Segmentation and Classification of Meeting Events using Multiple Classifier Fusion and Dynamic Programming

Stephan Reiter and Gerhard Rigoll
Institute of Human-Machine-Communication
Technische Universität München
Arcisstr. 21, 80290 München, Germany
{reiter, rigoll}@ei.tum.de

Abstract

In this paper the segmentation of a meeting into meeting events is investigated as well as the recognition of the detected segments. First the classification of a meeting event is examined. Five different classifiers are combined through multiple classifier fusion. Then a way for finding the optimal segment boundaries is presented. With a Dynamic Programming approach quite encouraging results can be obtained. The results show further that by classifier fusion a more stable result can be achieved than using only one single classifier.

1. Introduction

In the everyday life of organizations meetings are an important part. Usually meeting minutes are taken in order to preserve the most important issues for those who were not able to attend the meeting. Nowadays there is a growing interest in automatically deriving a meeting protocol. With this it should be possible to obtain the relevant information without the need to watch the whole video or listen to the entire recording.

A number of groups are concerned with developing a meeting recorder or a meeting browser system. In the meeting project at ICSI [7], for example, the main goal is to produce a transcript of the speech. At CMU the intention is to develop a meeting browser, which includes challenging tasks like speech transcription and summarization [9] and the multimodal tracking of people throughout the meeting [1], [8]. Microsoft is developing a distributed meeting system that provides features like teleconferencing and recording of meetings [2]. In the European research project M4, in which this work is integrated, the main concern is the construction of a demonstration system to enable structuring, browsing and querying of an archive of automatically analyzed meetings. At one of the partner sites of the M4 project the human interaction is modeled by using a dynamic approach [5].

Due to the complex information flow of visual, acoustic and other information sources in meetings (e.g. from documents or beamers) the segmentation of a meeting in appropriate sections represents a very challenging pattern recognition task, which is currently investigated by only a few research teams.

In this paper we present a method to divide a meeting into meeting events like discussion, monologue, note-taking, white-board activities and presentations, using multiple classifier fusion and dynamic programming.

The paper is organized as follows. Section 2 describes the meeting data. In Section 3 the used classifiers and the fusion technique are discussed. Section 4 then presents the segmentation of the meetings using dynamic programming.

2. Meeting Data

For our experiments with meeting segmentation and meeting event recognition special scripted meetings were recorded in the IDIAP Smart Meeting Room. This is a $8.2\,\mathrm{m} \times 3.6\,\mathrm{m} \times 2.4\,\mathrm{m}$ rectangular room containing a $4.8\,\mathrm{m} \times 1.2\,\mathrm{m}$ rectangular meeting table. The room is equipped with fully synchronized multichannel audio and video recording facilities. Each participant has a close-talk lapel microphone attached to his clothes. Additionally a microphone array on top of the table was used. Three closed-circuit television video cameras provide PAL quality video signals that are recorded onto separate digital video tape recorders. For full details of the hardware setup see [6].

The recorded meetings consist of a set of predefined meeting events in a specific order. The appearing meeting events were

- Monologue (one participant speaks continuously without interruption)

From: ICPR 2004, Cambridge, UK, IEEE Computer Society

- Discussion (all participants engage in a discussion)
- Note-taking (all participants write notes)
- White-board (one participant at front of room talks and makes notes on the white board)
- Presentation (one participant at front of room makes a presentation using the projector screen)

A total of 53 scripted meetings with two disjoint sets of meeting participants were recorded. The complete recording task is specified in [5].

Originally the idea was to take advantage of the results of single specialized recognizers, like speech and gesture recognizers, intent and emotion recognizers, person identifiers and source localization and tracking methods and use them to derive the higher semantic items, the meeting events. However, most of these specialized recognizers exist at most at a developing level, only the speaker segmentation extracted from the audio signal was available. So the remaining necessary features were labeled by hand to represent the results that were not available.

3. Classification

This section presents the various classifiers that were used, as well as the fusion technique that was applied for getting best recognition result. The distinguished meeting events were discussion, monologue, note-taking, presentation and whiteboard. Additionally the leading actor was distinguished at monologues, presentations and whiteboards. As monologues could be performed by each of the 4 participants whereas presentations and whiteboards were restricted to only two participants of the meetings, there is a resulting number of 10 classes. From the scripted meetings 30 videos (corresponding 144 events) where taken for training purposes, the remaining 23 videos (corresponding 122 events) served as evaluation data.

3.1. Classifiers

For the classification task we use a number of various classifiers. Since we use a static feature vector, the classifiers for dynamic modeling are not applicable. So the classifiers we used were the following:

- a simple hybrid Bayesian Network (BN) consisting of a discrete node as parent with five states (one for each meeting event) and nine continuous nodes directly connected to the parent node, representing the nine dimensions of the feature vector,
- Gaussian Mixture Models (GMM) with various numbers of Gaussians depending on the number of training material,
- a Neural Net with Multilayer Perceptrons (MLP) with 3 layers,
- a Radial Basis Network (RBN)with maximum 10 neurons,
- Support Vector Machines (SVM) with RBF-Kernel.

Classifier	Recognition Rate
BN	95.90 %
GMM	88.04 %
MLP	96.72 %
RBN	97.54 %
SVM	97.54 %
FUSED	96.72 %

Table 1. Recognition rates of the classifiers (BN: Bayesian Network, GMM: Gaussian Mixture Models, MLP: Multilayer Perceptron Network, RBN: Radial Basis Network, SVM: Support Vector Machines)

Each of the classifiers has been trained with the meeting events of the 30 training meetings. For evaluation purposes the remaining 23 meetings were used. At this stage the boundaries of the meeting events have been specified by hand, so the task is only the recognition of the event. In table 1 the recognition rates of each classifier is shown. Two classifiers (RBN and SVM) yield a quite good result with 97.54% whereas the GMMs seem not to be able to adapt well enough and achieve a recognition rate of 88.04%. One cause of this difference may be the relatively low number of training material available.

3.2. Classifier fusion

Classifier fusion is often used to enhance the recognition results of single recognizers. Here the goal is to provide more solid results throughout the recognition process. The fusion method is derived from a proposal of [4]. Each classifier i produces a pseudo-probability $d_{i,j} \in [0,1]$ for each class j by normalizing the output via a limitation function. Since this method requires no training, it is quite easy and quick to implement. These classifier outputs are organized in a decision profile (DP) as a matrix.

$$DP = \begin{bmatrix} d_{1,1} & d_{1,2} & d_{1,3} & d_{1,4} & d_{1,5} \\ d_{2,1} & d_{2,2} & d_{2,3} & d_{2,4} & d_{2,5} \\ d_{3,1} & d_{3,2} & d_{3,3} & d_{3,4} & d_{3,5} \\ d_{4,1} & d_{4,2} & d_{4,3} & d_{4,4} & d_{4,5} \\ d_{5,1} & d_{5,2} & d_{5,3} & d_{5,4} & d_{5,5} \end{bmatrix}; \qquad (1)$$

Here the $d_{i,j}$ is the pseudo probability of classifier i for class j. The rows represent the output of one classifier whereas

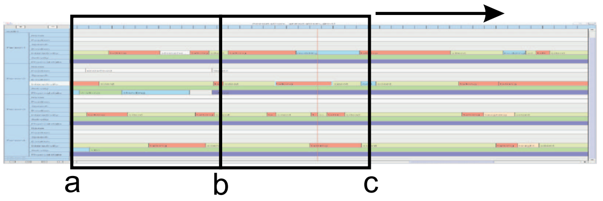

Figure 1. Two connected windows are shifted over the time scale to produce potential boundaries.

the columns show all probabilities for one class of all used classifiers. In this work we search the minimum of the decision profile columnwise and get the class C as the one with the maximum value as shown in Eq. 2.

$$\begin{aligned} \mu &= \left[\min(DP_{:,1}) \; \min(DP_{:,2}) \; \ldots \; \min(DP_{:,5})\right]; \\ C &= \arg\max \mu; \end{aligned} \qquad (2)$$

Also a pseudo-probability $\mu(C)$ is returned that reflects the support of the fused classifier for this class. Unfortunately this approach yields only a recognition rate that is as high as using the MLP (see table 1). A better result should be achieved if more training data was available. Until now we have only the mentioned 53 meetings. With more material the single recognizers could be trained on a distinct set of training data as it is recommended for fusion techniques.

4. Segmentation

The segmentation task is performed in two steps. At first, potential segment boundaries are searched; in the second step from all these possible boundaries those are chosen that give the highest overall score.

4.1. Finding the potential boundaries

First the possible boundaries have to be found. This is accomplished by the following procedure. Two connected windows with a length of 10 seconds each are shifted over the time scale as shown in Figure 1. Inside these two windows the feature vector is calculated and classified. If the results differ a potential segment boundary is assumed. In the same step a clustering of all found boundaries is performed. As long as the classification result $K(a, b)$ in the left window remains equal, the new assumed boundary is appended to the existing cluster $G\{i\}$. Otherwise a new cluster $G\{i+1\}$ is created. After that all clusters that contain less than three possible boundaries are discarded so that only important boundaries remain. Now we have a collection of arrays $G\{i\}, \; i = 1, \ldots, N$, where N is the number of clusters, consisting in the potential boundaries.

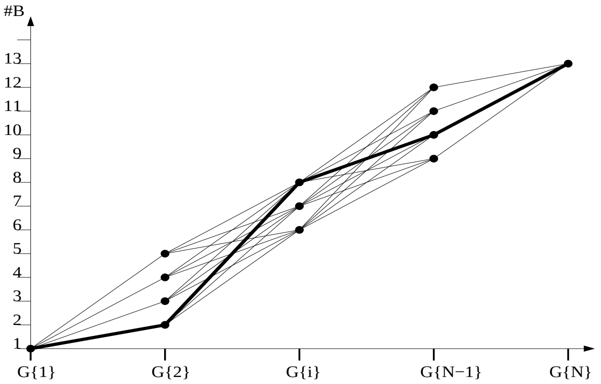

Figure 2. Finding the optimal boundaries; the path with the highest overall score if found through backtracking. The abscissa denotes the clusters of potential boundaries, the ordinate the number of the boundary.

4.2. Finding the optimal boundaries

Having found all boundaries that come into question, in each cluster $G\{i\}$ the in some sense 'best' boundary has to be chosen. This is accomplished via Dynamic Programming (DP). This approach assumes that the meeting events are mutually independent. So each boundary of a meeting event can be found if only the direct predecessor is known. The first and the last boundary are known a priori (beginning and end of the meeting), so the task is to choose the boundaries between that give the highest overall score. The score of a meeting event is calculated as the pseudo-probability that the classifier returns for the examined interval. As additional constraint only those boundaries could be chosen that ensure a minimum length of a meeting event of 15 seconds.

In figure 2 the procedure for finding the optimal segment boundaries is illustrated. For each boundary $x \in G\{i\}$ the score $s_x(y)$ to each boundary $y \in G\{i-1\}, \quad i = 2, \ldots, N$ is calculated. Then the maximum score $s_{max}$for each x is chosen.

$$s_{x,max} = \max \; s_x(y); \qquad (3)$$

The sum of this score and the overall score until $i-1$ is calculated and saved in a score-matrix $SG\{i\}$ together with the predecessor y.

$$SG\{i\} = \begin{bmatrix} \vdots & \vdots & \vdots \\ x & s_{x,max} + SG\{i-1\}_{y,2} & y \\ \vdots & \vdots & \vdots \end{bmatrix}; \qquad (4)$$

This is done for all clusters $G\{i\}$. Afterwards the best path through all score matrices is found through backtracking.

Classifier	Insertion	Deletion	Accuracy	Error
BN	0.1415	0.0383	9.2352	0.2143
MLP	0.1420	0.0217	9.9569	0.1960
RBN	0.1611	0.0083	8.7900	0.2091
FUSED	0.0942	0.0494	7.7776	0.1762

Table 2. Segmentation results (BN: Bayesian Network, MLP: Multilayer Perceptron Network, RBF: Radial Basis Network, FUSED: Fused Classifiers). The columns denote the insertion rate, the deletion rate, the accuracy in seconds and the classification error rate.

Starting with the last score matrix $SG\{N\}$, which contains only one boundary, and following the indices in the third column those boundaries are chosen that produce the best overall score. In a completing step two segments that contain the same meeting event are merged.

4.3. Segmentation results

As mentioned in section 2 there were 53 short scripted meetings available. From these 30 were chosen for the training of the classifiers, the remaining 23 were used for evaluation purposes. For all meetings an annotated script existed, from which the feature vector was calculated. These annotations contain single actions that occur in a meeting like the talking times of a person or the section when a participant is performing a presentation.

First we used only one classifier at once to segment the video streams. The results are shown in the upper part of table 2. In this table the error measures, derived from [3], show the insertion rates, the deletion rates, the accuracy of the detected boundaries in seconds and the classification error rate of the detected meeting events. In every column lower numbers denote better results. As can be seen the Radial Basis Network gives a pretty low deletion rate, but the accuracy with 8.79 seconds to over 9 seconds is quite poor. This means the average difference between the detected and the true boundary is of around 9 seconds.

In comparison with the fused classifiers we made the following observations. In three of the four error rates the fused classifiers yield a better result than any of the classifiers alone. The insertion rate decreases by about 30 %, the accuracy is about 2 seconds better and the classification error rate decreases by about 10 %.

5. Conclusion

In this work we presented an approach to automatically segment a meeting, that is available together with an annotation, by multiple classifier fusion and dynamic programming. In the single meeting event recognition task the fusion technique yielded no enhancement. An improvement should be achieved when more data is available.

In the combined segmentation and classification task the fused classifiers gave more accurate and stable results. Unfortunately some good results from single classifiers (especially the deletion rate) got worse but on the whole a better segmentation result with a lower insertion rate, a better accuracy and a lower classification error rate could be achieved.

These results show that the segmentation of meetings using the presented methods is possible. Enhanced results could be achieved by finding a better alternative to calculate the score for the dynamic programming. Also it would be possible to accomplish the segmentation with the fused classifiers and then use only a single one (e.g. SVMs) for the classification of the detected events.

References

[1] M. Bett, R. Gross, H. Yu, X. Zhu, Y. Pan, J. Yang, and A. Waibel. Multimodal meeting tracker. In *Proceedings of RIAO2000*, Paris, France, April 2000.

[2] R. Cutler, Y. Rui, A. Gupta, J. Cadiz, I. Tashev, L. wei He, A. Colburn, Z. Zhang, Z. Liu, and S. Silverberg. Distributed meetings: A meeting capture and broadcasting system broadcasting system. In *Proceedings of ACM Multimedia Conference*, 2002.

[3] S. Eickeler and G. Rigoll. A novel error measure for the evaluation of video indexing systems. In *IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Istanbul, Turkey, June 2000.

[4] L. I. Kuncheva, J. C. Bezdek, and R. P. Duin. Decision templates for multiple classifier fusion: An experimental comparison. *Pattern Recognition*, 34(2):299–314, 1999.

[5] 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 International Conference on Acoustics, Speech and Signal Processing*, Hong Kong, April 2003. IDIAP-RR 02-59.

[6] D. Moore. The idiap smart meeting room. IDIAP-COM 07, IDIAP, 2002.

[7] N. Morgan, D. Baron, J. Edwards, D. Ellis, D. Gelbart, A. Janin, T. Pfau, E. Shriberg, and A. Stolcke. The meeting project at icsi. In *Proceedings of the Human Language Technology Conference*, San Diego, CA, March 2001.

[8] R. Stiefelhagen. Tracking focus of attention in meetings. In *IEEE International Conference on Multimodal Interfaces*, Pittsburgh, PA, USA, October 14–16 2002.

[9] K. Zechner. Automatic generation of concise summaries of spoken dialogues in unrestricted domains. In *Proceedings of the 24th ACM-SIGIR International Conference on Research and Development in Information Retrieval*, New Orleans, LA, September 2001.