. Evaluating content selection in summarization: The Pyramid method. In *Proc. of HLT-NAACL 2004, Boston, MA, USA*, pages 145–152.

O. Rambow, L. Shrestha, J. Chen, and C. Lauridsen. 2004. Summarizing email threads. In *Proc. of HLT-NAACL 2004, Boston, USA*.

R. Valenza, T. Robinson, M. Hickey, and R. Tucker. 1999. Summarization of spoken audio through information extraction. In *Proc. of the ESCA Workshop on Accessing Information in Spoken Audio, Cambridge UK*, pages 111–116.

A. Waibel, M. Bett, M. Finke, and R. Stiefelhagen. 1998. Meeting browser: Tracking and summarizing meetings. In D. E. M. Penrose, editor, *Proc. of the Broadcast News Transcription and Understanding Workshop, Lansdowne, VA, USA*, pages 281–286.

J. Yeh and A. Harnly. 2006. Email thread reassembly using similarity matching. In *Proc of CEAS 2006*.

D. Zajic, B. Dorr, and J. Lin. 2008. Single-document and multi-document summarization techniques for email threads using sentence compression. *Information Processing and Management, to appear*.

K. Zechner. 2002. Automatic summarization of open-domain multiparty dialogues in diverse genres. *Computational Linguistics*, 28(4):447–485.