Extractive Meeting Summarization through speaker zone detection

Mohammad Hadi Bokaei[1], *Hossein Sameti*[1], *Yang Liu*[2]

[1]Department of Computer Engineering, Sharif University of Technology, Tehran, Iran
[2]Department of Computer Science, The University of Texas at Dallas, U.S.A

bokaei@ce.sharif.edu, sameti@sharif.edu, yangl@hlt.utdallas.edu

Abstract

In this paper we investigate the role of discourse analysis in extractive meeting summarization task. Specifically our proposed method comprises of two distinct steps. First we use a meeting segmentation algorithm in order to detect various functional parts of the input meeting. Afterwards, a two level scoring mechanism in a graph-based framework is used to score each dialogue act in order to extract the most valuable ones and include them in the extracted summary. We evaluate our proposed method on AMI and ICSI corpora and compare it with other state-of-the-art graph based algorithms according to various evaluation metrics. The experimental results show that our algorithm outperforms the other state-of-the-art ones according to most of the metrics and on both datasets.

1. Introduction

Improvements of automatic speech recognition systems and increasing amount of audio data (such as broadcast news, voice mail, telephony conversations and meetings) have attracted plenty of research interest in the field of speech summarization. On the other hand, conversation and in its specific form, meetings are an integral part of each organization. Not surprisingly then, there is growing interest in developing automatic methods which summarize meetings in a way that by reading only the extracted summary, the reader will be informed about the important key-notes of the meeting.

Traditionally and based on earlier works on summarization task [1], extractive methods are expressed as a combination of two distinct goals in selecting each sentence: maximizing the information covered by the sentence while minimizing its redundancy according to previously extracted sentences. Different summarization algorithms use different strategies to calculate how much informative a sentence is, but few of them incorporate meeting specific structural information in calculation [2, 3, 4, 5].

In this work, we propose to consider discourse by detecting speaker zones in the meeting flow. We try to segment a meeting transcript into functionally coherent parts such as monologue or discussion. We hypothesize that detecting these parts and summarizing meeting transcript accordingly can improve the accuracy of the extracted output. We compare our proposed algorithm against state-of-the-art algorithms and show improvement over them using standard summarization measures (ROUGE-1, ROUGE-2) and classification measures (Precision, Recall, F-Measure) on AMI and ICSI standard meeting corpora.

This paper is organized as follows. Section 2 describes relevant works. Section 3 illustrates our proposed algorithm. In Section 4 evaluation metrics and experimental results are shown. Finally in Section 5 we describe our future work and conclude the usefulness of our proposed algorithm.

2. Related works

Generally speaking, there are two distinct categories of summarization approaches: supervised methods which need annotated training data to train their models [2, 6, 7, 8, 9] and unsupervised methods which need no labeled training set and use only the information in words of the document to be summarized [10, 11, 12, 13].

Most of the approaches applied for meeting summarization purpose are inherited from text summarization task like Maximal Marginal Relevance (MMR) [1, 14], topic-based [15, 16, 17], graph-based [10, 13, 18], and optimization-based [12, 19, 20]. It has been shown that the graph-based approaches are the most successful ones compared to other methods in the meeting summarization domain [21].

One successful method proposed in graph-based category is ClusterRank [13]. In this method, adjacent utterances are clustered according to their similarity values. A graph is constructed where nodes are these clusters and Random-walk procedure is applied on this graph to score each cluster. Afterwards, each utterance is scored according to its associated cluster and the similarity between the utterance and its corresponding cluster. The main difference between our work and ClusterRank is the segmentation step. While ClusterRank uses a very simple algorithm which clusters similar adjacent utterances to one segment, our segmentation algorithm tries to segment the meeting into zones in which speakers distribution is steady. We also use weighted combination schema to compute the final score for each utterance.

Another state-of-the-art algorithm proposed in graph-based framework constructs two-layer graph [10]. This algorithm uses speakers information in order to score each utterance. Utterances are represented as nodes in utterance-layer and speakers are represented as nodes in speaker-layer of the graph. Applying Random-walk procedure on this constructed graph, scores from different layers are reinforced so that final utterance scores are influenced by utterances from the same speaker and similar utterances.

There are a few previous works which incorporate discourse information in the summarization task. [2] adds some very simple discourse features (such as existence of specific keywords in the utterance and the position of that utterance in the meeting) to calculate informative score for each sentence. In [3] it is shown that adding structural features can improve the effectiveness of the summarization algorithms. [5] studies the effect of turn-taking and participant involvement on the task of finding more informative segments in a meeting. The authors in [4] study the usefulness of discourse more profoundly. They use Conditional Random Fields to extract rhetorical structure and summary in a single step. However the main difference between this work and ours is that this work is supervised in the

Copyright © 2015 ISCA

sense that it needs an annotated training data to learn the parameters of CRF model. The main goal of our work is to design an unsupervised algorithm which improves the accuracy of state-of-the-art algorithms. Another difference between these works is that in [4], the authors use different categories to segment the meeting accordingly. We aim to segment a meeting into functional coherent parts which discriminates between monologue part from discussion one.

We try to segment a meeting into separate parts, where in each a different set of speakers is engaged. There are some previous works that pursue the same goal [22, 23, 24]. These works try to segment a meeting transcript according to these categories: Monologue, Discussion, Note-taking, Presentation, and Presentation with white-board. However our segmentation algorithm differs from these previous ones according to the following aspects:

1. Our segmentation algorithm is unsupervised[1], while all the previous approaches are supervised.
2. Our goal is to discriminate between monologue and various kinds of discussion parts. In previous approaches there are other unrelated types of segments (like presentation, note-taking, etc.). Discriminating these additional segments has no effect on the summarization task. However previous approaches don't discriminate between various kinds of discussion parts. In natural meetings there may be a large discussion segment constructed from smaller parts where different speakers are engaged in. Discriminating these parts can improve the performance of summarization algorithm.

3. Our proposed approach

Our proposed algorithm consists of two separate steps. The first one segments the input document into functional coherent parts. In this step, we aim to segment and structure the meeting transcript into sequences of meeting actions such as monologue or discussion between two or more speakers. The second step uses this segmentation and scores each utterance in the meeting according to its associated segment using a graph-based approach. In the following we describe these two steps separately. Here we assume that the utterance boundaries and the speaker of each utterance is known in advance.

3.1. Functional segmentation step

Analyzing multiple meetings in both AMI [25] and ICSI [26] corpora, we notice that a meeting can be segmented according to the distribution of speakers. This is the main idea behind our proposed algorithm. In fact we try to detect various parts of a meeting where speakers distribution changes. An example is shown in Figure 1. In this figure, two segments are shown. In the left one,there is a key speaker who dominates the segment and lectures all other participants about an aspect of the discussed topic. In this segment, speakers distribution has a peak over that dominant speaker. In the other one there is a debate between participants resulting in a flat distribution where all or some of speakers have the same contribution. We try to find a segmentation in which each segment differs from adjacent ones according to their speakers' distribution. To achieve this goal, we first segment a meeting into smaller equal segments. Using this initial segmentation, we iterate through segments and merge ones that have similar speakers' distribution. We stop the repetition when we reach a segmentation which is not changed anymore. The complete algorithm is illustrated in Algorithm 1. Applying this algorithm, we expect to reach a segmentation where each segment differs from next and previous ones according to their speakers distribution.

[1]Our algorithm needs a very small development set (one meeting in our study) to tune its parameters. However its performance is not very sensitive on the changes of these parameters.

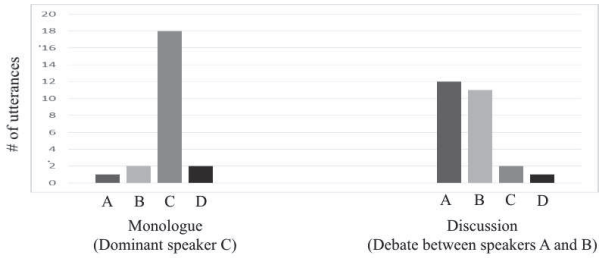


Figure 1: Example of speakers' distribution for two kinds of segments, monologue and discussion. In this figure, each column represents the number of utterances spoken by that speaker in the segment.

Algorithm 1 functional segmentation algorithm

Input: seq (Sequence of speaker id's of a meeting)
Parameters: T, TH_{cu}
$seq \leftarrow$ Segmentation of seq into T length parts.
while No changes occured in seq **do**
  $num_{seg} \leftarrow$ Number of segments in seg
  for all segment s in seg **do**
    $dist_s \leftarrow$ speakers distribution for s
    Merge s with its consecutive segments until distance between the merged segment and s reach TH_{cu}
  end for
end while
Output: seg

Generally speaking, the idea here is similar to bottom-up approaches for clustering. We start with over clustering in which the number of clusters is more than the final number of desired clusters. We then iteratively combine adjacent clusters with similar speaker distributions. The performance of this proposed algorithm is shown later in experimental results.

3.2. Utterance ranking step

Segmenting utterances in a meeting, now we need a procedure to compute a salience score for each utterance. Similar to ClusterRank algorithm introduced in [13], we use a two level approach for scoring each utterance. At the first step, a graph G is constructed where each segment is represented as a node and directed edges between nodes are weighted according to the segments similarity value. We compute similarity between clusters using Cosine measure as shown in Eqn. 1.

$$Cosine(X,Y) = \frac{\sum_{w \in X,Y} tfidf(X,w) * tfidf(Y,w)}{\sqrt{\sum_{x_1 \in X} tfidf(X,x_1)^2 * \sum_{y_1 \in Y} tfidf(Y,y_1)^2}} \quad (1)$$

We use tfidf to compute score of word w in segment X. This score is the product of two terms. The first one is the number of occurrences of w in X (tf) and the second one is the inverse of number of all segments that contains w (idf). We

normalize the scores of outgoing edges of a node, so the graph can be seen as a Markov chain. We apply random walk process on G to compute salience score for each segment. Specifically Eqn. 2 is used iteratively to compute segments scores.

$$P(u) = \frac{d}{N} + (1-d) \sum_{v \in adj(u)} weight_{edge}(u,v) * P(v) \quad (2)$$

In Eqn. 2, P is a vector whose elements are salience scores of each segment. d is damping factor which is typically chosen in the interval [0.1, 0.2] and ensures convergence [18]. u and v are two nodes of G and $adj(u)$ denotes nodes that are adjacent to u. Vector P can be chosen randomly at first and Eqn. 2 is then iteratively applied on all nodes of G until changes on vector P becomes lower than a predefined threshold.

At this point, scores for each segment is computed. Now utterance (U) in the meeting must be scored according to its associated segment (S) score and the degree of importance of U in S. As previous works, the degree of importance of U in S can be computed according to their similarity. Both U and S are considered as bag of words and cosine similarity measure is used to compute the similarity. These two scores are combined in a weighted manner to compute the final score for each utterance according to Eqn. 3.

$$score(U) = cosine(U,S) + \omega * P(S) \quad (3)$$

In the above equation, P is computed according to Eqn. 2. ω and other parameters of the algorithm are tuned according to a small development set as shown in the experimental result. Using the procedure described above, we obtain the salience score for each utterance. Now all utterances are sorted according to their scores and the best ones are greedily selected until summary length exceeds predefined threshold.

4. Experimental results

4.1. Setup

The AMI meeting corpus [25] is a collection of 100 hours of meeting data that includes annotations in various layers such as speech audio, transcripts, focus of attention, and etc. In order to evaluate the effectiveness of our proposed algorithm, a subset of 11 meetings in the AMI corpus were manually annotated and used as test set[2]. We employed graduate students as annotators and asked them to segment the meetings according to different events. They were given a guideline which included the task definition and various examples used to clarify the concept of events and function segmentation of a meeting. Each meeting was annotated by one annotator. One special trained annotator then checked and finalized the annotations. The average number of segments in this set is 19.6. One of these meetings was used as our development set on which we tuned the parameter TH_{cu} of the segmentation algorithm.

We used P_k [27], which is the most widely used metric for segmentation evaluation. Given two points in a sequence, P_k specifies the probability of segmentation error, which is the average probability that the segmenter's decision is incorrect. Note that P_k is a measure of error and thus a lower score means better segmentation performance.

To evaluate the whole summarization algorithm, we used 20 meetings in the same AMI corpus[3]. Each meeting is prepared with one reference summary. These summaries have no unique compression ratio. The average compression ratio for our test set is 0.38 and its variance is 0.0091.

We also tested our summarization algorithm on ICSI meeting corpus [26] which contains 75 recordings from natural meetings. Each meeting is about an hour long. We evaluated the results of our summarization algorithm according to the whole meetings in this corpus. For our evaluation purpose we used human transcription of the meeting and also assumed that the speaker of each utterance is specified in advance.

According to the fact that extractive meeting summarization is indeed a classification task where important utterances must be distinguished from not-important ones, we can use common evaluation metrics in classification tasks such as precision, recall and F-measure. From another viewpoint, ROUGE [28] evaluates a summarization system based on the number of overlapping units such as n-grams, between the system generated summary and the ideal summary created by human annotator. We show results of our summarization algorithm using ROUGE-1 (unigram overlap) and ROUGE-2 (bigram overlap) along with classification measures to compare the result with the other state-of-the-art algorithms[4].

4.2. Results

Using the single meeting in our development set, we tune the parameters of our algorithm. Specifically we choose $TH_{cu} = 0.2$ and $T = 10$. We use $\epsilon = 10^{-4}$ for the threshold used in utterance ranking step. We also test various values for ω and found that the best result is achieved on $\omega = 0.25$.

First we analyze the performance of our proposed segmentation algorithm. The result of applying this algorithm on our test set is shown in Table 2. In order to further analyze the algorithm, the segmentation found in a sample meeting is shown in Figure 2.a. This figure shows the boundaries found by our proposed algorithm (dashed lines) and the reference boundaries (solid lines). Figure 2.b shows the speaker distribution for the first 6 segments in the segmentation found for this meeting. As this figure shows, we have some segments in which one person lectures all others and thus it will be a monologue segment (segments 1, 4 and 6). We also have segments which contain discussions between specific participants in the meeting (segments 2, 3 and 5).

We also compare our proposed summarization algorithm against state-of-the-art algorithms proposed in literature. Specifically these algorithms are:

- ClusterRank [13]: This is our base algorithm. As stated, our utterance ranking step is inspired by this algorithm.
- MRRW-WBP [10]: This is the state-of-the-art algorithm, recently proposed. It uses Probabilistic Latent Semantic Indexing in a graphical framework in order to score each utterance.

To compare results fairly, we consider different compression ratios for meetings. For each meeting we compute the compression ratio according to the one used to generate reference extractive summary in the corpus, which is the number of words in the reference extracted summary divided by the total number of the

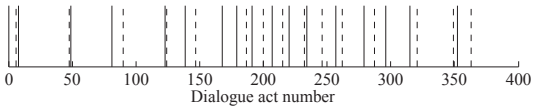
[2]The ids of annotated meetings are: *es2008a*, *is1000a*, *is1001a*, *is1001b*, *is1001c*, *is1003b*, *is1006b*, *is1008a*, *is1008b*, *is1008c* and *ts3005a*. We chose these meetings since they have more annotations in the AMI corpus, which can be useful for our future studies. The reference segmentation as well as the annotation guide can be found here: http://ce.sharif.edu/ bokaei/resources/funseg/

[3]The ids of meetings are: *ES2004*, *ES2014*, *IS1009*, *TS3003* and *TS3007*.

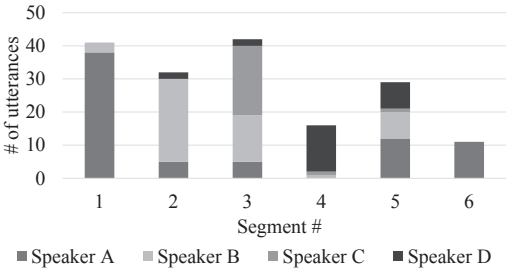
[4]We uses ROUGE package (http://www.berouge.com/) to evaluate our proposed summarization algorithm.

Table 1: Results of our proposed algorithm compared to ClusterRank [13] and MRRW-WBP [10]

Data	Algorithm	Classification measure (%)			ROUGE1 (%)			ROUGE2 (%)		
		P	R	F	P	R	F	P	R	F
AMI	**ClusterRank**	39.07	**42.26**	39.74	78.31	74.06	74.87	54.73	50.92	51.90
	MRRW-WBP	**54.43**	37.53	**43.67**	80.03	74.97	76.14	61.15	56.21	57.63
	Proposed Method	53.46	37.87	43.58	**80.51**	**76.22**	**77.02**	**61.75**	**57.61**	**58.65**
ICSI	**ClusterRank**	18.70	16.21	17.07	60.47	70.99	64.94	31.81	37.07	34.04
	MRRW-WBP	25.67	17.97	20.87	63.64	74.93	68.36	35.36	41.48	37.91
	Proposed Method	**26.36**	**23.36**	**24.48**	**64.45**	**75.76**	**69.18**	**37.03**	**43.31**	**39.66**



(a)



(b)

Figure 2: Results of applying the segmentation algorithm on the sample meeting *es2008a*: (a) Location of found boundaries (dashed line) in comparison with the reference boundaries (solid line). (b) Speakers' distribution for the first six segments after applying the proposed functional segmentation algorithm. Each column represents total number of utterances in that segment differed in color according to their speakers..

utterances in that meeting. Results are shown in Table 1. Comparing the results of our proposed algorithm and ClusterRank, we see that using segmentation based on discourse analysis, improves the performance of the whole summarization algorithm.

The major advantage of our proposed algorithm over other base-line methods is its capability of detecting local important utterances. The importance of these utterances are due to the context they are used in. The similarity between these utterances and the whole document is not considerable. Algorithms that calculate the score of each utterance according to its similarity to the whole document can not detect them. On the other hand, our proposed algorithm detects them by segmenting the transcript and then calculating the score of each utterance according to its similarity to the corresponding segment. However, the calculated score for a segment containing these types of utterances is lower than a segment containing utterances which are more similar to the whole document. This is the main reason that the best result is achieved when the weight of cluster score is decreased to 0.25 in order to lower the effect of cluster score in comparison with utterance similarity score.

Table 2: Results of our proposed segmentation algorithm

Algorithm	P_k
Random segmentation	0.50
Proposed segmentation algorithm	0.43

5. Conclusion and future works

In this work we investigated the effect of considering discourse structure in the performance of summarization systems. Specifically we tried to segment meeting discourse into functionally coherent segments and then scored each utterance according to this segmentation. Results showed improvement over other state-of-the-art algorithms.

We believe that the performance of this algorithm can be further improved using other levels of discourse analysis such as extracting relations between utterances. From another viewpoint, literature proves that using other weighting schema other than tfidf, which we used in this study, can improve the accuracy of summarization system in multi-party conversations [29].

However, the main track of our future works will concentrate on designing a system which summarizes each segment separately. According to this fact that demands of users change according to the type of the segments. While in a monologue segment, readers are interested in finding the key notes of the lecture, in a discussion segment, they are interested in finding the topic of the discussion, its outcome and maybe sentiments of participants to that result. According to these needs, the way a summary for monologue segment is generated must be different from the way a summary is generated for discussion segment.

6. References

[1] J. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1998, pp. 335–336.

[2] G. Murray, S. Renals, J. Carletta, and J. Moore, "Incorporating speaker and discourse features into speech summarization," in *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*. Association for Computational Linguistics, 2006, pp. 367–374.

[3] S. Maskey and J. Hirschberg, "Comparing lexical, acoustic/prosodic, structural and discourse features for speech summarization." in *INTERSPEECH*, 2005, pp. 621–624.

[4] J. J. Zhang and P. Fung, "Automatic parliamentary meeting minute generation using rhetorical structure modeling," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 9, pp. 2492–2504, 2012.

[5] C. Lai, J. Carletta, S. Renals, K. Evanini, and K. Zechner, "Detecting summarization hot spots in meetings using group level involvement and turn-taking features." in *INTERSPEECH*, 2013, pp. 2723–2727.

[6] M. Galley, "A skip-chain conditional random field for ranking meeting utterances by importance," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2006, pp. 364–372.

[7] A. H. Buist, W. Kraaij, and S. Raaijmakers, "Automatic summarization of meeting data: A feasibility study," in *Proceedings of the 15th CLIN conference*, 2004.

[8] S. Maskey and J. Hirschberg, "Automatic summarization of broadcast news using structural features," in *Proceedings of the Conference of the International Speech Communication Association (INTERSPEECH)*, 2003.

[9] S. Xie and Y. Liu, "Improving supervised learning for meeting summarization using sampling and regression," *Computer Speech & Language*, vol. 24, no. 3, pp. 495–514, 2010.

[10] Y.-N. Chen and F. Metze, "Two-layer mutually reinforced random walk for improved multi-party meeting summarization," in *Spoken Language Technology Workshop (SLT)*. IEEE, 2012, pp. 461–466.

[11] L. Wang and C. Cardie, "Focused meeting summarization via unsupervised relation extraction," in *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Association for Computational Linguistics, 2012, pp. 304–313.

[12] D. Gillick, K. Riedhammer, B. Favre, and D. Hakkani-Tur, "A global optimization framework for meeting summarization," in *Proceedings of the International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*. IEEE, 2009, pp. 4769–4772.

[13] N. Garg, B. Favre, K. Riedhammer, and D. Hakkani-Tür, "Clusterrank: a graph based method for meeting summarization." 2009, pp. 1499–1502.

[14] K. Zechner, "Automatic summarization of open-domain multiparty dialogues in diverse genres," *Computational Linguistics*, vol. 28, no. 4, pp. 447–485, 2002.

[15] G. Murray, S. Renals, J. Carletta, and J. Moore, "Evaluating automatic summaries of meeting recordings," in *Proceedings of the ACL MTSE Workshop, Ann Arbor, MI, USA*, 2005, pp. 33–40.

[16] Y. Gong and X. Liu, "Generic text summarization using relevance measure and latent semantic analysis," in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2001, pp. 19–25.

[17] S.-Y. Kong and L.-s. Lee, "Improved spoken document summarization using probabilistic latent semantic analysis (plsa)," in *Proceedings of the International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, vol. 1. IEEE, 2006, pp. 941–944.

[18] G. Erkan and D. R. Radev, "Lexrank: Graph-based lexical centrality as salience in text summarization," *journal of artificial intelligence research (JAIR)*, vol. 22, no. 1, pp. 457–479, 2004.

[19] S. Xie, B. Favre, D. Hakkani-Tür, and Y. Liu, "Leveraging sentence weights in a concept-based optimization framework for extractive meeting summarization." in *Proceedings of 10th Conference of the International Speech Communication Association (INTERSPEECH)*, 2009, pp. 1503–1506.

[20] K. Riedhammer, B. Favre, and D. Hakkani-Tür, "Long story short–global unsupervised models for keyphrase based meeting summarization," *Speech Communication*, vol. 52, no. 10, pp. 801–815, 2010.

[21] Y. Liu and D. Hakkani-Tür, "Speech summarization," *Spoken language understanding: Systems for extracting semantic information from speech*, pp. 357–396, 2011.

[22] I. McCowan, S. Bengio, D. Gatica-Perez, G. Lathoud, F. Monay, D. Moore, P. Wellner, and H. Bourlard, "Modeling human interaction in meetings," in *Proceedings of the International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, vol. 4. IEEE, 2003, pp. 748–751.

[23] S. Reiter and G. Rigoll, "Segmentation and classification of meeting events using multiple classifier fusion and dynamic programming," in *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, vol. 3. IEEE, 2004, pp. 434–437.

[24] A. Dielmann and S. Renals, "Automatic meeting segmentation using dynamic bayesian networks," *Multimedia, IEEE Transactions on*, vol. 9, no. 1, pp. 25–36, 2007.

[25] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal *et al.*, "The ami meeting corpus: A pre-announcement," in *Machine learning for multimodal interaction*. Springer, 2006, pp. 28–39.

[26] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke *et al.*, "The icsi meeting corpus," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, vol. 1. IEEE, 2003, pp. I–364.

[27] D. Beeferman, A. Berger, and J. Lafferty, "Statistical models for text segmentation," *Machine learning*, vol. 34, no. 1-3, pp. 177–210, 1999.

[28] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Proceedings of the ACL Workshop*, 2004, pp. 74–81.

[29] G. Murray and S. Renals, "Term-weighting for summarization of multi-party spoken dialogues," in *Machine Learning for Multimodal Interaction*. Springer, 2008, pp. 156–167.