IDIAP RESEARCH REPORT

Role Recognition in Multiparty Recordings using Social Affiliation Networks and Discrete Distributions

Sarah Favre [a b] Hugues Salamin [a b]
Alessandro Vinciarelli [a b]

IDIAP–RR 08-64

July 2008

to appear in
Proceedings of ICMI International Conference on Multimodal Interfaces (2008)

[a] Ecole Polytechnique Federale de Lausanne - 1015 Lausanne, Switzerland
[b] Idiap Research Institute - CP592, 1920 Martigny, Switzerland

IDIAP Research Institute **www.idiap.ch**
Av. des Prés–Beudin 20 P.O. Box 592 1920 Martigny – Switzerland
Tel: +41 27 721 77 11 Fax: +41 27 721 77 12 Email: info@idiap.ch

IDIAP Research Report 08-64

Role Recognition in Multiparty Recordings using Social Affiliation Networks and Discrete Distributions

Sarah Favre Hugues Salamin Alessandro Vinciarelli

July 2008

to appear in
Proceedings of ICMI International Conference on Multimodal Interfaces (2008)

Abstract. This paper presents an approach for the recognition of roles in multiparty recordings. The approach includes two major stages: extraction of Social Affiliation Networks (speaker diarization and representation of people in terms of their social interactions), and role recognition (application of discrete probability distributions to map people into roles). The experiments are performed over several corpora, including broadcast data and meeting recordings, for a total of roughly 90 hours of material. The results are satisfactory for the broadcast data (around 80 percent of the data time correctly labeled in terms of role), while they still must be improved in the case of the meeting recordings (around 45 percent of the data time correctly labeled). In both cases, the approach outperforms significantly chance.

1 Introduction

One of the main tenets of sociology is that people involved in social interactions play roles: "*People do not interact with one another as anonymous beings. They come together in the context of specific environments and with specific purposes. Their interactions involve behaviors associated with defined statuses and particular roles. These statuses and roles help to pattern our social interactions and provide predictability*" [12]. In this work, we address the problem of recognizing automatically the *role* of people in radio programs and meetings.

The approach we propose is composed of two main stages (see Figure 1): the first is the extraction of feature vectors accounting for relationships between people, the second is the mapping of the feature vectors into categories corresponding to the roles. The *feature extraction stage* (left dotted box in Figure 1) starts by splitting the data into single speaker segments. The speaker sequence is then used to extract a Social Affiliation Network [14] and to model the intervention time distribution associated to each role. The *recognition stage* (right dotted box in Figure 1) maps the feature vectors into classes corresponding to the different roles. This task is performed using either Bernoulli or Multinomial distributions [5]. Moreover, the fraction of time each role accounts for in a given recording is modeled with Gaussian distributions.

To the best of our knowledge, only a few works have been dedicated to the automatic recognition of roles. Some of them recognize *functional* roles in broadcast data [4][13], i.e. the tasks that different people perform in television and radio programs (e.g. *anchorman* or *guest*), and another recognizes *functional* roles in movies [15] (e.g. *hero* or *hero's friends*). The recognition is based on lexical features like the n-gram distribution in [4], and on Social Network Analysis [14] in [13][15]. Other works recognize the *social roles* of meeting participants [17] (e.g. *attacker* or *supporter*) using features like the overall amount of movement and speech energy, or the roles corresponding to specific actions [3] (e.g. *presentation* and *briefings*) using the total speaking time of each person and turn-taking statistics.

We performed experiments over three different corpora (see Section 4.1 for more details): a collection of radio news bulletins (around 20 hours), a dataset of radio talk-shows (around 25 hours), and the AMI meeting corpus (around 45 hours) [11]. For the first two datasets, the accuracy, percentage of time correctly labeled in terms of role, is close to 80%, while it is around 45% for the meeting data. One probable reason is that the interactions are more constrained in the case of the broadcast data and this leads to more stable patterns associated to the different roles. However, the performance of the system is significantly higher than chance for both kinds of data and several roles are recognized with high accuracy.

Role recognition can be useful in several applications: browsers can be enhanced by enabling users to select interventions corresponding to a given role, retrieval systems can use the role as a clue for filtering the results, summarization systems can use the role as a criterion for the selection of information rich data segments, etc.

The rest of the paper is organized as follows: Section 2 presents the interaction pattern extraction, Section 3 describes the role assignment technique, Section 4 presents experiments and results, and Section 5 draws some conclusions.

2 Feature Extraction

This section presents the technique used to extract and represent the interaction pattern of each person. The technique includes two steps: the first is the segmentation of the recordings into single speaker segments (speaker diarization), the second is the extraction of an Affiliation Network from the resulting speaker sequence (see left dotted box in Figure 1).

In our experiments, we considered two kinds of data: broadcast material where there is a single audio channel and meeting recordings where each participant wears a headset microphone. This requires the application of different speaker diarization techniques: in the first case (single audio channel), an unsupervised speaker diarization technique identifies the voices of the different people

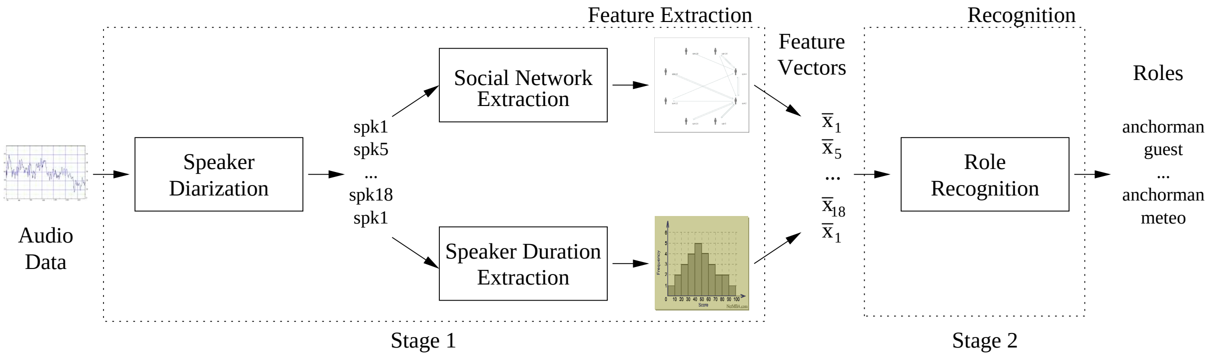

Figure 1: Role recognition approach. The picture show the two main stages of the approach: the feature extraction and the actual role recognition.

(see Section 2.1). In the second case (headset microphones), the diarization splits the channel of each microphone into speech and non-speech segments (see Section 2.2). Section 2.3 shows how the output of the speaker diarization is used to build an Affiliation Network and represent people with vectors accounting for their social relationships.

2.1 Speaker Diarization for Single Audio Channel Data

A full description of the speaker diarization technique used for the single audio channel data (broadcast material) is given in [1][2]. The algorithm is based on a fully connected continuous density Hidden Markov Model (HMM) where each state corresponds to a cluster of observation vectors and, in principle, to a single speaker voice. The emission probability is modeled with a Gaussian Mixture Model (GMM) [5]. Each observation vector has 12 dimensions corresponding to the *Mel Frequency Cepstral Coefficients* (MFCC) extracted every 10 ms from a 30 ms long window [9]. The MFCC are used because they have been shown to be more effective than other features in speaker recognition tasks, thus they seem to capture the different voices and their characteristics [1].

The first step of the process is the initialization of the above HMM. The audio data is segmented into M uniform non-overlapping segments, where M is the initial number of states in the HMM. Since the number of speakers is not known a-priori, an initial guess must be provided for M. This parameter is set to a value much higher than the expected number of speakers because the algorithm achieves good results only when starting from an oversegmentation. The resulting HMM is trained using the uniform segmentation as groundtruth and the result is a parameter set $\Theta^{(0)}$. The resulting HMM can be aligned with the data using the Viterbi algorithm to find the best sequence of states (i.e. speakers):

$$q^{(0)} = \arg\max_{q \in \mathcal{Q}} \mathrm{p}(q \mid O, \Theta^{(0)}) \tag{1}$$

where q is a sequence of states, $\mathcal{Q}$ is the set of all possible sequences of states, and $O = \{\vec{o}_1, \ldots, \vec{o}_K\}$ is the sequence of the observation vectors. The alignment results into a segmentation different from the uniform one used for the initialization. The HMM can thus be retrained and a new parameter set $\Theta^{(1)}$ is obtained:

$$\Theta^{(1)} = \arg\max_{\Theta} \mathrm{p}(q^{(0)} \mid O, \Theta) \tag{2}$$

where $\Theta = \{\theta_1, \ldots, \theta_M\}$, i.e. the parameter set of the HMM, can be thought of as a set of GMM parameters, if the transition probabilities and the intial state probabilities are kept uniform.

Since the number M is higher than the actual number of speakers, the data is oversegmented and there are clusters that should be merged since they contain data belonging to the same speaker. For

Step	Parameter	Setting	Step	Parameter	Setting
Training	Training examples	> 22M	Inference	Minimum duration	20 states
	Feature sampling rate	100 Hz		Insertion penalty	-40
	Feature dimensionality	54		Silence/speech prior	0.8/0.2
	Input layer	810 (54×15) units		Silence collar	100 ms
	Hidden layer	25 arctan units		Silence merge	250 ms

Table 1: Summary of parameters in the training and inference steps in the automatic speech segmentation system.

this reason, the two most similar states (in terms of the GMM parameters) are merged when the following condition is met:

$$\log \mathrm{p}(O_{m+n} \,|\, \theta_{m+n}) \geq \log \mathrm{p}(O_m \,|\, \theta_m) + \log \mathrm{p}(O_n \,|\, \theta_n) \tag{3}$$

where O_m, O_n and O_{m+n} are the observation vectors attributed to cluster m, n and their union respectively, θ_m and θ_n are the parameters of GMMs in states m and n and θ_{m+n} are the parameters of a GMM trained with Expectation-Maximization on O_{m+n}.

After the merging, the HMM has fewer states and it can be realigned with the data in order to obtain a new segmentation which can be used to train again the HMM. The new states satisfying the above condition will be thus merged again and the whole procedure will be iterated. The merging between states is performed by keeping constant the number of parameters:

$$|\theta_{m+n}| = |\theta_m| + |\theta_n|, \tag{4}$$

the above condition is achieved by setting the number of Gaussians in the state resulting from the merging to the sum of the numbers of Gaussians in the merged states. In this way, the likelihood will increase until the states that are merged actually correspond to the same or similar voices and will decrease when the states that are merged correspond to voices too different. This provides the stopping criterion for the iteration process. In fact, the alignment and training steps are repeated until the likelihood reaches its maximum. The segmentation corresponding to the maximum likelihood is retained as the result of the speaker diarization process.

2.2 Speech/ Non-Speech Segmentation for Headset Microphone Data

The approach for the segmentation of the headset microphone channels employs a *Multilayer Perceptron* (MLP) for estimating the posterior probability of audio frames as speech or non-speech classes [6]. Input to the classifier uses standard speech recognition features combined with features specifically designed for the detection of cross-talk in headset microphone recordings, as this has been found to be a major source of segmentation errors in such meeting room data [16]. The input features are summarised as follows: 13 *Mel filterbank perceptual linear predictive coefficients* (MF-PLP) including $C0$, plus normalised log-energy; *Log cross-channel normalised energy* which is estimated as the logarithm of the energy of the current headset microphone minus the logarithm of the sum of energies across all headset microphones for the current meeting; *Signal kurtosis*, which should be large during single speaker activity (since speech signals tend to be super-Gaussian) and approach zero during silence; *Mean cross-channel correlation* and *Maximum cross-channel correlation*, where, for a given time frame, we take the maximum cross-correlation values between the current headset microphone channel and all other headset microphones and obtain the *mean* measure as the arithmetic mean of these cross-correlation values and the *maximum* measure as the maximum cross-correlation value. In practice, we concatenate the first and second order differences of these features thus giving a feature dimensionality of 54. Finally, we take several consecutive frames and provide these as input to the MLP.

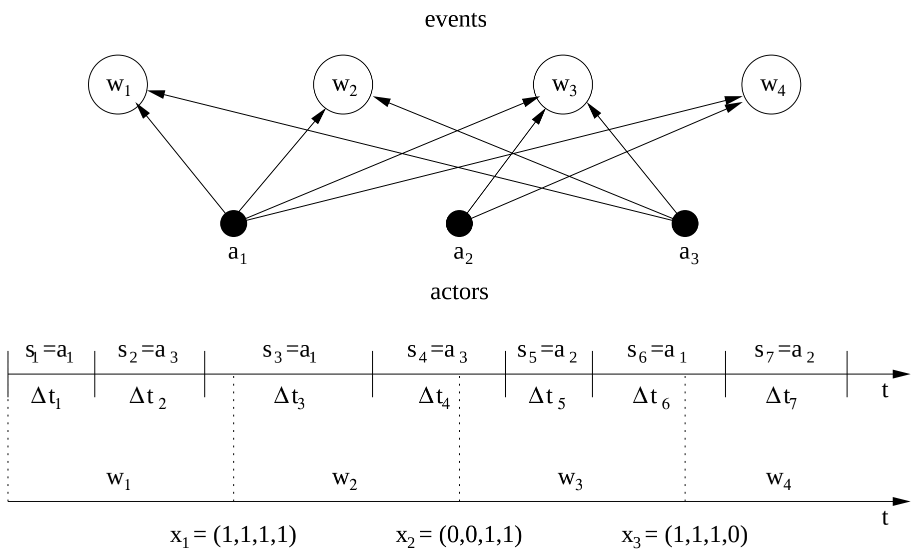

Figure 2: Interaction pattern extraction. The picture shows the Affiliation Network extracted from a speaker segmentation. The events of the network correspond to the windows w_j and the actors are linked to the events when they talk during the corresponding windows. The actors are represented using vectors $\vec{y}_i$ where the components account for the links between actors and events.

The classifier is trained using meeting room data from several non-AMI meeting room corpora (specifically, ICSI, NIST, CMU corpora, see [6] for further details). Error back-propagation was used to train the MLP parameters, with a separate validation data set being used to control overfitting of the parameters. A frame error of 3.9% was measured on this validation set. The ground truth segmentation for training of the classifier was generated using forced alignment of manual transcriptions of the corpora using acoustic models from our meeting transcription system [8]. We have previously observed that such semi-automatic ground truth segmentations are more reliable than the original manual corpus segmentations since manual segmentations have a tendency to have too-coarse granularity and are inconsistent between different corpora.

The segmentation is carried out using hidden Markov models (HMM) for speech and non-speech classes with minimum duration and insertion penalty constraints to ensure that the segmentation is consistent with that observed for the ground truth. Emission likelihoods for the HMM states are estimated as scaled likelihoods in which MLP posterior probabilities are divided by their respective prior class probability. A final smoothing step is applied by padding speech segments by an additional amount and merging resulting speech segments which have a silence gap less than a predefined duration. The tuning of the various system parameters has been carried out to maximise performance for meeting room automatic speech recognition performance. Table 1 summarises the main parameters in the training and inference steps.

The system described above was run on the entire AMI corpus to provide automatic segmentation of the audio data for subsequent processing for speaker role analysis.

2.3 Affiliation Network Extraction

The result of the speaker diarization process is that each recording is split into a sequence $S = \{(s_i, \Delta t_i)\}$, where $i = 1, \ldots, |S|$, s_i is the label assigned to the speaker voice detected in the i^{th} segment of audio, and Δt_i is the duration of the i^{th} segment. The label s_i belongs to the set $A = \{a_1, \ldots, a_G\}$ of unique speaker labels, output by the speaker diarization process (see lower part of Figure 2). The sequences extracted from the speaker diarization are used to create a Social Network representing the relationships between the speakers, more specifically an *Affiliation Network*. An Affiliation Network is a graph with two kinds of nodes: the *actors* and the *events* [14]. Actors can be linked to events, but no links are allowed between nodes of the same kind (see upper part of Figure 2). In our experiments, the actors correspond to the speakers in the broadcast news and in the meetings and the events correspond to uniform non-overlapping windows spanning the whole length of the recordings. The reason is that this network is expected to capture the relationships between the speakers and one of the most reliable evidences of interaction is the proximity in time [7]. In fact, two persons talking during the same window are more likely to interact with each other than two people talking in different windows.

One of the main advantages of this representation is that each actor a_i can be represented with a vector $\vec{x}_i$ where the component j accounts for the participation of a_i in the j^{th} event. In our experiments, we used two kinds of representation: in the first one, the j^{th} component is 1 if the speaker talks during the j^{th} window and 0 otherwise (the corresponding vectors are shown at the bottom of Figure 2). In the second the j^{th} component is the number of times that speaker a_i talks during the j^{th} window. In the first case the vectors are binary, in the second case they have integer components higher or equal to 0. In both cases, people that interact more with each other tend to talk during the same windows and are represented by similar vectors. The choice of the number of windows used to segment the recordings as well as the length of the windows used in this work, are justified in Section 4.

3 Role Recognition

This section describes the statistical foundations of the role assignment process used in our experiments. Section 2 has shown that the relationship pattern of each speaker i can be represented with a vector $\vec{x}_i = (x_{i1}, \ldots, x_{iD})$, where D is the number of windows, that can have either binary or positive integer components. Furthermore, every speaker i talks during a fraction τ_i of the total time of a bulletin. We can thus represent every speaker by a vector $\vec{y}_i = (\tau_i, \vec{x}_i)$.

Consider the vector $\vec{r} = (r_1, \ldots, r_G)$, where r_i is the role of speaker i, and the vector of observation $Y = \{\vec{y}_1, \ldots, \vec{y}_G\}$, where $\vec{y}_i$ is the vector representing speaker i. The problem of assigning the role to all speakers can be thought of as the maximization of the *a-posteriori* probability $\mathrm{p}(\vec{r} \,|\, Y)$. By applying the Bayes Theorem and by taking into account that $p(Y)$ is constant during the recognition this problem is equivalent to finding $\hat{\vec{r}}$ such that:

$$\hat{\vec{r}} = \arg\max_{\vec{r} \in \mathcal{R}^G} \mathrm{p}(Y \,|\, \vec{r})\, \mathrm{p}(\vec{r}), \tag{5}$$

where $\mathcal{R}$ is the set of the predefined roles. In order to simplify the problem, we make the assumption that the observations are mutually conditionally independent given the roles. In the case we are considering, it seems also reasonable to assume that the observation $\vec{y}_i$ of speaker i only depends on its role r_i and not on the role of the other speakers. Equation (5) can thus be rewritten as:

$$\hat{\vec{r}} = \arg\max_{\vec{r} \in \mathcal{R}^G} \mathrm{p}(\vec{r}) \prod_{k=1}^{G} \mathrm{p}(\vec{y}_k \,|\, r_k). \tag{6}$$

In order to further simplify the problem, we assume that $\vec{x}_i$ and τ_i are statistically independent given the role, thus:

$$\mathrm{p}(y_i \,|\, r_i) = \mathrm{p}(\vec{x}_i \,|\, r_i)\, \mathrm{p}(\tau_i \,|\, r_i). \tag{7}$$

In this work, we only considered the most simple model for $\mathrm{p}(\vec{r})$. This model assumes that the roles are independent and thus that $\mathrm{p}(\vec{r})$ is simply the product of the a-priori probabilities of the roles. In this model, Equation (5) boils down to:

$$\vec{\hat{r}} = \arg\max_{\vec{r}\in\mathcal{R}^G} \prod_{k=1}^{G} \mathrm{p}(r_k)\,\mathrm{p}(\vec{x}_k \,|\, r_k)\,\mathrm{p}(\tau_k \,|\, r_k) \tag{8}$$

The main advantage of this approach is that the observations became independent and the maximization of the product can be achieved by maximizing separately each factor $\mathrm{p}(\vec{y}_k \,|\, r_k)\,\mathrm{p}(r_k)$.

In the next subsections, we will show how we estimate $\mathrm{p}(\vec{x} \,|\, r)$, $\mathrm{p}(\tau \,|\, r)$ and $\mathrm{p}(r)$

3.1 Modeling Binary Interaction Patterns

This subsection shows how we model the interaction patterns extracted from the Affiliation Networks when the components of the vectors $\vec{x}_i$ are binary, i.e. $x_{ij} = 1$ when actor a_i talks during window j and 0 otherwise. Given a labeled training set, there are N_r speakers playing the role r. Each one of them is represented by a binary vector $\vec{x}$. The most natural way of modeling binary vectors is to use Bernoulli discrete distributions:

$$\mathrm{p}(\vec{x}_i \,|\, \vec{\mu}_r) = \prod_{j=1}^{D} \mu_{rj}^{x_{ij}} (1-\mu_{rj})^{1-x_{ij}}, \tag{9}$$

where D is the number of events in the network (see Section 2), and $\vec{\mu}_r = (\mu_{r1}, \dots, \mu_{rD})$ is the parameter vector of the distribution related to role r. The maximum likelihood estimates of the μ_{ri} parameters are as follows [5]:

$$\mu_{ri} = \frac{1}{N_r} \sum_{n=1}^{N_r} x_{ni}, \tag{10}$$

where N_r is the number of people playing the role r in the training set, and x_{nj} is the j^{th} component of the vector representing the n^{th} person playing the role r. A different Bernoulli distribution is trained for each role.

3.2 Modeling Multinomial Interaction Patterns

This subsection details the model we use for the Affiliation Networks when the components x_{ij} correspond to the number of times that actor a_i talks during window j, i.e. the components are integers greater or equal to 0. Given a vector $\vec{x} = (x_1, \dots, x_D)$, where D is the number of windows, each component x_i can be represented with a vector $\vec{z}_i$ defined as follows:

$$\vec{z}_i = (z_{i1}, \dots, z_{iT}), \tag{11}$$

where T is the maximum number of times that an actor can talk during a given window, $z_{ij} \in \{0, 1\}$, and $\sum_{j=1}^{T} z_{ij} = 1$. In other words, x_i is represented with a T-dimensional vector where all the components are 0 except one, i.e. the component $z_{in} = 1$, where n is the number of times that the actor represented by $\vec{x}$ talks during event i. As a result, $\vec{x}$ is represented with a concatenation of vectors $\vec{z} = (\vec{z}_1, \dots, \vec{z}_D)$. The vector $\vec{z}$ can thus be modeled with a multinomial distribution:

$$\mathrm{p}(\vec{z} \,|\, \vec{\mu}) = \prod_{i=1}^{D} \prod_{j=1}^{T} \mu_{ij}^{z_{ij}}, \tag{12}$$

The parameters $\vec{\mu}$ can be estimated by maximizing the likelihood over a training set $\mathcal{X}$. This leads to a closed form expression for the parameters:

$$\mu_{ij} = \frac{1}{N_r} \sum_{l=1}^{N_r} z_{ij}^{(l)}, \tag{13}$$

where N_r is the number of vectors corresponding to role r.

3.3 Modeling Durations

This section shows how we estimate the probabilities $\mathrm{p}(\tau \mid r)$. Given a labeled training set, there is a number N_r of speakers playing role r. Each one of them accounts for a fraction τ_n of the bulletin he or she is involved in, where $n = 1, \ldots, N_r$. We model $\mathrm{p}(\tau \mid r)$ using a Gaussian Distribution $\mathcal{N}(\tau \mid \mu_r, \sigma_r)$, where μ_r and σ_r are mean and variance respectively. The Maximum Likelihood estimates of the parameters are the sample mean:

$$\mu_r = \frac{1}{N_r} \sum_{n=1}^{N_r} \tau_n \tag{14}$$

and the sample variance:

$$\sigma_r = \frac{1}{N_r} \sum_{n=1}^{N_r} (\tau_n - \mu_r)^2. \tag{15}$$

A different Gaussian distribution is obtained for each role.

3.4 Estimating Role Probabilities

This section shows how we estimate the probability $p(r)$ of a given role being observed. Given a labeled training set, the total number of people is denoted by N, and the number of people playing role r is N_r. Then, we have the following estimation of the a-priori probability:

$$\mathrm{p}(r) = \frac{N_r}{N}, \tag{16}$$

i.e. the fraction of individuals in the training set labeled with the role r.

4 Experiments and Results

This section presents experiments and results obtained in this work. The next three sections describe data and roles, the performance measures and the role recognition results.

4.1 Data and Roles

The experiments of this work have been performed over three different corpora. The first, referred to as C1 in the following, contains 96 news bulletins with an average length of 11 minutes and 50 seconds. The corpus contains all news bulletins broadcasted by *Radio Suisse Romande*, the French speaking Swiss National broadcasting service, during February 2005 and can thus be considered a representative sample of these kinds of programs. The second corpus, referred to as C2 in the following, contains 27 one hour long talk-shows broadcasted by *Radio Suisse Romande* (see above) during February 2005. Also in this case, the corpus can be considered a representative sample of this specific kind of program. The third corpus, referred to as C3 in the following, is the AMI corpus [10], a collection of 138 meeting recordings for a total of 45 hours and 38 minutes of material. The meetings are simulated and are based on a scenario where the participants are the members of a team working on the development of a new remote control.

The set of the predefined roles is the same for C1 and C2: the *Anchorman* (AM), i.e. the person managing the program, the *Second Anchorman* (SA), i.e. the person supporting the AM, the *Guest* (GT), i.e. the person invited to report about a single and specific issue, the *Interview Participant* (IP), i.e. interviewees and interviewers, the *Abstract* (AB), i.e. the person reading a short abstract at the beginning of the program, and the *Meteo* (MT), i.e. the person reading the weather forecasts. In C3, the set of the roles is different and contains the *Project Manager* (PM), the *Marketing Expert* (ME), the *User Interface Expert* (UI), and the *Industrial Designer* (ID).

Corpus	AM	SA	GT	IP	AB	MT
C1	41.2%	5.5%	34.8%	4.0%	7.1%	6.3%
C2	17.3%	10.3%	64.9%	0.0%	4.0%	1.7%

Table 2: Role distribution. The table reports the percentage of data time each role accounts for in C1 and C2.

Corpus	PM	ME	UI	ID
C3	36.6%	22.1%	19.8%	21.5%

Table 3: Role distribution. The table reports the percentage of data time each role accounts for in C3.

Table 2 shows the distribution of the data time across the roles in C1 and C2. The fraction of data time each role accounts for in C3 is reported in Table 3. Roles and distributions are significantly different in C1, C2 and C3 and this enables us to test the robustness of our approach with respect to changes in the data.

4.2 Speaker Diarization Results

The relationship patterns used at the role assignment step are extracted from the speaker segmentation obtained with the diarization process. Errors in the diarization (e.g. people detected as speaking when they are silent) lead to spurious interactions that can mislead the role assignment process.

The effectiveness of the diarization is measured with the *Purity* π, a metric showing on one hand to what extent all feature vectors corresponding to a given speaker are detected as belonging to the same voice, and on the other hand to what extent all vectors detected as a single voice actually correspond to a single speaker. The Purity ranges between 0 and 1 (the higher the better) and it is the geometric mean of two terms: the *average cluster purity* π_c and the *average speaker purity* π_s. The definition of π_c is as follows:

$$\pi_c = \sum_{k=1}^{N_c} \sum_{l=1}^{N_s} \frac{n_k}{N} \frac{n_{lk}^2}{n_k^2}, \tag{17}$$

where N_s is the number of speakers, N_c is the number of voices detected in the diarization process, n_{lk} is the number of vectors belonging to speaker l that have been attributed to voice k, n_k is the number of feature vectors in voice k and N is the total number of feature vectors. The definition of π_s is as follows:

$$\pi_s = \sum_{l=1}^{N_s} \sum_{k=1}^{N_c} \frac{n_l}{N} \frac{n_{lk}^2}{n_l^2} \tag{18}$$

(see above for the meaning of the symbols).

The application of the speaker diarization process in the case of radio programs requires the setting of the initial number of states M in the fully connected Hidden Markov Model (see Section 2). The value of M must be significantly higher than the number of expected speakers for the diarization process to work correctly. In our experiments, we set *a-priori* $M = 30$ for C1 and $M = 90$ for C2. No other values have been tested. The average purity is 0.81 for C1 and 0.79 for C2. The average purity for C3 is 0.99. The difference in purity is explained by the different methods used to obtain the speaker segmentation.

	all	AM	SA	GT	IP	AB	MT
B	81.2	97.8	4.0	92.6	6.0	47.8	74.9
M	81.0	97.8	2.7	92.6	2.1	47.8	75.6

Table 4: Role recognition performance for C1. The table reports the role recognition results for the corpus C1. The results show both the overall accuracy and the accuracy for each role. The "B" stands for *Bernoulli*, and the "M" stands for *Multinomial*.

	all	AM	SA	GT	IP	AB	MT
B	83.9	75.0	88.3	90.6	0.0	54.5	13.3
M	82.0	62.3	92.3	86.7	0.0	98.4	22.8

Table 5: Role recognition performance for C2. The table reports the role recognition results for the corpus C2. The results show both the overall accuracy and the accuracy for each role. The "B" stands for *Bernoulli*, and the "M" stands for *Multinomial*.

4.3 Role Recognition Results

For our experiment, we used the *accuracy* α as the performance measure. The accuracy is defined as the percentage of data time correctly labeled in terms of role. We used a *leave-one-out* approach [5] to train our models and select the number D of windows used to split the recordings (see Section 2). This means that each recording in a corpus is used iteratively as test set, while the others are used as training set. In this way, the whole corpus can be used as test set while still preserving the rigorous separation between training and test data necessary to assess realistically the performance of the role recognition system.

Tables 4, 5 and 6 report the role recognition results for corpora C1, C2 and C3 respectively. The distribution used to model the interaction patterns is indicated with B (Bernoulli) and M (multinomial). The overall α is above 82 percent for both C1 and C2 and this means that the role recognition approach is robust with respect to changes in the time distribution across the roles. This is important because the same role is played in different ways depending on the specific program and the approach seems to be capable of adapting automatically to the different situations.

The 20 percent of mislabeled data time is due to two main sources of error: the first is the delay of the diarization process in correspondence of speaker changes. On average, the speaker changes in the output of the diarization process are delayed by around 2 seconds with respect to the actual speaker changes. The average number of changes in C1 is 30 and this results into roughly 60 seconds of mislabeling (around 10 percent of the average C1 recording length). Similar figures can be found for C2 where roughly 10 percent of the time again is mislabeled because of the delays between actual and detected speaker changes. The performance of the system when using the ground-truth speaker segmentations rather than the output of the speaker diarization is 95.3 percent for C1 and 96.5 for C2. This seems to confirm that around 10 percent of the error is actually due to the above phenomenon (the results have been obtained using a single Bernoulli distribution).

The second major source of error is the classification of IP, MT and AB into GT. Such roles have similar interaction patterns, but the higher a-priori probability of the GT bias the recognition toward the latter. Fortunately, the IP, MT and AB do not account for a large fraction of the data time and the impact on the overall performance is small.

In the C3 corpus, the overall α is around 43 percent. The results show that the role recognition approach presented in this paper is less effective for a more spontaneous database with small groups such as the AMI meeting corpus. The relationship features are not stable, and thus the models are not able to classify correctly the participants into the four different roles. The *Project Manager* is the only role that is correctly captured. Its interaction pattern is distinct of the other roles (with a high

	all	PM	ME	UI	ID
B	43.6	79.4	19.5	33.0	13.0
M	42.8	76.4	14.4	30.2	22.5

Table 6: Role recognition performance for C3. The table reports the role recognition results for the corpus C3. The results show both the overall accuracy and the accuracy for each role. The "B" stands for *Bernoulli*, and the "M" stands for *Multinomial*.

activity throughout the meeting). This difference is well captured by our approach and the PM is labeled with an accuracy close to 80%. The three other roles, i.e. ME, UI, ID, have similar interaction patterns, thus our approach does not achieve a good accuracy. The accuracy over the groundtruth speaker segmentation is 49.5% (achieved with a Bernoulli distribution). Like in the case of C1 and C2, the relative loss when passing to the automatic speaker segmentation is around 12%. The main reason the approach is less effective over C3 is probably that the number of meeting participants is too small (only 4 per recording) to build meaningful Affilitation Networks [14].

5 Conclusions

In this paper, we have presented an approach for the automatic recognition of people's roles in multiparty recordings. The approach is based on Social Network Analysis and it has been applied over different kinds of data to assess its robustness and its limits. The results show that the approach is effective for the broadcast data where the interactions between people are sufficiently constrained, while still requires improvements for the meetings where the interactions are more spontaneous.

The approach uses only the audio channel even if the AMI corpus includes videos captured with three different cameras and synchronized with the audio. One one hand, this is an advantage because it allows the application of the approach to data like the radio programs where only the audio is available. On the other hand, it is a disadvantage because for data where the video is available important information is probably missed if the visual aspects are not taken into account. For this reason, the future work will focus on the inclusion of visual features in the meetings data to build a multi-modal approach (see [11][17] for examples of techniques based on multiple modalities). Moreover, in the experiments of this work the roles are considered statistically independent, while they are dependent and they must respect several constraints, e.g. the news bulletins must have only one anchorman. These constraints can be included in the recognition stage to improve the performance.

References

[1] J. Ajmera. *Robust Audio Segmentation.* PhD thesis, École Polytechnique Fédérale de Lausanne (EPFL), 2004.

[2] J. Ajmera and C. Wooters. A robust speaker clustering algorithm. In *Proceedings of IEEE Workshop on Automatic Speech Recognition Understanding*, 2003.

[3] S. Banerjee and A. Rudnicky. Using simple speech based features to detect the state of a meeting and the roles of the meeting participants. In *proceedings of International Conference on Spoken Language Processing*, 2004.

[4] R. Barzilay, M. Collins, J. Hirschberg, and S. Whittaker. The rules behind the roles: identifying speaker roles in radio broadcasts. In *Proceedings of American Association of Artificial Intelligence Symposium*, 2000.

[5] C. Bishop. *Pattern Recognition and Machine Learning.* Springer Verlag, 2006.

[6] J. Dines, J. Vepa, and T. Hain. The segmentation of multi-channel meeting recordings for automatic speech recognition. In *Proceedings of Interspeech*, pages 1213–1216, 2006.

[7] E. Glaeser and J. Scheinkman. Measuring social interactions. In S. Durlauf and H. Young, editors, *Social Dynamics*, pages 83–132. MIT Press, 2001.

[8] T. Hain, L. Burget, J. Dines, G. Garau, V. Wan, M. Karafiát, J. Vepa, and M. Lincoln. The AMI system for the transcription of speech in meetings. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 357–360, 2007.

[9] X. Huang, A. Acero, and H.-W. Hon. *Spoken Language Processing: A guide to theory, algorithm and system development*. Prentice Hall, 2001.

[10] I. McCowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, W. Post, D. Reidsma, and P. Wellner. The AMI meeting corpus. In *Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research*, 2005.

[11] I. McCowan, D. Gatica-Perez, S. Bengio, G. Lathoud, M. Barnard, and D. Zhang. Automatic analysis of multimodal group actions in meetings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):305–317, 2005.

[12] H. Tischler. *Introduction to Sociology*. Harcourt Brace College Publishers, 1990.

[13] A. Vinciarelli. Speakers role recognition in multiparty audio recordings using social network analysis and duration distribution modeling. *IEEE Transactions on Multimedia*, 9(6), 2007.

[14] S. Wasserman and K. Faust. *Social Network Analysis*. Cambridge University Press, 1994.

[15] C. Weng, W. Chu, and J. Wu. Movie analysis based on roles social network. In *proceedings of IEEE International Conference on Multimedia and Expo*, pages 1403–1406, 2007.

[16] S. Wrigley, G. Brown, V. Wan, and S. Renals. Speech and crosstalk detection in multichannel audio. *IEEE Transactions on Speech and Audio Processing*, 13(1):84–91, 2005.

[17] M. Zancanaro, B. Lepri, and F. Pianesi. Automatic detection of group functional roles in face to face interactions. In *proceedings of International Conference on Mutlimodal Interfaces*, pages 47–54, 2006.

Acknowledgments

This work is supported by the Swiss National Science Foundation through the National Center of Competence in Research on Interactive Multimodal Information Management (IM2). The authors wish to thank John Dines for his help.