A Rhetorical Syntax-Driven Model for Speech Summarization

Jian Zhang and **Pascale Fung**
Human Language Technology Center
Department of Electronic & Computer Engineering
Hong Kong University of Science & Technology (HKUST)
{zjustin, pascale}@ece.ust.hk

Abstract

We show a novel approach of parsing and reordering rhetorical syntax tree for extractive summarization of presentation speech. Our previous work showed (Fung et al., 2008) that rhetorical structures are embedded in this type of speech and that exploring this structure helps improve summarization quality. We further demonstrate that speakers do not follow the strict order of bullet points in the presentation slides, and that a re-ordering of these points occurs. We therefore propose a method of parsing presentation transcriptions into a rhetorical syntax tree and then re-order the leaf nodes to transform the speech transcriptions into an extractive summary, akin to a process of presentation slide generation. Chunking, parsing, and reordering are carried out by 28-class Hidden Markov Support Vector Machine(HMSVM) classifier trained from reference presentations and presentation slides. Using ROUGE-L F-measure we showed that our rhetorical syntax-driven model gives a 35.8% relative improvement over a binary summarizer with no rhetorical information, a 14.3% improvement over Rhetorical State Hidden Markov Model(RSHMM) (Fung et al., 2008), and a 4.3% improvement over our proposed model with no reordering.

1 Introduction

In this paper, we propose to improve extractive summarization of presentation speech using parsing and reordering of the salient points in the speech. Presentation speech includes classroom lectures, conference talks, business seminars, as well as political debates and parliamentary speech where the speaker gives a presentation according to some prepared slides containing bullet points. Some of the speech are transcribed into text, others might even be accompanied by short abstracts. Nevertheless, for learning and collaboration purposes, transcribed text is too long to read whereas short abstracts do not contain enough information (Teufel and Moens, 2002). Especially, in our conference presentation corpus, on average only about 40% bullet points of each transcription appears in the corresponding conference paper abstract. The accompanying presentation slides, on the other hand, are much better in summarizing the gists.

In recent years, more research has been conducted on exploring the hierarchical structure for better summarization performance (Fung et al., 2003; Murray et al., 2006; Sauper and Regina, 2009; Tatar et al., 2008). Unlike text documents, the structure of a spoken document is not immediately apparent in terms of its layout. However, researchers have shown that structural characteristics of a speech are clearly rendered by not just its linguistic features but also its acoustic features (Fung et al., 2008; Hirschberg and Nakatani, 1996; Nakatani et al., 1995). The hierarchical layout structure of Power Point slides enhances

Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pages 1299–1307,
Beijing, August 2010

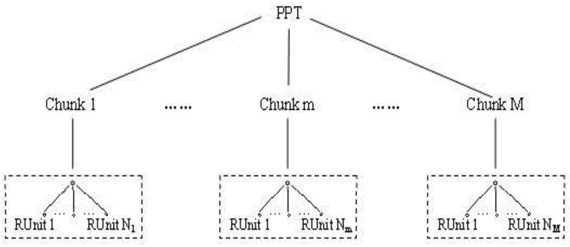


Figure 1: Rhetorical syntax tree

the understanding by the audience. In fact, they even provide a kind of extractive summarization that is superior in terms of informativeness than short abstracts. Unfortunately, presentation slides are not always made available to the audience or for the archive. In some cases, presentation slides consist of mostly figures and graphs, even videos, but without sufficient text bullet points to summarize the content. Meanwhile, there are significant amounts of presentation speech online (e.g. political speech, lectures and seminars) that can be rendered more useful if we can summarize them in a format similar to presentation slides.

Following previous research showing that modeling hierarchical structure indeed helps improve summarization performance, we are interested in going a step further in proposing a rhetorical syntax driven summarization model. Presentation speech is transcribed automatically by an ASR system, then "parsed" into a rhetorical syntax tree. The leaf nodes of the tree, representing actual utterances with rhetorical unit labels, are then "reordered" and organized into a target summary. In this paper, we also propose using a HMSVM (Altun et al., 2003) for the parser and the summarizer. HMSVM has the advantage of considering the interdependence between neighboring sentences. Reordering rules are automatically learned and candidate sequences generated, before they are scored by the final summarizer.

This paper is organized as follows: Section 2 describes our motivation, and the rhetorical structure characteristics in lecture speech. Section 3 details how to parse rhetorical structure of the lecture speech. Section 4 describes the reordering process. Section 5 then describes how to produce extractive summaries. We then describe the corpus, how to create reference summaries, the acoustic/prosodic, linguistic and discourse characteristics of lecture speech, baselines, and our in-house automatic speech recognition system are presented in Section 6. Section 7 presents the experiment results. Section 8 describes related work. We then conclude at the end of this paper.

2 Rhetorical Syntax Tree of Presentation Speech

Unlike conversational speech, lectures and presentations are planned. Lecture speakers follow a relatively rigid rhetorical structure at the document level: s/he starts with an overview of the topic to be presented, followed with the actual content with more detailed descriptions, and then concludes at the end. The speech is given in several coherent "chunks" corresponding to the talk outline. By looking at presentation slides as shown in Figure 1, we can clearly see the chunk boundaries, which always exist at slide transitions, delineate content changes. Each of the chunks, in turn, contains many coherent text spans, namely the rhetorical units. Each rhetorical unit contains one or more slides. We represent the rhetorical structure of presentations by a hierarchical text plan, or a rhetorical tree. Since lecture speeches are mostly based on presentation slides with main bullet points, the structural format of the presentation slides is a faithful representation of the document-level rhetorical structure of the speech. At the top level of the tree are the rhetorical chunks, and at the lower level child nodes are rhetorical units. Each of the chunks contains several rhetorical units, where each unit may contain a number of utterances corresponding to a list of bullet points in the slides.

We propose using the annotation labels commonly shared by most presentation speech for labeling rhetorical chunks as shown in the left column of Table 1. The chunk label definitions are derived from the general structure of presentations in the specific domain of our corpus, namely conference presentations. Note that whereas we chose to use 7 labels for presentation speech, label definitions are fairly obvious and easy to derive for other genres of speech. We use a machine-aided manual annotation method to label the training presentation speech data. Referring to the slides

Table 1: Rhetorical Chunk and Unit Description

Rhetorical Chunk	Rhetorical Unit
c_1 (Title)	title, author of the presentation
c_2 (Outline)	texture structure;
c_3 (Motivation)	aim; problem/phenomenon
c_4 (Related work)	rival/contrast; continuation
c_5 (Methodology)	solution/inventive step
c_6 (Experiment)	corpus description; detailed experimental setup
c_7 (Conclusion)	conclusion; future work

of each presentation, each sentence in the speech transcription is assigned a label corresponding to one of the 7 chunks defined in Table 1. First, all bullet point sentences in the slides are assigned a chunk label according to its section. Referring to our previous work, a Relaxed Dynamic Time Warping program (Zhang et al., 2008) is used to roughly align transcribed sentences to the corresponding slide bullet points. Chunk labels are also included in this alignment. Human inspection quickly corrects any alignment mistakes made by the program. We then label all the sentence of each type "c_i" rhetorical chunk as "c_i". For example, the sentences of a rhetorical chunk which describes "Methodology: solution/inventive step" is labeled as "c_5".

Rhetorical units are described in the right column of Table 1. The rhetorical units, as we explain below, are clustered automatically without explicit labels. These rhetorical units correspond more or less to the definitions in Table 1, without human effort. To obtain the reference rhetorical unit labels of each type of rhetorical chunks in the entire experiment corpus described in Section 6.1, we cluster all utterances that belong to the same chunk in all presentation speech into several rhetorical units by using modified k-means (MKM) clustering algorithm (Wilpon and Rabiner, 1984; Fung et al., 2003). MKM starts from the centroid of all utterances in one rhetorical chunk and splits the clusters top down until the sub-clusters stabilize. Each final cluster represents one rhetorical unit. The clustering algorithm is shown as follows.

Given all utterances within the same type of chunk from all presentation speech:
(1) **Compute** the centroid;
(2) **Assign** sentence feature vectors closest to each centroid to its cluster;
(3) **Update** each centroid feature vector using all sentence feature vectors assigned to each cluster;
(4) **Iterate** step(2) to step (4) until sentence feature vectors stop moving between clusters;
(5) **Stop** if clusters stabilizes, and **get** final clusters, else **goto** step (6);
(6) **Split** the cluster with largest intra-cluster distance into two by finding the pair of vectors as new centroids, and **repeat** steps (2) to step (5).

Using the above algorithm, we find out the following rhetorical unit clusterings of different kinds of rhetorical chunks on our experiment corpus: (1) two rhetorical units in "Title"(c_1) chunk: "r_i:c_1, $i = 1, 2$";
(2) three rhetorical units in "Outline"(c_2) chunk and "Conclusion"(c_7) chunk:
"r_i:c_j, $i = 1, 2, 3; j = 2, 7$";
(3) five rhetorical units in "Motivation"(c_3) chunk,
"Related work"(c_4) chunk,
"Methodology"(c_5) chunk,
and "Experiment"(c_6) chunk:
"r_i:c_j, $i = 1, 2, 3, 4, 5; j = 3, 4, 5, 6$".

These rhetorical unit clusters correspond roughly to the definitions in the right hand column of Table 1, though no manual labeling is involved.

3 Parsing Presentation Speech

By using our rhetorical syntax-driven model, the process of parsing presentation speech can be described as follows. First we extract acoustic/prosodic features from presentation speech, and linguistic, discourse features from the ASR transcribed text. These features are described in Section 6.1. Next we parse the presentation speech into rhetorical units. Given the ASR transcription of a presentation speech, our task is to parse the transcription sentences into chunks and then into rhetorical units (leaf nodes) that roughly correspond to the rhetorical chunks and units in Table 1 according to their feature vectors. We consider the parsing process as a multi-class classification problem. Each rhetorical unit of each rhetorical chunk is represented by one class.

Considering that HMSVM (Altun et al., 2003) combines the advantages of maximum margin classifier and kernels with the elegance and efficiency of HMMs, and can effectively handle the dependency between neighboring sentences, we train a twenty-eight-class HMSVM classifier for parsing speech, with one class representing each rhetorical unit. As an example, the sentences labeled as "r_2:c_5" belong to the second rhetorical unit of "Methodology"(c_5) chunk. We have found that, by looking at our corpus of conference presentations, speakers indeed follow the "chunk order" of the slides they use. We add some constraints existing between "r_m:c_i" and "r_n:c_j" where "$i \neq j$" according to the "chunk order" of the slides. Function f_1 maps each given speech or transcription **y** to a rhetorical unit label sequence **z**. For example we want to learn a discriminant function $F_1 : \mathcal{Y} \times \mathcal{Z} \rightarrow \mathcal{R}$ over input/output pairs from which we produce a prediction by maximizing F_1 over the output variable for a given input **y**. The general form of our hypotheses f_1 is:

$$\mathbf{z}^* = f_1(\mathbf{y}; \mathbf{w}) = \underset{\mathbf{z} \in \mathcal{Z}}{\operatorname{argmax}} F_1(\mathbf{y}, \mathbf{z}; \mathbf{w}) \quad (1)$$

where **w** denotes a weighting parameter vector to learn.

We assume F_1 to be linear in some combined feature representation of inputs, described in Section 6.1, and outputs $\Psi(\mathbf{y}, \mathbf{z})$. We then get $F_1(\mathbf{y}, \mathbf{z}; \mathbf{w}) = \langle \mathbf{w}, \Psi(\mathbf{y}, \mathbf{z}) \rangle$. Moreover, we apply a kernel function K over the joint input/output space such that:

$$K((\mathbf{y}, \mathbf{z}), (\bar{\mathbf{y}}, \bar{\mathbf{z}})) = \langle \Psi(\mathbf{y}, \mathbf{z}), \Psi(\bar{\mathbf{y}}, \bar{\mathbf{z}}) \rangle \quad (2)$$

Ψ can be written as a sum over the length of the sequence and decomposed as:

$$\Psi(\mathbf{O}, \mathbf{z}) = (\sum_{i=1}^{l(\mathbf{O})} \Psi_{\sigma,\tau}(\upsilon_i, \upsilon_{i+1}, \mathbf{O}, i))_{\sigma,\tau \in \gamma} \quad (3)$$

where γ is the rhetorical unit label set. $l(\mathbf{O})$ is the length of the observation sequence **O** in our case. Ψ is composed by mapping functions that depend only on labels at position i and $i+1$, **O** as well as i (Markov property).

We then rewrite F_1 using $\mathbf{w} = (\mathbf{w}_{\sigma,\tau})_{\sigma,\tau \in \gamma}$ as Equation 4.

$$\begin{aligned} F_1(\mathbf{O}, \mathbf{z}) &= \sum_{\sigma,\tau \in \gamma} \langle \mathbf{w}_{\sigma,\tau}, \sum_{i=1}^{l(\mathbf{O})} \Psi_{\sigma,\tau}(\upsilon_i, \upsilon_{i+1}, \mathbf{O}, i) \rangle \\ &= \sum_{i=1}^{l(\mathbf{O})} \underbrace{\sum_{\sigma,\tau \in \gamma} \langle \mathbf{w}_{\sigma,\tau}, \Psi_{\sigma,\tau}(\upsilon_i, \upsilon_{i+1}, \mathbf{O}, i) \rangle}_{=: g(\upsilon_i, \upsilon_{i+1}, \mathbf{O}, i)} \end{aligned} \quad (4)$$

In decoding process, using this decomposition (Altun et al., 2003) we can define

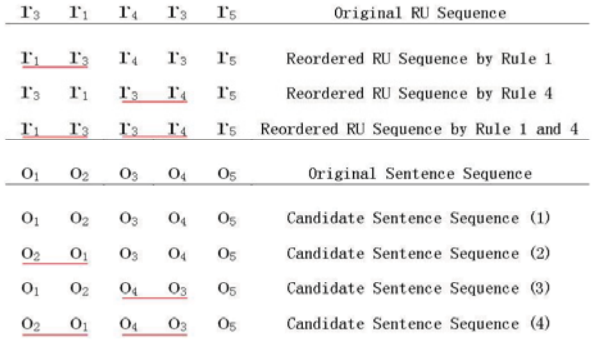
$$\begin{aligned} V(i, \upsilon) := \max_{\upsilon' \in \gamma}(V(i-1, \upsilon')) + g(\upsilon', \upsilon, \mathbf{O}, i-1) \\ when\ i > 1 \\ or\ := 0 \quad otherwise \end{aligned} \quad (5)$$

as the maximal score for all labels with label υ at position i. Using dynamic programming we compute $\max_{\upsilon \in \gamma} V(l(\mathbf{O}), \upsilon)$. The optimal label sequence is recovered by backtracking.

4 Reordering Rhetorical Unit Sequence

4.1 Extracting Reordering Rules

We found that, by looking at our corpus of conference presentations described in Section 6.1, presentation speakers do not always follow the bullet point order within a chunk. When demonstrating a current slide they may be already introducing the next slide. About 11% of the rhetorical units in the transcriptions are out of order vis-a-vis the corresponding bullet points in the presentation slide. As an integral part of our rhetorical syntax-driven summarization model, the rhetorical unit sequence and consequently the sentence sequence are reordered within a chunk. The extraction of reordering rules is based on the alignment between source rhetorical unit sequence in the speech transcription and target rhetorical unit sequence in the Power Point slide sentences. Each sentence is represented by its rhetorical unit label. For example, from the training set and development set in one of our held-out experiment settings described in Section 7, we extracted the following reordering rules: (1)$(r_3, r_1) \rightarrow (r_1, r_3)$;



r3	r1	r4	r3	r5	Original RU Sequence
r1	r3	r4	r3	r5	Reordered RU Sequence by Rule 1
r3	r1	r3	r4	r5	Reordered RU Sequence by Rule 4
r1	r3	r3	r4	r5	Reordered RU Sequence by Rule 1 and 4
o1	o2	o3	o4	o5	Original Sentence Sequence
o1	o2	o3	o4	o5	Candidate Sentence Sequence (1)
o2	o1	o3	o4	o5	Candidate Sentence Sequence (2)
o1	o2	o4	o3	o5	Candidate Sentence Sequence (3)
o2	o1	o4	o3	o5	Candidate Sentence Sequence (4)

Figure 2: Candidate sentence sequences after applying reordering rules

(2)$(r_3, r_2) \rightarrow (r_2, r_3)$; (3)$(r_4, r_2) \rightarrow (r_2, r_4)$; (4)$(r_4, r_3) \rightarrow (r_3, r_4)$; (5)$(r_5, r_3) \rightarrow (r_3, r_5)$; (6)$(r_5, r_4) \rightarrow (r_4, r_5)$.

In each reordering rule, the left item represents rhetorical unit sequence of the transcription sentences while the right item represents rhetorical unit sequence of bullet points of the corresponding Power Point slides. From these reordering rules, we can see that the speakers in our corpus talk about content described by future bullet points (i.e in the subsequent rhetorical units), but never seem to repeat content from bullet points in the previous unit(s).

4.2 Applying Reordering Rules

Given a sentence sequence and its corresponding rhetorical unit sequence within each chunk, from left to right, with a shifting window of two, we search for the matching reordering rule and adjust the order of the sentences one matched rule at a time, yielding a set of at most 2^L sentence sequence candidates for each chunk where L equals to the length of the sentence. From our data, we found that there are at most 2 matched rules per sentence sequence. So including the original sequence, at most 4 candidate sequences are generated for each chunk.

A reordering example is shown in Figure 2. We apply the reordering rules on a sentence sequence "$(o_1, o_2, o_3, o_4, o_5)$" and the corresponding unit sequence "$(r_3, r_1, r_4, r_3, r_5)$". Four candidate reordered sentence sequence are produced. Without any reordering, we get "Candidate Sentence Sequence (1)". Using reordering Rule 1, we get "Candidate Sentence Sequence (2)".

5 Rhetorical Syntax-driven Summarization

Following sentence reordering, the extractive summarizer selects salient sentences from each chunk using a binary-class classifier. The classifier is run over all candidate sequences from Section 4 and the system selects the best sequence and its summary sentences according to the output probability of the classifier. The best sequence $\{o_1...o_i...o_k\}$ satisfies

$$\text{argmax} \sum_{i=1}^{k} \lg P(o_i \in summary\ sentence\ set|c_j) \quad (6)$$

where c_j represents the rhetorical chunk c_j which has several candidate sequences, including the sequence $\{o_1...o_i...o_k\}$.
$P(o_i \in summary\ sentence\ set|c_j)$ is output probability of that the sentence o_i in the rhetorical chunk c_j is summary sentence.

Again, an HMSVM classifier is used at this stage. The sentence feature vector **o** now has its rhetorical unit label as an additional feature, to yield a new sentence feature vector $\hat{\mathbf{o}}$. For the sentence vector sequence **z** of each chunk, we label it by using the optimal function $F_2(\mathcal{Z}, \mathcal{V})$. The training stage is similar to that of training the HMSVM parser. The difference is that the HMSVMs for summarization are binary classifiers, while the HMSVM parser is a multi-class classifier.

6 Experimental Setup

6.1 Corpus

We use a lecture speech corpus containing wave files of 71 presentations recorded from different mandarin speakers at two technical conferences, together with well-formatted Power Point slides, manual transcriptions, and their associated audio data. Each presentation lasts about 15 minutes on average. The 71 presentations are split into 391 chunks, and each sentence is assigned a rhetorical chunk label, using the machine-aided human labeling method as described in Section 2. Each chunk has on average 4.3 rhetorical units. The reference rhetorical unit labels are created by using unsupervised MKM algorithm described in Section 2. Since the labeling process also yields an alignment path between transcription sentences and Power Point slide bullet points, we extract those sentences that have the highest alignment scores with the bullet points to form reference summary sentences, then corrected by five human subjects according to the rhetorical chunk descriptions in the right column of Table 1. We use the kappa coefficient (Krippendorff, 1980) for measuring stability of each annotator and reproducibility between each pair of annotators. The average kappa coefficient is higher than 0.85.

6.2 Features and Baselines

We use the discourse feature PossionNoun proposed in our previous work (Zhang et al., 2008) which is based on the following assumptions: first, if a sentence contains new noun words, it probably contains new information. The noun word's Poisson score varies according to its position. We use Poisson distribution to approximate the variation. Second, if a noun word occurs frequently, it is likely to be more important than other noun words, and the sentence with these high frequency noun words should be included in a summary. We also use the following acoustic and linguistic features for representing sentences. The acoustic features are: duration of the sentence, average syllable duration of the sentence, F0 and Energy min/max/mean/slope/range value of the sentence. The linguistic features are: sentence word count, TF/IDF of each word in the sentence, and the word identity in each sentence.

We use three alternate summarization models for comparison. One is a binary classifier without any rhetorical information, one class for summary sentence and the other for non-summary sentence. The second is RSHMM (Fung et al., 2008). The third is our rhetorical syntax tree without reordering. The two above are built by using acoustic, linguistic, and discourse features for representing the sentences. We apply rhetorical unit label as an additional feature for building our proposed models(with/without reordering).

Table 2: Summarization average performance of 6-fold cross validation experiment in ROUGE-L F-measure (F-measure)

Bianry	RSHMM	Without reordering	Syntax tree with reordering
.53(.52)	.63(.56)	.69(.61)	.72(.65)

(1)Binary: binary classifier as baseline
(2)RSHMM: Rhetorical State HMM proposed in our previous work (Fung et al., 2008)
(3)without reordering: rhetorical syntax tree based summarizer without reordering
(4)Syntax tree with reordering: rhetorical syntax tree based summarizer with reordering

6.3 Lecture Speech ASR Transcription System

We apply our rhetorical syntax-driven summarization model for ASR transcriptions. The database for building our in-house ASR system contains 29 hours of audio data from the technical conferences in our corpus. We choose approximately 21 hours of speech as the training data. The test data comprises of 12 presentations with approximately 3 hours of audio data. Our decoding system runs in multiple passes. Automatic segmentation is first performed on the lecture speech audio data. This is followed by bigram decoding with the gender independent (GI) acoustic model (AM) and lattice generation. Trigram and four-gram branches are created for AM adaptation through lattice expansion and rescoring. Re-decoding with both adapted AM and adapted language model (LM) is performed to produce 1-best results. System combination via recognizer output voting error reduction scheme (ROVER) (Fiscus, 1997) is employed by using character based alignment from the trigram and four-gram branch outputs. The final system obtains a recognition performance of 79.2% character accuracy.

7 Experimental Results

For evaluating different summarization systems, we perform 6-fold cross validation experiments, and two held-out experiments. There are many kinds of metrics for evaluating speech summarization performance (Zhu and Penn, 2005; Penn and Zhu, 2008). We choose ROUGE-L F-

Table 3: Summarization performance of held-out experiments in ROUGE-L F-measure (F-measure)

(A) on manual transcription

Binary	RSHMM	Without reordering	Syntax tree with reordering
.50(.48)	.61(.52)	.67(.59)	.69(.61)

(B) on ASR transcription

Binary	RSHMM	Without reordering	Syntax tree with reordering
.46(.43)	.59(.50)	.66(.57)	.68(.61)

(1)Binary: binary classifier as baseline
(2)RSHMM: Rhetorical State HMM proposed in our previous work (Fung et al., 2008)
(3)without reordering: rhetorical syntax tree based summarizer without reordering
(4)Syntax tree with reordering: rhetorical syntax tree based summarizer with reordering

measure (Lin, 2004) and F-measure (Van Rijsbergen, 1979) as evaluation metrics in our experiments. 11 documents from the 71 presentations are excluded as our development set. In the 6-fold cross validation experiments, we divide the remaining 60 presentations into six subsets of equal size. For each experiment, we use five subsets to train all models and the remaining subset for testing. The average performance of these 6-fold cross validation experiments is shown in Table 2.

Among these 60 presentations, 50 are randomly selected as training data for the two held-out experiments, while the remaining ten are used as test data. Table 3-(A) shows the result of the one held-out experiment on manual transcriptions of test data. Table 3-(B) shows the other held-out experiment on ASR transcriptions of test data.

From these results, we can see that the proposed rhetorical syntax-driven summarizer with reordering outperforms all other methods. Table 2 shows that our rhetorical syntax-driven model gives a 35.8% relative improvement over a binary summarizer with no rhetorical information, a 14.3% improvement over RSHMM (Fung et al., 2008), and a 4.3% improvement over our proposed model with no reordering. These findings suggest that our rhetorical syntax-driven summarization model built by using binary HMSVM classifier apply the sequence information of the sentences within the same rhetorical chunk for improving summarization performance because of the Markov property of HMSVM.

In the above experiments, we all use the rhetorical unit labels produced by our rhetorical structure parser for improving summarization performance. The average accuracy of the rhetorical structure parser is about 83.2%. When we use the reference rhetorical unit labels on our cross-validation experiments, the average summarization performance is 0.75 of ROUGE-L F-measure, a 4.2% improvement over that using the rhetorical unit labels produced by the rhetorical structure parser.

Although the overall performance on ASR transcriptions is worse than that of manual transcriptions, the performance is also satisfying. Furthermore, we also find that using only acoustic features our model obtains satisfying result, 0.65 of ROUGE-L F-measure, on 6-fold cross validation experiment.

8 Related Work

(Furui et al., 2008) has shown that feature-based extractive summarization is an approach that is efficient and more effective than MMR-based approach for lecture speech summarization.

(Marcu, 1997) described the first experiment that shows the concepts of rhetorical analysis and nuclearity can be used effectively for text summarization. (Fung et al., 2003) presented a stochastic HMM framework with modified K-means and segmental K-means algorithms for extractive text summarization. (Fung and Ngai, 2006) further presented a stochastic Hidden Markov Story Model for multilingual and multi-document summarization and proposed that monolingual documents recounting the same story (i.e., in the same topic) share a unique story flow (one story, one flow), and such a flow can be modeled by HMMs. (Barzilay and Lee, 2004) presented an unsupervised method for the induction of content models, which capture constraints on topic selection and organization for texts in a particular domain (Branavan et al., 2007) proposed a structured discriminative model for table-of-contents generation on written text that accounts for a wide range of phrase-based and collocation features. (Eisenstein and Barzilay, 2008) describes a novel Bayesian approach to unsupervised lexical cohesion driven topic segmentation.

Many researchers have suggested that rhetorical information also exists in spoken documents and efficient modeling of this information is helpful to the summarization task. (Tatar et al., 2008) and (AKITA and Kawahara, 2007) used the Hearst method (Hearst, 1997) to segment documents and detect topics for text summarization and topic adaptation of speech recognition systems for long speech archives respectively. (Hirohata et al., 2005) consider that humans tend to summarize presentations by extracting important sentences from introduction and conclusion sections, and further propose a summarization method based on this structural characteristic. They estimated the introduction and conclusion section boundaries based on the Hearst method (Hearst, 1997), using sentence cohesiveness which is measured by a cosine value between content word-frequency vectors, before performing summarization. Teufel and Moens (Teufel and Moens, 2002) also proposed "rhetorical status" as summary unit, but without hierarchical structure or reordering. Furthermore, many linguists believe that speech acoustics contribute to rhetorical and discourse structure. (Nakatani et al., 1995) provide empirical evidence that discourses can be segmented reliably, and that acoustic characteristics are used by speakers to convey linguistic structure at the discourse level in the English domain. There is a large amount of previous work seeking to demonstrate that acoustic prosodic profile of speech closely models its discourse or rhetorical structure (Halliday, 1967; Ladd, 1996; Hirschberg and Nakatani, 1996).

Our work described in this paper is closely related to (Marcu, 1997; Teufel and Moens, 2002) in that we propose using rhetorical units for summarization. Our method differs from (Teufel and Moens, 2002) in that we assume the relevance or saliency and function of certain text pieces can be determined by analyzing the full hierarchical structure of the text. Instead of annotating the training data with rhetorical labels manually, we propose using Power Point slides as references. (He et al., 2000; He et al., 1999) also investigate the correlation between Power Point slides and extractive summaries. Our learning method is based on classifiers, while (Marcu, 1997) uses rule-based method for parsing rhetorical structure. We train our rhetorical parser by using those reference rhetorical units of the transcriptions created by aligning Power Point slides. We propose a syntax-driven parsing model with reordering for summarization, and we propose a different classifier, HMSVM, for handling the dependency between neighboring sentences within each chunk, when we accomplish our summarization task.

9 Conclusion and Discussion

In this paper, we have shown a rhetorical syntax-driven summarization method for presentation speech. In view of the fact that rhetorical structure in speech is inherently hierarchical, our method chunks, parses, and reorders the presentation utterance sentences before selecting some of them as summary sentences.

We have proposed to use HMSVM classifiers for the parser and the summarizer, taking into account the dependency between neighboring chunks, rhetorical units and sentences with Markov property. Our rhetorical syntax-driven summarizer with reordering outperforms a binary summarizer without rhetorical information with 35.8% relative improvement and outperforms RSHMM (Fung et al., 2008) with 14.3% relative improvement. It gives a 4.3% relative improvement over the same model without reordering. For future work, we are interested in investigating how to apply our rhetorical syntax-driven method to other genres of speech, such as meetings and parliamentary speech.

10 ACKNOWLEDGEMENT

This work is partially supported by ITS/189/09 of the Hong Kong Innovation and Technology Fund(ITF). The authors would like to thank Chan Ho Yin for his valuable input to this work.

References

AKITA, Y., Nemoto Y. and T. Kawahara. 2007. PLSA-based topic detection in meetings for adaptation of lexicon and language model. *Proc. Interspeech 2007*, pages 602–605.

Altun, Y., I. Tsochantaridis, and T. Hofmann. 2003. Hidden Markov Support Vector Machines. In *Machine Learning-International Workshop Then Conference-*, volume 20.

Barzilay, R. and L. Lee. 2004. Catching the drift: Probabilistic content models, with applications to generation and summarization. *Proceedings of HLT-NAACL*, pages 113–120.

Branavan, SRK, P. Deshpande, and R. Barzilay. 2007. Generating a table-of-contents. In *ANNUAL MEETING-ASSOCIATION FOR COMPUTATIONAL LINGUISTICS*, volume 45, page 544.

Eisenstein, J. and R. Barzilay. 2008. Bayesian unsupervised topic segmentation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 334–343. Association for Computational Linguistics.

Fiscus, J.G. 1997. A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER. In *In Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding*.

Fung, P. and G. Ngai. 2006. One story, one flow: Hidden Markov Story Models for multilingual multidocument summarization. *ACM Transactions on Speech and Language Processing (TSLP)*, 3(2):1–16.

Fung, P., G. Ngai, and P. Cheung. 2003. Combining optimal clustering and hidden Markov models for extractive summarization. *Proceedings of ACL Workshop on Multilingual Summarization*, pages 29–36.

Fung, P., H.Y. Chan, and J.J. Zhang. 2008. Rhetorical-State Hidden Markov Models For Extractive Speech Summarization. *ICASSP2008. Proceedings*, pages 4957–4960.

Furui, Y., K. Yamamoto, N. Kitaoka, and S. Nakagawa. 2008. Class Lecture Summarization Taking into Account Consecutiveness of Important Sentences. *Proceedings of Interspeech 2008*, pages 2438–2441.

Halliday, M.A.K. 1967. *Intonation and grammar in British English*. Mouton.

He, L., E. Sanocki, A. Gupta, and J. Grudin. 1999. Autosummarization of audio-video presentations. In *Proceedings of the seventh ACM international conference on Multimedia (Part 1)*, page 498. ACM.

He, L., E. Sanocki, A. Gupta, and J. Grudin. 2000. Comparing presentation summaries: slides vs. reading vs. listening. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 177–184. ACM New York, NY, USA.

Hearst, M.A. 1997. TextTiling: Segmenting Text into Multi-paragraph Subtopic Passages. *Computational Linguistics*, 23(1):33–64.

Hirohata, M., Y. Shinnaka, K. Iwano, and S. Furui. 2005. Sentence extraction-based presentation summarization techniques and evaluation metrics. *Proc. ICASSP2005*, 1.

Hirschberg, J. and C.H. Nakatani. 1996. A prosodic analysis of discourse segments in direction-giving monologues. *Proceedings of the 34th conference on Association for Computational Linguistics*, pages 286–293.

Krippendorff, K. 1980. Content analysis: An introduction to its methodology. *Beverly Hills,: Sage Publications*.

Ladd, D.R. 1996. *Intonational Phonology*. Cambridge University Press.

Lin, C.Y. 2004. Rouge: A Package for Automatic Evaluation of Summaries. *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*, pages 25–26.

Marcu, D. 1997. From discourse structures to text summaries. In *Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, pages 82–88.

Murray, G., M. Taboada, and S. Renals. 2006. Prosodic correlates of rhetorical relations. In *Proceedings of the Analyzing Conversations in Text and Speech (ACTS) Workshop at HLT-NAACL*, pages 1–7.

Nakatani, C.H., J. Hirschberg, and B.J. Grosz. 1995. Discourse structure in spoken language: Studies on speech corpora. *AAAI 1995 Spring Symposium Series: Empirical Methods in Discourse Interpretation and Generation*, pages 106–112.

Penn, G. and X. Zhu. 2008. A critical reassessment of evaluation baselines for speech summarization. *Proceedings of ACL-HLT. Columbus, OH*.

Sauper, C. and B. Regina. 2009. Automatically Generating Wikipedia Articles: A Structure-Aware Approach. In *Proceedings of the ACL 2009*, pages 208–216.

Tatar, D., E. Tamaianu-Morita, A. Mihis, and D. Lupsa. 2008. Summarization by Logic Segmentation and Text Entailment. *Advances in Natural Language Processing and Applications*, pages 15–26.

Teufel, S. and M. Moens. 2002. Summarizing scientific articles: experiments with relevance and rhetorical status. *Computational Linguistics*, 28(4):409–445.

Van Rijsbergen, CJ. 1979. Information Retrieval. *Buttersworth, London*.

Wilpon, JG and LR Rabiner. 1984. A modified K-means clustering algorithm for use in speaker-independent isolated word recognition. *The Journal of the Acoustical Society of America*, 75:S93.

Zhang, J.J., S. Huang, and P. Fung. 2008. RSHMM++ for extractive lecture speech summarization. In *IEEE Spoken Language Technology Workshop, 2008. SLT 2008*, pages 161–164.

Zhu, X. and G. Penn. 2005. Evaluation of sentence selection for speech summarization. In *ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization*. Citeseer.