SEMI-SUPERVISED MEETING EVENT RECOGNITION WITH ADAPTED HMMS

Dong Zhang, Daniel Gatica-Perez and Samy Bengio

IDIAP Research Institute
Swiss Federal Institute of Technology, Lausanne, Switzerland
{zhang, gatica, bengio}@idiap.ch

ABSTRACT

This paper investigates the use of unlabeled data to help labeled data for audio-visual event recognition in meetings. To deal with situations in which it is difficult to collect enough labeled data to capture event characteristics, but collecting a large amount of unlabeled data is easy, we present a semi-supervised framework using HMM adaptation techniques. Instead of directly training one model for each event, we first train a well-estimated general event model for all events using both labeled and unlabeled data, and then adapt the general model to each specific event model using its own labeled data. We illustrate the proposed approach with a set of eight audio-visual events defined in meetings. Experiments and comparison with the fully-supervised baseline method show the validity of the proposed semi-supervised approach.

1. INTRODUCTION

Audio-visual analysis enables us to recognize diverse events ranging from sports highlights to unusual events in surveillance. Recently, automatic meeting analysis has attracted interest from researchers in the fields of speech, vision and multimedia. Detecting and recognizing audio-visual events in meetings can be useful for meeting browsing and finding relevant segments of interest.

Current approaches to event recognition follow the supervised paradigm, in which event models, suiting the goals of a particular domain, are trained from labeled data, and then used for recognition on test data. Most existing work has used Hidden Markov Models (HMMs) [8] and extensions, including coupled HMMs, input-output HMMs, multi-stream HMMs, and asynchronous HMMs (see [6] for a recent review of models). Although the basic HMM, a discrete state-space model with an efficient learning algorithm, works well for temporally correlated sequential data, it is challenged by a large number of parameters, and runs the risk of over-fitting when learned from limited data [7]. In the case of meeting events recognition, this situation might occur since large vectors of audio-visual features from all meeting participants are concatenated to define the observation space [5]. This situation is aggravated by the labeling difficulties. Meeting event labeling is both laborious and time-consuming since meetings are often lengthy, and events in meetings are jointly defined by audio-visual patterns. The focus of this paper is to present a semi-supervised approach for event recognition in situations in which there is not enough labeled training data, and the high dimensionality of the observation space would require a large amount of labeled data to capture the event characteristics.

Our work is motivated by the fact that while obtaining sufficient labeled training data for audio-visual events is a difficult and time-consuming task, collecting a large amount of unlabeled data is usually easier. In this view, learning with both labeled and unlabeled data, referred to as semi-supervised learning, becomes a very attractive option [10]. In this paper, we propose a semi-supervised HMM framework for audio-visual event recognition, as an alternative to the fully supervised approach. Pooling labeled and unlabeled training data together, we first build a well-estimated general event model. Each specific event model is derived from the general event model using its own labeled training data via Bayesian adaptation. The proposed framework is general and can be easily applied to many cases in which collecting labeled data is difficult, but collecting a large amount of unlabeled data is easy. We apply our framework to a set of eight events defined based on multimodal turn-taking patterns in meetings, and illustrate its effectiveness compared with the supervised method. The rest of the paper is organized as follows. Section 2 introduces the proposed approach. Section 3 presents experiments and discussion. Concluding remarks are provided in Section 4.

2. SEMI-SUPERVISED FRAMEWORK

In this section, we first introduce our semi-supervised HMM framework. We then describe the implementation details and the set of eight meeting events we used.

2.1. Framework Overview

Our framework is based on Hidden Markov Models (HMMs) for temporal event modeling. Instead of training one HMM

This work was carried out in the framework of the Swiss NCCR (IM)2, and the European projects M4 and AMI.

for each event using the corresponding labeled data, we first train a well-estimated HMM, referred to as *general HMM* (G-HMM), using all labeled and unlabeled data for all events, according to Equation 1.

$$\theta^* = \arg\max_{\theta} \prod_{j=1}^{N} P(X_j|\theta), \tag{1}$$

where the set of parameters θ^* is learned by maximizing the likelihood of both labeled and unlabeled data $\{X_1, X_2, ..., X_N\}$ for all events. The probability density function of each HMM state is assumed to be a Gaussian Mixture Model (GMM). We use Expectation-Maximization (EM) algorithm [2] to train GMM parameters.

G-HMM can be viewed as a *general* event model since it is trained by pooling various events data (both labeled and unlabeled) together. Next, we adapt the parameters of this *general* model to derive models for each *specific* event using its own labeled samples, *i.e.* we move from the *general* event model to a *specific* event model using the corresponding labeled data and adaptation techniques (see section 2.2 for implementation details). In this way, we can overcome the lack of labeled data for each event for a good estimate of the model's parameters.

Given HMMs for all events, a meeting is modeled as the concatenation of single event HMMs. The corresponding sequence of events is obtained by applying the Viterbi decoding algorithm, a standard technique for segmentation and recognition with HMMs [8]. Given a sequence of audio-visual features extracted from a meeting, the Viterbi algorithm produces the sequence of states most likely to have generated the features. The state sequence corresponds to meeting events, so that the meeting events are segmented and recognized.

2.2. MAP Adaptation

Several adaptation techniques have been proposed for GMM-based HMMs, such as Gaussian clustering, Maximum Likelihood Linear Regression (MLLR) and Maximum a posteriori (MAP) adaptation (also known as Bayesian adaptation) [9]. These techniques have been widely used in tasks such as speaker and face verification [9, 3]. In these cases, a general world model of speakers / faces are trained and then adapted to the specific speaker / face model.

The parameters of a GMM-based HMM include *the number of Gaussian components, means, variances, mixture weights and state-transition probabilities*. When using MAP adaptation, different parameters can be chosen for adaptation [9]. In our case, the parameters adapted are *mean, variance, mixture weights*, while *the state-transition probabilities* are kept fixed and equal to their corresponding values in the general model. This is because the path chosen by the Viterbi algorithm is mostly influenced by the emission probabilities [1]. According to the MAP principle, we select parameters θ^* such that they maximize the posterior probability density, that is:

$$\theta^* = \arg\max_{\theta} P(\theta|X) = \arg\max_{\theta} P(X|\theta) \cdot P(\theta), \tag{2}$$

where $P(X|\theta)$ is the data likelihood and $P(\theta)$ is the prior distribution.

Following [9], there are two steps in adaptation. First, estimates of the statistics of the training data are computed for each component of the old model. We use $\{w_i^{new}, \mu_i^{new}, \sigma_i^{new}\}$ to represent the weight, mean and variance for component i in the new model, respectively. These parameters are estimated by Maximum Likelihood (ML), given by the well-known equations [2],

$$w_i^{new} = \frac{1}{M} \sum_{j=1}^{M} P(i|x_j, \theta), \tag{3}$$

$$\mu_i^{new} = \frac{\sum_{j=1}^{M} x_j P(i|x_j, \theta)}{\sum_{j=1}^{M} P(i|x_j, \theta)}, \tag{4}$$

$$\sigma_i^{new} = \frac{\sum_{j=1}^{M} P(i|x_j, \theta)(x_j - \mu_i^{new})(x_j - \mu_i^{new})^T}{\sum_{j=1}^{M} P(i|x_j, \theta)}, \tag{5}$$

where M is the number of data examples.

In the second step, the parameters of a mixture i are adapted using the following set of update equations [4].

$$\hat{w}_i = \alpha \cdot w_i^{old} + (1 - \alpha) \cdot w_i^{new}, \tag{6}$$

$$\hat{\mu}_i = \alpha \cdot \mu_i^{old} + (1 - \alpha) \cdot \mu_i^{new}, \tag{7}$$

$$\begin{aligned} \hat{\sigma}_i &= \alpha \cdot (\sigma_i^{old} + (\hat{\mu}_i - \mu_i^{old})(\hat{\mu}_i - \mu_i^{old})^T) \\ &+ (1 - \alpha) \cdot (\sigma_i^{new} + (\hat{\mu}_i - \mu_i^{new})(\hat{\mu}_i - \mu_i^{new})^T), \end{aligned} \tag{8}$$

where $\{\hat{w}_i, \hat{\mu}_i, \hat{\sigma}_i\}$ are the weight, mean and variance of the adapted model in component i, $\{w_i^{old}, \mu_i^{old}, \sigma_i^{old}\}$ are the corresponding weight, mean and variance in old component i, respectively, and α is a weighting factor to control the balance between old model and new estimates. The smaller the value of α, the more contribution the new data makes to the adapted model. We will investigate the effect of α on the performance in Section 3.

2.3. Audio-visual Events in Meetings

As an implementation of the proposed framework, we use the set of events first defined in [5] (see Table 1). We model a meeting (assumed to have four participants) as a sequence of exclusive events taken from the set of 8 events: {discussion, monologue1, monologue2, monologue3, monologue4, note-taking, presentation, white-board}. Note that we differentiate monologue events by different participants, *i.e*, monologue1 is a monologue by meeting participant 1, *etc.* Given the audio-visual feature sequence extracted from a meeting, our goal is to segment and recognize the event sequence $E = \{E_1, E_2, ...\}$, where E_i belongs to one of the eight meeting events in Table 1.

Table 1. Description of meeting events

Events	Description
Discussion	most participants engaged in conversations
Monologue	one participant speaking continuously without interruption
Note-taking	most participants taking notes
Presentation	one participant presenting and using the projector screen
White-board	one participant speaking and using the white-board

Fig. 1. Multi-camera meeting room

3. EXPERIMENTS AND RESULTS

In this section, we describe the experiments. First, we describe the meeting corpus and the audio-visual features we extracted. We then present our performance measures and experimental setup. Finally, we present results and discuss our findings.

3.1. Meeting Corpus

The meeting corpus we used consisted of 59 five-minute, four-participant meetings [5], collected in a meeting room equipped with cameras and microphones[1]. A snapshot of the meeting room is shown in Figure 1. There are three cameras in the meeting room. Two cameras capture a frontal view of the meeting participants, and the third camera captures the white-board and the projector screen. Audio was recorded using lapel microphones attached to participants, and an eight-microphone array in the center of the table.

3.2. Feature Extraction

We extracted a set of standard audio-visual features [5]. Visual features were extracted from the three cameras. For the two cameras looking at people, visual features extracted consist of head vertical centroid position and eccentricity, hand horizontal centroid position, eccentricity, and angle. The motion magnitude for head and hand blobs were also extracted. Average intensity of difference images computed by background subtraction, were extracted from the third camera. For Audio features, from microphone array signals, we first computer a speech activity measure (SRP-PHAT). Three acoustic features, namely energy, pitch and speaking rate, were estimated on speech segments, zeroing silence segments. We used the SIFT algorithm to extract pitch, and a combination of estimators to extract speaking rate [5].

[1] http://mmm.idiap.ch/

Table 2. Number of frames in different data sets (NA: *Not Applicable*, because the supervised method does not use unlabeled data in the training process; [400,10000] means from 400 to 10000.).

method	train		valid-ation	test
	labeled	Unlabeled		
supervised	[400,10000]	NA	4552	43400
semi-supervised		30048		

All visual and audio features were extracted at 5 frames per second, and then concatenated.

3.3. Measures and Experimental Setup

We use the *action error rate* (AER) to evaluate our results. AER is equivalent to the *word error rate* (WER) widely used in continuous speech recognition, and is defined as the sum of *insertion* (Ins), *deletion* (Del), and *substitution* (Subs) errors, divided by the total number of events in the ground-truth: AER $= \frac{\text{Subs+Del+Ins}}{\text{total events}} \times 100\%$

We then compare the proposed semi-supervised approach with supervised HMMs. For the supervised method, one HMM for each event is directly trained using its own labeled data. For testing, the Viterbi algorithm [8] is applied to segment and recognize meeting events.

The meeting corpus is divided into training (27 meetings), validation (3 meetings) and testing (29 meetings) sets. Each meeting in the training data set was randomly assigned to either the labeled or the unlabeled set. The number of frames in different data sets is summarized in Table 2. In order to investigate performance with respect to the size of the labeled data, the size of labeled training data is progressively increased. Starting from 400 labeled frames, which correspond to 50 frames for each of the eight possible events, the number of labeled frames increases with a step size of 400. Therefore, we get trained (adapted) models over 400, 800, ..., 10000 labeled frames respectively. The general model in the semi-supervised method were trained using all training data (both labeled and unlabeled).

The model parameters (the number of HMM states, and the number of Gaussians per mixture) are determined using a validation data set, randomly generated from 30 training meetings. The parameter space, *i.e.* the number of states and number of Gaussians, ranges between 1 and 10.

3.4. Results and Discussion

We first study the effects of adapting different parameter combinations given the fixed α value of 0.5. We investigate three parameter combinations, namely mean, mean+weight, mean+weight+variance. As shown in Figure 2(a), the best performance was obtained by adapting *mean* while adapting *mean+weight+variance* gave the worse performance. This might be explained by the fact that adapting more parameters would require more labeled data.

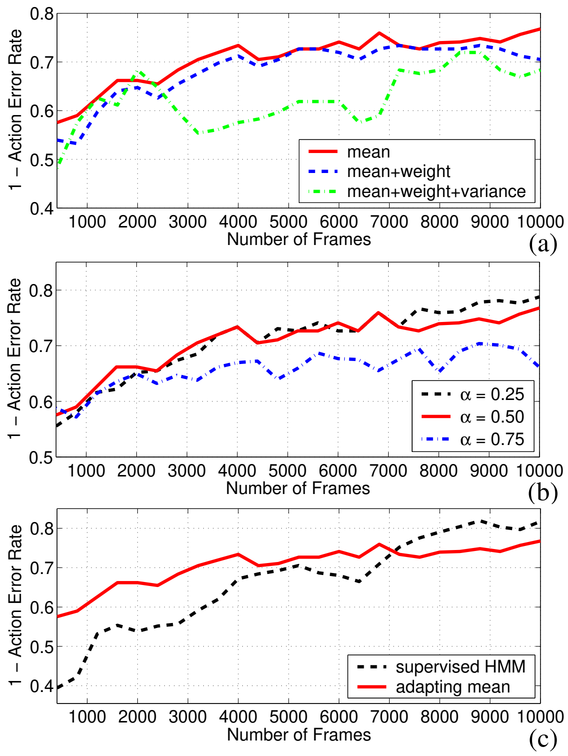

Fig. 2. (a) Results of the semi-supervised method for adapting different paramter combinations. (b) Results of the semi-supervised method for different α values. (c) Comparison of the semi-supervised and the supervised methods. The x-axis represents the number of labeled frames used in model training / adaptation (features were extracted at 5 frames per second).

We then investigate effects of α on the performance by adapting mean. From Section 2.2, we know α represents the contribution of the general model to the adapted model. In 2(b), we can see that the performances of different α were very similar with the small number of labeled frames (less than 2000), but $\alpha = 0.25$ gave better performance when more labeled frames were used.

The comparison between the semi-supervised approach (adapting mean, $\alpha = 0.25$) and the baseline supervised method are shown in Figure 2(c). We can see that the performances of both the semi-supervised HMM and supervised HMM increases as additional labeled samples are used in training (adaptation). The performance of the supervised HMM increases faster than that of the semi-supervised HMM. When the number of the labeled samples ranges from 400 to 7000, the performance of the semi-supervised HMM is better than that of the supervised HMM. The improvement in relative performance in this frame range (from 400 to 7000) fluctuates between around 3% to 46%.

With more than 7000 labeled samples used, the supervised HMM began to perform better than the semi-supervised approach. In other words, at least 7000 labeled samples are needed for the supervised HMM to perform better than the semi-supervised approach. This shows that the best performance can be achieved by training directly over enough labeled training samples while the benefit of using semi-supervised HMM is to achieve better performance when there are little (insufficient) labeled data. Motivated by this observation, we suggest designing a hybrid system: using a validation set, one can decide when to switch to the supervised HMM from semi-supervised HMM.

Finally, note that although the performance using a full training data set (30 meetings) can be as high as 90% [5] in terms of $(1 - AER)$, the amount of labeled data required is around six times the amount we used in these experiments. This clearly highlights the tradeoff between the performance and the cost of collecting / labeling training data.

4. CONCLUSION

We presented a semi-supervised framework for audio-visual event recognition using HMM adaptation techniques. Instead of directly training one model for each event, we first train a well-estimated general model for all events using both labeled and unlabeled events, and then adapts the general event model to each specific event model using its own labeled data. We illustrate the proposed approach with a set of eight audio-visual events commonly found in meetings. Experiments and comparison with the supervised HMM method show that our method could be a good alternative to the supervised method, especially for little data, and could be worth investigating in other meeting events.

5. REFERENCES

[1] Y. Bengio. Markovian Models for Sequential Data. Neural Computing Surveys 2:129–162, 1999.

[2] J. Bilmes. A gentle tutorial of the EM algirthm and its application to parameter estimation for gaussian mixture and hidden markov models. ICSI-TR-97-021 U.C. Berkeley, 1997.

[3] F. Cardinaux, C. Sanderson, and S. Bengio. Adapted generative models for face verification. *IEEE, Face and Gesture*, 2004

[4] J. L. Gauvain and C.-H. Lee. Maximum a posteriori estimation for multivariate gaussian mixture observation of markov chains. In *IEEE Transactions on Speech Audio Processing*, April 1994.

[5] I. McCowan, S. Bengio, D. Gatica-Perez, G. Lathoud, F. Monay, D. Moore, P. Wellner, and H. Bourlard. Modeling human interactions in meetings. In *Proc. IEEE ICASSP*, Hong Kong, April 2003.

[6] K. Murphy. Dynamic bayesian networks: Representation, inference and learning. *Ph.D. dissertation, UC Berkeley*, 2002.

[7] N. Oliver, E. Horvitz, and A. Garg. Layered representations for learning and inferring office activity from multiple sensory channels. In *Proc. ICMI*, Pittsburgh, Oct. 2002.

[8] L. R. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall, 1993.

[9] Douglas A. Reynolds, Thomas F. Quatieri, and Robert B. Dunn. Speaker verification using adapted gaussian mixture models. *Digital Signal Processing*, 10(1-3), 2000.

[10] M. Seeger. Learning with labeled and unlabeled data. *Technical Report*, 2000