Multi-Stream Segmentation of Meetings

Alfred Dielmann and Steve Renals
Centre for Speech Technology Research
University of Edinburgh
Edinburgh EH8 9LW, UK
Email: {a.dielmann,s.renals}@ed.ac.uk

***Abstract*— This paper investigates the automatic segmentation of meetings into a sequence of group actions or phases. Our work is based on a corpus of multiparty meetings collected in a meeting room instrumented with video cameras, lapel microphones and a microphone array. We have extracted a set of feature streams, in this case extracted from the audio data, based on speaker turns, prosody and a transcript of what was spoken. We have related these signals to the higher level semantic categories via a multistream statistical model based on dynamic Bayesian networks (DBNs). We report on a set of experiments in which different DBN architectures are compared, together with the different feature streams. The resultant system has an action error rate of 9%.**

I. Introduction

Meetings form a major part of many professional activities, in which work is planned, problems are highlighted and solved, decisions are made, knowledge is shared, etc. Preserving and accessing [1] the information in such meetings is an important task, to enable a deeper understanding of meeting contents, to make links across meetings, and to disseminate knowledge to people who did not attend a meeting. By using multiple cameras and microphones, devices to capture handwritten notes and other varieties of recording equipment it becomes possible to record the multimodal information contained in a meeting. However, simply recording a meeting doesn't correspond to understanding what went on, and even relatively simple information access from meetings requires additional processing. Features corresponding to the communicative modalities (such as speech, gestures, handwriting and facial expressions) may be extracted from raw data streams. These individual feature streams can then be integrated to enable the identification of important events in a meeting.

In this paper we are concerned with the automatic segmentation of meetings into a set of predefined actions or phases: monologue (per speaker) , dialogue, note taking, presentation, presentation at the white-board [2]. This dictionary of meeting actions represents just one example of the possible points of view under which meetings can be analysed. Nevertheless it provides a useful first step in relating low level multimodal signals to higher level categories.

The following section will provide an overview about the meeting data set used in our experiments and the meeting collection process. Section III describes the feature set used to characterize these multi-party meetings. Three feature classes will be proposed: prosodic based features (III-A), location based speaker turn (III-B) and semantic based lexical features (III-C). Section IV gives an introduction to dynamic Bayesian networks (DBN) and their graphical formalism. The multi-stream DBN model adopted to segment a meeting into actions will be presented in section IV-A, and an enhanced version will be outlined in section IV-B. Finally in section V we propose and discuss some experimental results, achieved using four different configurations of our system.

II. Meeting collection

Our experiments have been performed using a corpus of thirty short meetings, recorded at IDIAP by Mc Cowan et al [2].[1] Each meeting has four participants and lasts about five minutes. The meeting structure was generated a priori, drawing "meeting actions" from the dictionary described above (extended with two further symbols: consensus and disagreement). Note that these symbols are mutually exclusive and exhaustive: only one "meeting action" at a time is feasible, and gaps between actions are not allowed. Although the broad progress ("agenda") of each meeting was scripted, the behaviour and interactions of the participants was natural. The meetings were recorded using three wall mounted cameras, an eight element circular microphone array and four lapel microphones (one for each participant). The recording conditions were realistic and without any constraint over factors such as noise, reverberation, cross-talk and visual occlusions.

III. Features

We used three classes of features in this work: prosodic features; speaker turn features; and lexical features. We have based our work mainly on speech and audio communicative modalities, since these are predominant in meetings; work in progress is using further streams based on video features.

A. Prosodic features

The prosodic features were based on a denoised and stylised version of the intonation contour [3], an estimate of the syllabic rate of speech [4] and the energy. These acoustic features comprise a 12 dimensional feature vector (3 features, 4 speakers), highlighting the currently active speakers and may indicate the level of engagement in the conversation for each participant.

[1]This corpus is publicly available from http://mmm.idiap.ch/

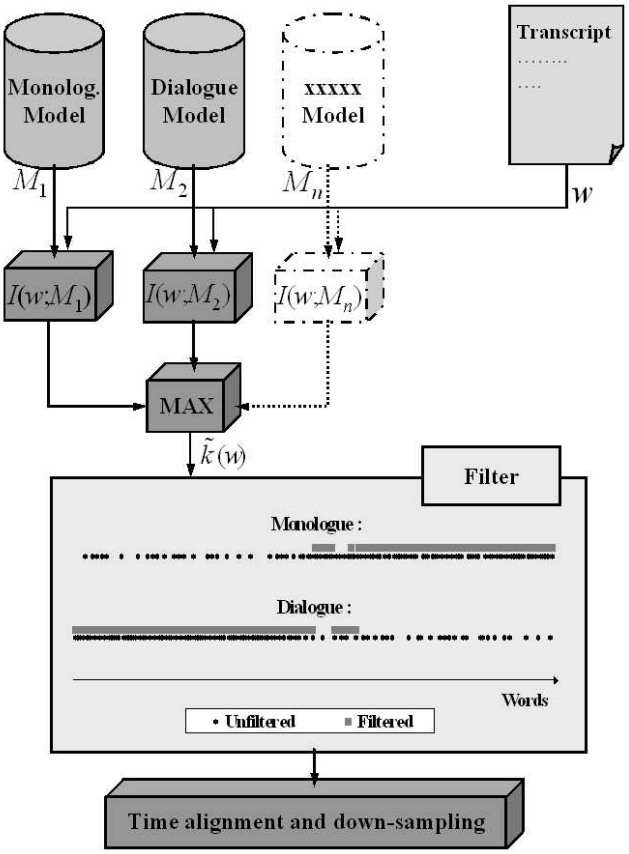

Fig. 1. Overview of the "lexical features" evaluation process

B. Speaker turn features

Information about the locations of the active speakers was extracted using a sound source localization process based on a microphone array. A 216 element feature vector resulted from all the 6^3 possible products of the 6 most probable speaker locations (four seats and two presentation positions) during the most recent three frames [5]. These features attempt to find statistical patterns in the conversational process, thus modelling how the interaction pattern evolves in time.

C. Lexical features

In addition to the lower level, continuous features outlined above, we have also used the transcript for each speaker, resulting in a feature stream consisiting of a sequence of words. In these experiments we use human generated transcriptions; work is in progress using automatic speech recognition on these meetings, but this is a challenging task due to non-native accents, natural speech, unconstrained topics and the fact that recordings were made on lapel and table-top microphones.

To correlate low-level text transcriptions with high level "meeting phases", the system outlined in figure 1 has been adopted. Monologue and dialogue classes were modelled using multinomial distributions over words (although the principles are valid for the other actions also). The mutual information between each word w in the transcript and the models M_k is computed, and the winning class $\tilde{k}$ is the one that maximizes mutual information:

$$\tilde{k}(w) = \arg\max_{k \in K} \{I(w; M_k)\}$$

Unfortunately the true classification output is concealed under a constellation of mis-classified words. Therefore the recognized sequence of symbols was filtered, leaving only the most frequent symbols. Smoothing was performed across a sliding window of 24 words, and the resulting filtered sequence classifies the hand labeled transcription with an accuracy of 93.6% (percentage of correct classified words). The resulting symbols sequence is then translated into the same temporal scale of prosodic features and speaker turns. All these features are down-sampled to a common sampling frequency of 2Hz.

IV. Dynamic Bayesian Networks

Bayesian Networks (BNs) are directed acyclic graphical models. In a BN, nodes represent random variables, and arcs represents conditional dependencies. Thus an arc from node A to node B means B depends on A. An arc from C to B means that although B is also dependent on C, C and A are conditionally independent. Dynamic Bayesian Networks (DBNs) are the generalization of BNs to dynamic processes. Each temporal slice is represented by a BN, and oriented arcs, representing the time flow, connect variables of different time-slices. A large variety of statistical models, such as Hidden Markov Models (HMMs), Semi-Markov HMMs, factorial HMMs, etc. are unified under the graphical/mathematical formalism provided by DBNs [6].

A. Multi stream DBN model

Compared with a basic HMM, a DBN is able to factorize the internal hidden state using a set of connected variables. This principle is the basis of our model (figure 2a): the state space is decomposed in two levels of resolution: "meeting actions" (nodes A) and "meeting sub actions" (nodes S^F). The "meeting sub actions" space is further subdivided according to the nature of features that are processed. We have a "sub-state" node S^F for each feature class F (prosodic features, speaker turns, lexical features), thus independent feature streams are modeled independently. The joint distribution for a sequence of T temporal slices is:

$$P(A_{1:T}, S^1_{1:T}, S^2_{1:T}, S^3_{1:T}, Y^1_{1:T}, Y^2_{1:T}, Y^3_{1:T}) =$$
$$P(A_1) \cdot \prod_{F=1}^{3} \left\{ P(S^F_1 \mid A_1) \cdot P(Y^F_1 \mid S^F_1) \right\} \cdot \prod_{t=2}^{T} \{ P(A_t \mid A_{t-1}) \cdot$$
$$\cdot \prod_{F=1}^{3} \left\{ P(S^F_t \mid S^F_{t-1}, A_t) \cdot P(Y^F_t \mid S^F_t) \right\} \} \quad (1)$$

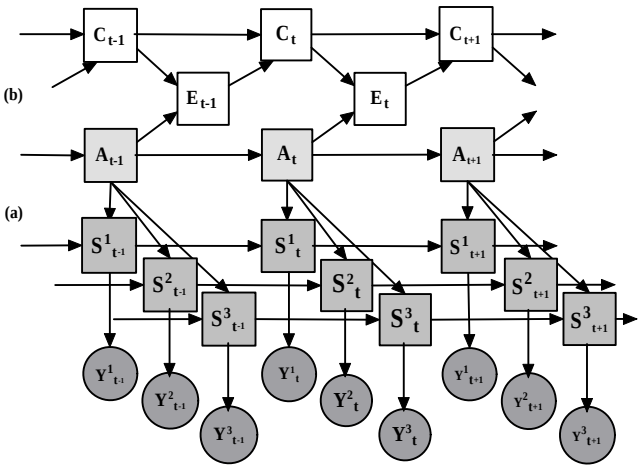

Fig. 2. Multi-stream DBN model (a) enhanced with the "Counter Structure" (b); square nodes represent discrete hidden variables and circles must be intend as continuous observations

Sub state nodes S^F,$F = [1, 3]$ follow a Markov chain with parameters determined by which meeting action it is in, which is encoded by the current state k of the the action variable A ($A_t = k$).

$$P(S_t^F = j \mid S_{t-1}^F = i, A_t = k) = \tilde{A}_k^F(i, j) \tag{2}$$

$\tilde{A}_k^F(i, j)$ is the transition matrix for the sub action variable S_t^F given that the parent action variable is in state k ($A_t = k$)

$$P(S_1^F = j \mid A_1 = k) = \tilde{\pi}_k^F(j) \tag{3}$$

is the initial sub state distribution for the stream F given an initial action $A_1 = k$.

Note that here, unlike in hierarchical HMM, there is no feedback from S^F to A, which prevents state transitions of A until S^F has not reached an "end state" [7].

The Markov chain associated with action nodes A acts like an ordinary HMM: having an action transition matrix $P(A_t = j \mid A_{t-1} = i) = A(i, j)$ and an initial state probability vector $P(A_1 = i) = \pi(i)$. Sub action nodes S^F , $F = 1, 2, 3$ are parents of the Markov chain A. Therefore instead of directly generating a sequence of observable discrete nodes Y through a standard state emission matrix

$$P(Y_t = k \mid A_t = i) = B(i, k) \tag{4}$$

A generates three hidden sub-action sequences S^1 , S^2 , S^3 through $\tilde{A}_k^1(i, j)$, $\tilde{A}_k^2(i, j)$ and $\tilde{A}_k^3(i, j)$ respectively.

Arcs between discrete "sub-states" S^F and continuous observation vectors Y^F, are implemented using mixtures of M_F Gaussians:

$$P(Y_t^F = y \mid S_t^F = i) = \sum_{m=1}^{M_F} C(F, m, i) N(y; \mu_{F,m,i}, \Sigma_{F,m,i}) \tag{5}$$

where $C(F, m, i)$ is the conditional prior weight of each mixture component for each stream F, and $N(y; \mu_{F,m,i}, \Sigma_{F,m,i})$ is the Gaussian density with mean $\mu_{F,m,i}$ and covariance $\Sigma_{F,m,i}$, evaluated at the point y. Note that sub-state cardinalities

$$\left|S^1\right| = 6 \ , \ \left|S^2\right| = 6 \ , \ \left|S^3\right| = 2 \tag{6}$$

are part of the model parameter set, and each sub-action is shared between different "meeting actions". The cardinality of A is equal to the dictionary number of actions: $|A| = 8$. This model presents many advantages over a model where features are "early integrated" into a single feature vector:

- feature classes are processed independently according to their nature
- more freedom is allowed in the state space partitioning and in the optimization of the sub-state space assigned to each feature class
- higher flexibility, for example when the feature set need to be modified
- knowledge from different streams is integrated together at an higher level of the model structure

Unfortunately all this advantages, and the better performances that can be achieved, are balanced by an increased model size, and therefore by an increased computational complexity.

B. Counter Structure

The probability to remain in an HMM state corresponds to an inverse exponential [8]: a similar behavior is displayed by the above model. Unfortunately "meeting actions" don't fit this assumption well, and the number of wrongly inserted actions tend to be high. In speech recognition this behaviour is often dealt with using an explicit duration model, or (more often) ad hoc solutions such as additional transition penalties. In this work, we have increased the flexibility of state duration modeling by adding an additional "counter structure" (figure 2b). The counter variable C, being ideally incremented during each action transition, attempts to model the expected number of recognized actions. Action variables A now also generate the hidden sequence of counter nodes C, together with the sequence of sub-action nodes S^j. Binary enabler variables E have an interface role between action variables A and counter nodes C. The joint distribution for the "counter structure" alone, computed over T time slices is:

$$\begin{aligned} &P(C_{1:T}, E_{1:T}, A_{1:T}) = P(C_1) \cdot P(E_1) \cdot P(A_1) \cdot \\ &\cdot \prod_{t=2}^{T} \{P(C_t \mid C_{t-1}, E_{t-1}) \cdot P(E_t \mid C_t, A_t) \cdot P(A_t \mid A_{t-1})\} \end{aligned} \tag{7}$$

with initial probabilities $P(C_1 = 0) = 1$ and $P(E_1 = 0) = 1$. The counter nodes C is iteratively incremented only if the enabler variable E was high ($E_{t-1} = 1$) during the previous temporal slice:

$$P(C_t = j \mid C_{t-1} = i, E_{t-1} = f) = \begin{cases} j = i + 1 & \text{if} \quad f = 1 \\ j = i & \text{if} \quad f = 0 \end{cases} \tag{8}$$

$$P(E_t = f \mid C_t = j, A_t = k) = D_{j,k}(f) \quad (9)$$

$D_{j,k}(f)$ represents the state transition probability for the enabler variable E_t given that the action variable is in state k and the counter in state j. If k is the j^{th} recognised "meeting action" the probability to "have E activated" (and start evaluating the $j+1^{th}$ "meeting action") is modelled by $D_{j,k}(f)$. The adoption of an enabler variable E has also the effect to reduce the dimension of conditional probability tables. Removing the enabler variable E and integrating (8) and (9) into a $P(C_t \mid C_{t-1}, A_{t-1})$, the number of parameters required by the "counter structure" will be increased by a factor:

$$\frac{|C|\,|A|}{2(|C| + |A|)} \quad (10)$$

The joint distribution of the multi stream model enhanced with a counter structure (figure 2a and 2b) can be easily obtained multiplying (1) by

$$P(C_1) \cdot P(E_1) \cdot \prod_{t=2}^{T} \{P(C_t \mid C_{t-1}, E_{t-1}) \cdot P(E_t \mid C_t, A_t)\}$$

V. Experiments

Our experiments were conducted on 30 fully transcribed meeting of the corpus described in section II, using the Graphical Models Toolkit (GMTK)[9]. The evaluation is performed using a leave-one-out procedure, in which the system was trained using 29 meetings and tested on the remaining one, iterating this procedure 30 times. For evaluation we used the Action Error Rate, a metric that privileges the recognition of the correct action sequence, rather than the precise temporal boundaries, obtained by summing the insertion, deletion and substitution errors when aligned against the reference sequence:

$$AER = \frac{Substitutions + Deletions + Insertions}{Correct\ number\ of\ actions}$$

Table I shows experimental results achieved using our base multi-stream approach and the counter enhanced variant. These models are tested using two features sets. Prosodic features and speaker turns are common to both the sets (A) and (B); and (B) contains lexical features too. Therefore to evaluate (A) we use a double-stream model with only sub-actions S^1 and S^2. To process the set (B) we have an additional Markov chain composed by sub-states S^3 and observable lexical features Y^3. The introduction of the lexical based feature, independently of the adoption of a "counter structure", improves the percentage of correct recognized actions by about 6 % and reduces AER by 5 %. The "counter structure" allows the number of insertions to be limited, enabling the model to better fit experimental data and to have a further small improvement in AER. Therefore we reached our best results (9 % AER) employing both the fully comprehensive feature set and the "counter structure".

Model	Corr.	Sub.	Del.	Ins.	AER
(A) multi-stream	84.6	9.0	6.4	1.3	16.7
(B) multi-stream	91.7	4.5	3.8	2.6	10.9
(A) multi-str. + counter	86.5	6.4	7.1	1.3	14.7
(B) multi-str. + counter	92.9	5.1	1.9	1.9	9.0

TABLE I

Action error rates (%) for the multi-stream model with and without the "counter structure" using: (A) prosody and speaker turns or (B) prosody, speaker turns and lexical feature.

VI. Conclusion

We have presented a framework for automatic segmentation of meetings into a sequence of phases. The audio information captured through individual lapel microphones has been exploited using a set of prosodic features. Location based speech activities evaluated through microphone array processing has been used to extract patterns from speaker turns. Lexical information embedded into textual transcriptions has been employed to build a monologue/dialogue discriminator. These three multi-modal features are then integrated through a specialized DBN model. Individual processing of different feature sets, and a mechanism to improve action duration modelling are two key points of our model. Experiments conducted on the IDIAP meeting corpus has shown that this infrastructure is capable of AER in the range from 15% to 9%. Therefore the DBN approach has proven to be an effective framework for the integration of features from different communicative modalities. Further multimodal-features will be integrated into this system, and a multi time scale version of the model will be soon investigated. Lexical based monologue/dialogue discrimination provided good results, therefore its natural extension with more than two actions will be soon integrated.

References

[1] R. Kazman, R. Al Halimi, W. Hunt, and M. Mantei, "Four paradigms for indexing video conferences," *IEEE Multimedia*, vol. 3, no. 1, 1996.

[2] I. McCowan, S. Bengio, D. Gatica-Perez, G. Lathoud, F. Monay, D. Moore, P. Wellner, and H. Bourlard, "Modelling human interaction in meetings," *Proc. IEEE ICASSP*, 2003.

[3] K. Sonmez, E. Shriberg, L. Heck, and M. Weintraub, "Modelling dynamic prosodic variation for speaker verifi cation," *Proc. ICSLP*, vol. 7, no. 920, pp. 3189–3192, 1998.

[4] N. Morgan and E. Fosler-Lussier, "Combining multiple estimators of speaking rate," *Proc. IEEE ICASSP*, pp. 729–732, 1998.

[5] A. Dielmann and S. Renals, "Dynamic bayesian networks for meeting structuring," *Proc. IEEE ICASSP*, 2004.

[6] P. Smyth, D. Heckerman, and M. I. Jordan, "Probabilistic independence networks for hidden Markov probability models," *Neural Computation*, vol. 9, no. 2, pp. 227–269, 1997.

[7] A. D. L. Xie, S.-F. Chang and H. Sun, "Unsupervised discovery of multilevel statistical video structures using hierarchical hidden markov models," *Proc. IEEE ICME, Baltimore*, 2003.

[8] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proc. of the IEEE*, vol. 2, no. 77, pp. 257–286, 1989.

[9] J. Bilmes, "Graphical models and automatic speech recognition," *Mathematical Foundations of Speech and Language Processing*, 2003.