INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Automatic recognition of social roles using long term role transitions in small group interactions

Gaurav Fotedar[1,⋆], *Aditya Gaonkar P*[1,⋆], *Saikat Chatterjee*[2,†], *Prasanta Kumar Ghosh*[1,‡]

[1]Department of Electrical Engg., Indian Institute of Science (IISc), Bangalore 560012, Karnataka, India
[2]School of Electrical Engineering,
Royal Institute of Technology (KTH), Stockholm, Sweden
⋆{gfotedar,adityapgaonkar}@gmail.com, †sach@kth.se, ‡prasantg@ee.iisc.ernet.in,

Abstract

Recognition of social roles in small group interactions is challenging because of the presence of disfluency in speech, frequent overlaps between speakers, short speaker turns and the need for reliable data annotation. In this work, we consider the problem of recognizing four roles, namely Gatekeeper, Protagonist, Neutral, and Supporter in small group interactions in AMI corpus. In general, Gatekeeper and Protagonist roles occur less frequently compared to Neutral, and Supporter. In this work, we exploit role transitions across segments in a meeting by incorporating role transition probabilities and formulating the role recognition as a decoding problem over the sequence of segments in an interaction. Experiments are performed in a five fold cross validation setup using acoustic, lexical and structural features with precision, recall and F-score as the performance metrics. The results reveal that precision averaged across all folds and different feature combinations improves in the case of Gatekeeper and Protagonist by 13.64% and 12.75% when the role transition information is used which in turn improves the F-score for Gatekeeper by 6.58% while the F-scores for the rest of the roles do not change significantly.

Index Terms: social computing, social roles, dynamic programming, small group interaction

1. Introduction

Meetings are ubiquitous in structuring daily work in organizations, recalling important pieces of information (decisions, keypoints, milestones etc) and sharing these with people absent from those meetings. Advances in multimedia compression and digital storage technologies have resulted in several archives of meeting interactions. As the size of multimedia archives grows, it becomes very challenging. Among others, role information can help organize and index multimedia content from audio recordings of meetings. Banerjee et al [1] showed that meeting browsers annotated with speaker roles and topic segments were very effective for answering user queries. Role information can also be used for topic segmentation in conversation discourses [2] and summarization of spoken documents [3]. Roles in a meeting are of two types formal and social (or informal). Bales work [4] shows that social roles characterize the relationships between members in the meeting and are useful to capture the dynamics of the meeting. Typically, social roles answer semantic queries like, Who is doing what in an event?. Role Theory [5] observes that people behave in predictable ways based on their social roles. This shows that knowing the social role of a person can help determine his/her interactions with the environment and vice-versa. Also, the knowledge of social roles can help determine engagement, social dominance and hotspots [6] in meetings. Thus, automatic recognition of social roles is useful for a number of multimedia applications.

Typically social roles in a small group meeting could be [4]: *Gatekeeper* - a group moderator, *Neutral* - a passive participant, *Protagonist* - the driver of the conversation, *Supporter* - participant with cooperative attitude and *Attacker* - participant expressing disagreement. Every participant, in general, has different social role in each slice of a meeting. Computational models for recognizing roles could be useful for classifying the role of a participant in every slice of the meeting. However, there are several issues which make computational modeling of roles in a meeting challenging. For example, presence of disfluency [7] in speech and frequent overlaps between speakers increase the errors of automatic speech recognition (ASR) and speaker segmentation systems which are typically used for extracting features required for role recognition. Short speaker turns [7] decrease the amount of data available per speaker in a meeting slice thereby making feature extraction difficult. Also, limited availability of well annotated meeting corpora pose a challenge for obtaining a good-quality computational model for role recognition.

There have been several attempts at automatic recognition of social and formal roles in a meeting. Banerjee et al [8] annotated Carnegie Mellon multi-modal corpus [9] with simple participant role labels (presenter, information provider, participator and information consumer) and used a decision tree classifier with features including number of speaker changes, overlapped speech duration, to predict roles. Favre et al. [10, 11] used hidden Markov model (HMM) and n-gram based sequential probabilistic models along with social network analysis to predict formal roles in three different corpora. Laskowski et al. [12] used behavior model as defined in [13] to predict roles in the AMI corpus using features such as probability of interruption, holding the floor and backchannel behavior. However, roles in these works are imposed by the context of the meeting, e.g. Project Manager, Weather man. This makes it difficult to generalize the learned model to predict roles in other meetings where the context/scenario might be different. Similar to predicting formal roles, there have been several works to predict social roles in group interactions. For example, Sapru et al. [14] have addressed both social and formal role recognition using acoustic, lexical and structural features and multi-class boosting. Zancanaro et al. [15] experimented with the Mission Survival corpus [16] and used speech activity and body fidgeting features with a support vector machine (SVM) based classifier. Valente et al. [17] used prosodic and turn-taking features combined with influence of speakers on one another using data from the AMI corpus [18]. Wilson et al. [19] used combinations of speech activity, subjectivity and ex-

http://dx.doi.org/10.21437/Interspeech.2016-202

pressive prosodic features along with conditional random field (CRF) to determine roles in the AMI corpus meetings. However, these approaches have used relatively small datasets and a limited set of features. Sapru et al. [20, 21, 22, 7] annotated a large portion of the publicly available AMI corpus [18] with social role labels and trained their role recognition model using various combinations of linguistic, structural and acoustic features.

Typically a participant can change social role over time but his/her role will not change frequently within a short time window and, thus, in a meeting slice, a participant has a single social role. The turn-taking statistics across participants along with the dependency between social and formal roles have been used by Vinciarelli et al. [23] for recognition of socio-emotional roles in AMI corpus. Similarly, an influence model framework [24] has been proposed for capturing the influence of the social roles on the task based roles and vice-versa, where the feature vector has been modeled using HMM with states representing the roles. In this work, instead of having a generative model for the feature vector (such as HMM) we have formulated the role recognition problem as a decoding task where the participant specific role transition is used and the likelihood of the feature vector for a given role is generated by a discriminative classifier, hidden conditional random field (HCRF). The sequence of predicted roles is estimated by using a dynamic programming based approach. The proposed role recognition assumes the availability of all slices from a meeting for every participant.

Experiments on AMI corpus [18] show that precision averaged across all feature combinations and folds improves for Gatekeeper and Protagonist roles by 13.64% and 12.75% respectively when the proposed role recognition method is used compared to recognizing role independently in each meeting slice. Furthermore, this leads to an improvement of 6.58% in the F-score for Gatekeeper while making no changes in the F-scores for other roles.

2. Database

The AMI meeting corpus is a publicly available dataset containing over 100 hours of scenario and non-scenario meetings. Both audio and video recordings of the meetings have been carried out in specially equipped meeting rooms. Each scenario meeting consists of four participants who are tasked with designing a remote control. Each participant plays a role of either Project Manager, Industrial Designer, User Interface Designer or Marketing Expert. Out of all the scenario meetings, 59 meetings have social role annotation available [7]. Each meeting has been segmented into meeting slices of average duration less than 30 seconds based on the presence of pauses longer than 1 second [21]. Each speaker in each meeting slice has been assigned one role from among Protagonist, Gatekeeper, Supporter, Neutral and Attacker. These make up a total of 1714 meeting slices corresponding to ∼12.5 hours of meeting data and 6856 social role annotations. The number of role annotations corresponding to Gatekeeper, Neutral, Protagonist, Supporter and Attacker are 934, 3352, 629, 1923 and 18 respectively. Due to limited availability of data for the Attacker role, we have removed data pertaining to Attackers for the experiments in this work and consider classification among remaining 4 role classes.

3. Recognition using long term role transitions

Let $\mathcal{S}_k$ be the k-th ($1 \leq k \leq K$) slice of a group interaction, where K is the total number of slices. Let $\Upsilon = \{\rho_1, \rho_2, \rho_3, \rho_4\}$ be the set of four different roles considered in this work, where ρ_1, ρ_2, ρ_3, ρ_4 represent Gatekeeper, Neutral, Protagonist, and Supporter respectively.

Typically, role recognition in a meeting slice is posed as a classification problem using features from the respective slice. Let f_k be the feature vector for the k-th slice and $\mathcal{L}_k$ denote the role in the k-th slice for a participant in the meeting. A classifier is used to obtain the probabilities of four roles given the feature vector in the k-th slice for a participant. In particular, four probabilities $P_r^k = \text{Prob}(\mathcal{L}_k = \rho_r | f_k)$, $r = 1, 2, 3, 4$ are computed by the classifier and the role with the highest probability becomes the predicted role, $\hat{\mathcal{L}}_k$, for a participant in the k-th slice as: $\hat{\mathcal{L}}_k = \rho_{r^\star}$, where $r^\star = \arg\max_r P_r^k$.

Assuming all roles are equally likely, P_r^k can be rewritten as

$$P_r^k = \text{Prob}(\mathcal{L}_k = \rho_r | f_k) \propto p(f_k | \mathcal{L}_k = \rho_r) \tag{1}$$

where $p(f_k|\mathcal{L}_k)$ is the likelihood of the feature vector given the role $\mathcal{L}_k$.

However in a group interaction, the role of a participant could vary across slices. This could depend on the dynamics of the conversation in the meeting, the characteristics of the participant, and the response from other participants. Capturing these long term dynamics in the role sequence for a participant could help in predicting the role more accurately. In particular, the prediction of the role in the k-th slice can use the information of the roles in all the slices before k-th slice.

In this work, we consider maximizing the joint probability of the roles of a participant in all slices in a group interaction instead of maximizing the probability in each slice independently. In other words, we consider $\text{Prob}(\mathcal{L}_1, \mathcal{L}_2, \cdots, \mathcal{L}_K | f_1, f_2, \cdots, f_K)$, where $\mathcal{L}_k \in \Upsilon$, $\forall k$. Using the definition of conditional probability,

$$\begin{gathered}\text{Prob}(\mathcal{L}_1, \mathcal{L}_2, \cdots, \mathcal{L}_K | f_1, f_2, \cdots, f_K) \propto \\ p(f_1, f_2, \cdots, f_K | \mathcal{L}_1, \mathcal{L}_2, \cdots, \mathcal{L}_K)\text{Prob}(\mathcal{L}_1, \mathcal{L}_2, \cdots, \mathcal{L}_K)\end{gathered} \tag{2}$$

The first term in equation (2) is obtained from the classifier following equation (1). We assume that given the roles in all K slices the feature vectors in these slices are independent. Thus we obtain

$$p(f_1, f_2, \cdots, f_K | \mathcal{L}_1, \mathcal{L}_2, \cdots, \mathcal{L}_K) = \prod_{k=1}^{K} p(f_k | \mathcal{L}_k) \tag{3}$$

which can be further obtained from equation (1). While $p(f_k|\mathcal{L}_k)$ is often modelled using GMM, for example in a HMM framework [24], we propose to obtain $p(f_k|\mathcal{L}_k)$ using equation (1) from a discriminative classifier trained for role classification task. This is because estimation of GMM parameters in high dimensional feature space could lead to unstable solutions [25] with small number of training slices for some roles, such as Gatekeeper and Protagonist. On the other hand, the second term in equation (2) captures the long-term role dynamics of a participant in a meeting. We assume that the role sequence is a first-order Markov chain, i.e., the roles in the k-th meeting slice are conditionally independent of the roles of the $(k-2)$-th slice and the ones before that given the role of the $(k-1)$-th slice. Then, using the chain rule of probability, we can write $\text{Prob}(\mathcal{L}_1, \cdots, \mathcal{L}_K)$

$$\begin{aligned} &= \text{Prob}(\mathcal{L}_K | \mathcal{L}_1, \cdots, \mathcal{L}_{K-1})\text{Prob}(\mathcal{L}_1, \cdots, \mathcal{L}_{K-1}) \\ &= \text{Prob}(\mathcal{L}_K | \mathcal{L}_{K-1})\text{Prob}(\mathcal{L}_1, \cdots, \mathcal{L}_{K-1}) = \cdots \\ &= \text{Prob}(\mathcal{L}_1) \prod_{k=2}^{K} \text{Prob}(\mathcal{L}_k | \mathcal{L}_{k-1}) \propto \prod_{k=2}^{K} \text{Prob}(\mathcal{L}_k | \mathcal{L}_{k-1}) \\ &\qquad \text{[assuming roles are equally likely]} \end{aligned} \tag{4}$$

Using equations (3), and (4), we can see that the first term in equation (2) is determined by the classifier while the second term is determined by the role transition probabilities Prob$(\mathcal{L}_k|\mathcal{L}_{k-1})$. We use weights (1-$\gamma$) and γ on these two terms, where γ ($0 \leq \gamma \leq 1$) controls the contribution of the role transition probabilities in computing the overall probability of the role sequence.

$$\text{Prob}(\mathcal{L}_1, \mathcal{L}_2, \cdots, \mathcal{L}_K|f_1, f_2, \cdots, f_K) \propto \left(\prod_{k=1}^{K} p(f_k|\mathcal{L}_k)\right)^{1-\gamma} \left(\prod_{k=2}^{K} \text{Prob}(\mathcal{L}_k|\mathcal{L}_{k-1})\right)^{\gamma} \quad (5)$$

The estimated sequence of roles is obtained by maximizing the probability in equation (5) as follows:

$$\hat{\mathcal{L}}_k, \forall k = \arg \max_{\mathcal{L}_1, \cdots, \mathcal{L}_K} \text{Prob}(\mathcal{L}_1, \cdots, \mathcal{L}_K|f_1, f_2, \cdots, f_K) \quad (6)$$

Since there are four roles, a full search for solving equation (6) would have a complexity of $O(4^K)$. This is computationally prohibitive since a typical value of the total number of meeting slices (K) is in the range of 23-34. In order to circumvent this problem, we implement a dynamic programming (DP) based algorithm to find the solution for the optimization in equation (6). The DP based solution has a complexity of $O(16K)$. For the DP based solution, we define $D_r(k)$ as the maximum probability of assigning k many roles for first k slices with ρ_r as the role in the k-th slice of the meeting. Let the back-tracking pointer in DP be denoted by $\xi_r(k)$ which stores the role assigned to the $(k-1)$-th slice for obtaining the maximum probability $D_r(k)$. $D_r(k)$ is computed in a recursive manner and $\xi_r(k)$ is stored in each recursion of the DP as follows:

1. *Initialization:* Compute $D_r(1) = \left(P_r^1\right)^{(1-\gamma)}$ using equation (1).
2. *Iteration:* For $2 \leq k \leq K$ and $1 \leq r \leq 4$, compute the following:

   $$D_r(k) = \max_{1 \leq r' \leq 4} \{D_{r'}(k-1) \times (\alpha_{r,r'})^{\gamma}\} \times \left(P_r^k\right)^{(1-\gamma)}$$
   $$\xi_r(k) = \arg \max_{1 \leq r' \leq 4} \{D_{r'}(k-1) \times (\alpha_{r,r'})^{\gamma}\}$$

   where P_r^k is obtained using equation (1) and $\alpha_{r,r'} =$ Prob$(\rho_r|\rho_{r'})$. The role transition probability is obtained using the training data, which is also used to train the classifier from which P_r^k is computed.
3. *Backtracking:* $\hat{\mathcal{L}}_K = \arg\max_r D_r(K)$.

   $$\hat{\mathcal{L}}_k = \xi_{\hat{\mathcal{L}}_{k+1}}(k+1), \quad k = K-1, K-2, \cdots, 1 \quad (7)$$

4. Experiments and Results

4.1. Features

We extract verbal features from the audio and speech transcripts of the AMI meeting corpus to capture behaviors of the participants in each slice of the meeting.

Acoustic features: The speaking style and vocal expressions of participants in a meeting can give hints about their role. Following the work by Sapru et al. [7], we extract various statistical and regression functionals such as average, standard deviation (SD), skewness, kurtosis, range, maximum, minimum, linear and quadratic regression coefficients and approximation errors of different low level descriptor (LLD) contours and their derivatives from the audio of each meeting slice. LLDs include zero-crossing rate, sub-band energy, spectral roll-off, spectral flux, harmonicity, Mel frequency cepstral coefficients (MFCCs), short-time energy, pitch, voicing probability, jitter, shimmer, logarithm of harmonics to noise ratio. This results in a 448 dimensional acoustic feature vector.

Lexical features: The words used by participants in a meeting convey information about their roles and functions in the meeting. We use Linguistic Inquiry and Word Count (LIWC) [26] to analyze the speech transcripts of the meeting slices and for each transcript we obtain features as the weights of various linguistic categories resulting in a 59 dimensional lexical feature vector.

Structural features: The duration over which a participant is active as well as the number of speaking turns in a meeting slice could convey significant information about his/her role. The speech transcripts of a meeting slice provide the timestamps of the utterances of words, from which we extract the fraction of speaking time and the number of speaking turns taken by the participant in that slice as a two-dimensional structural feature vector.

4.2. Experimental Setup

We perform the role classification experiments in this work in a five fold cross validation setup. For this purpose, we divide the 59 meetings randomly into 5 sets - 4 sets with 12 meetings each and the remaining set with 11 meetings. In each fold, three sets are used for training our model, one for development and the remaining set is used for testing.

Various combinations of features have been used for role recognition task, namely Acoustic only (A), Lexical only (L), Structural only (S), Acoustic + Lexical (AL), Acoustic + Structural (AS), Lexical + Structural (LS), Acoustic + Lexical + Structural (ALS). Following the work by Sapru et al [7], we use hidden conditional random field (HCRF)[1] as the classifier to obtain P_r^k (equation (1)) for the proposed role recognition method. Three hidden states are used with 500 function evaluations to train the HCRF. For training, we standardize each feature dimension to zero-mean and unit-variance. In order to compute the transition probabilities (equation 4) we use a normalized count on the role sequence for every participant in the training set.

We use three metrics for evaluating the performance of the proposed method, namely precision, recall (accuracy) and F-score [7]. All metrics have been reported for each role averaged across five folds. The recall (accuracy) is also reported averaged over all roles.

The parameter γ in the proposed role recognition method is optimized on the development set. This is done using a grid search approach in which γ is chosen from 0 to 1 with a step of 0.1 and the γ that provides the highest accuracy on the development set is chosen for performing role recognition in the test set.

As a baseline method, we consider the most recent work by Sapru et al [7] on automatic recognition of emergent social roles where each meeting slice is classified independently into one of the four role classes using HCRF classifier.

4.3. Results and discussion

Figure 1 summarizes the results obtained for the baseline and proposed methods for all combinations of the feature sets. Table 1 gives the averages of the three performance metrics across all feature combinations for both baseline and proposed methods. It is evident from the table that in most of the cases, there is an

[1] We use a python implementation freely available at https://github.com/dirko/pyhcrf.

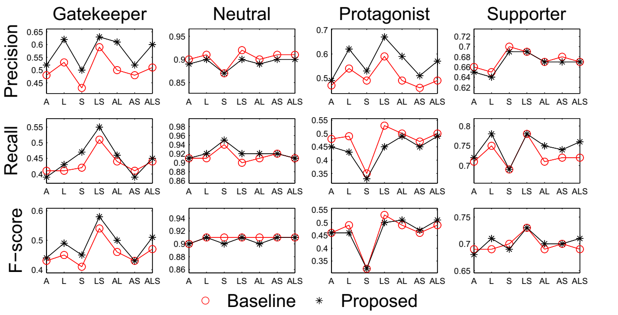

Figure 1: Various performance metrics for different feature combinations for all roles.

improvement in the performance metrics by using the proposed method (indicated by bold entries). Interestingly, precision improves for the classes with less training data, i.e., Gatekeeper and Protagonist, highlighting an advantage of the proposed method.

Figure 1 provides insights into the types of feature combinations that perform well in social role prediction. We see that, in terms of F-score, the two-dimensional structural features perform as well as the higher dimensional acoustic or lexical features, for both baseline and the proposed methods in case of Neutral role. This could be because when the participant is playing a Neutral role in a meeting slice, he/she remains silent for most of the time, the essence of which is captured by the structural features. We can also observe that in the case of Gatekeeper role, a combination of lexical and structural features performs the best in terms of F-score compared to the remaining six feature combinations using both the baseline and the proposed method. This is also true for the Protagonist role when using the baseline method.

	Method	Gatekeeper	Neutral	Protagonist	Supporter
Precision	Baseline	0.50	0.90	0.50	0.67
	Proposed	**0.57**	0.89	**0.57**	0.67
Recall	Baseline	0.43	0.91	0.47	0.73
	Proposed	**0.44**	**0.92**	0.44	**0.75**
F-score	Baseline	0.46	0.91	0.46	0.69
	Proposed	**0.49**	0.91	0.46	**0.70**

Table 1: Performance metrics averaged across all feature combinations for all roles. The recall (accuracy) averaged across all roles turns out to be 0.75 and 0.76 using the baseline and the proposed methods respectively.

It is interesting to observe that for the Gatekeeper role, the proposed method outperforms the baseline method in terms of the average F-score irrespective of the feature combination used. When averaged across different feature combinations, the F-score is found to improve by 6.58% over the baseline method. It is also interesting to see that for both Gatekeeper and Protagonist roles, the proposed method outperforms the baseline in terms of precision averaged across five folds by 13.64% and 12.75% respectively. This could be due to the inclusion of role transition probability in the proposed method. Unlike classification in each meeting slice independently, the information of the role transition from one slice to another could reduce the false positives resulting in better precision.

Table 2 shows the role transition probability matrix averaged across five folds in our experiments. While we see that the probability of retaining the same role in consecutive meeting slices is high (diagonal entries in the matrix), there are several large non-diagonal entries too indicating large probability of transition from one role to a different one. For example, the probability of transition from a Gatekeeper to a Neutral or Supporter is much higher than that to a Protagonist and vice versa. Similarly, there is a significant probability of transition from a Neutral to a Supporter and vice versa. This could be because a person in Neutral role in the current slice of the meeting could take up a Supporter role to establish his/her supporting point to the agenda or remain Neutral as the meeting continues. However, it is unlikely for a participant to transit from a Neutral role to the role of Protagonist. Also it would be very unusual for the Gatekeeper, who sets the agenda of the meeting to become the Protagonist in the following slice of the meeting, which is supported by the low probability value corresponding to the transition from the Gatekeeper to the Protagonist role.

To From	Gatekeeper	Neutral	Protagonist	Supporter
Gatekeeper	0.70	0.16	0.04	0.10
Neutral	0.02	0.72	0.03	0.23
Protagonist	0.10	0.14	0.62	0.14
Supporter	0.06	0.35	0.07	0.52

Table 2: Role transition probabilities averaged across five folds.

	Feature Combinations						
Fold	A	L	S	LS	AL	AS	ALS
1	0.2	0.3	0.1	0.4	0.6	0.4	0.4
2	0.1	0.2	0.1	0.4	0.5	0.3	0.5
3	0.0	0.3	0.6	0.6	0.4	0.4	0.4
4	0.5	0.3	0.0	0.4	0.4	0.4	0.4
5	0.4	0.2	0.0	0.1	0.4	0.3	0.5
Avg	0.24	0.26	0.16	0.38	0.46	0.36	0.44

Table 3: Optimal γ values for different folds and feature combinations

The optimal choices of γ for the five folds in our experiments are shown in Table 3. The last row in the table shows the value of γ averaged across five folds. The feature combinations in the decreasing order of γ are AL, ALS, LS, AS, L, A, S. It is interesting to observe that for AL and ALS the proposed method improves the F-score in the case of three roles over the baseline method as seen in Fig. 1. Incidentally the contribution of the role transition probability in role recognition (reflected by the γ value) is also high in these two cases compared to other feature combinations. For structural features (S), the proposed method improves the F-score only for one role over the baseline method and loses in two roles. The γ value for S is the lowest among all feature combinations. This indicates that the benefit due to inclusion of role transition probability varies from one feature combination type to another.

5. Conclusions

We find that when the role transition information for a participant across consecutive meeting slices is included, precision of the role recognition improves as compared to classifying roles in each meeting slice independently. The improvement in precision is more for roles such as Gatekeeper and Protagonist, which occur less frequently. Increased precision in turn improves the F-score of the role recognition for the Gatekeeper. In the present work, we consider the role transition in two consecutive slices. However, the role dynamics over three or more slices as well as interpersonal role dynamics could be used to improve the role recognition further. Unlike constructed corpus such as AMI, role recognition in realistic situation would require further investigation. These are parts of our future work.

6. References

[1] S. Banerjee, C. Rose, and A. I. Rudnicky, “The necessity of a meeting recording and playback system, and the benefit of topic–level annotations to meeting browsing,” in *IFIP Conference on Human-Computer Interaction*. Springer, 2005, pp. 643–656.

[2] A. Vinciarelli and S. Favre, “Broadcast news story segmentation using social network analysis and hidden markov models,” in *Proceedings of the 15th ACM international conference on Multimedia*. ACM, 2007, pp. 261–264.

[3] A. Vinciarelli, “Sociometry based multiparty audio recordings summarization,” in *18th International Conference on Pattern Recognition (ICPR)*, vol. 2. IEEE, 2006, pp. 1154–1157.

[4] R. F. Bales, *Personality and interpersonal behavior.* Holt, Rinehart, and Winston, 1969.

[5] B. J. Biddle, “Recent developments in role theory,” *Annual review of sociology*, vol. 12, pp. 67–92, 1986.

[6] B. Wrede and E. Shriberg, “Spotting “hot spots” in meetings: human judgments and prosodic cues.” in *Proc. INTERSPEECH*, 2003.

[7] A. Sapru and H. Bourlard, “Automatic recognition of emergent social roles in small group interactions,” *IEEE Transactions on Multimedia*, vol. 17, no. 5, pp. 746–760, 2015.

[8] S. Banerjee and A. I. Rudnicky, “Using simple speech–based features to detect the state of a meeting and the roles of the meeting participants,” in *Proc. of the Int. Conf. on Spoken Language Processing (ICSLP), Jeju Island*, 2004.

[9] S. Banerjee, J. Cohen, T. Quisel, A. Chan, Y. Patodia, Z. Al Bawab, R. Zhang, A. Black, R. M. Stern, A. I. Rudnicky, P. E. Rybski, and M. Veloso, “Creating multi-modal, user-centric records of meetings with the carnegie mellon meeting recorder architecture,” in *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, Meeting Recognition Workshop, Montreal*, 2004.

[10] S. Favre, H. Salamin, J. Dines, and A. Vinciarelli, “Role recognition in multiparty recordings using social affiliation networks and discrete distributions,” in *Proceedings of the 10th international conference on Multimodal interfaces*. ACM, 2008, pp. 29–36.

[11] S. Favre, A. Dielmann, and A. Vinciarelli, “Automatic role recognition in multiparty recordings using social networks and probabilistic sequential models,” in *Proceedings of the 17th ACM international conference on Multimedia*, 2009, pp. 585–588.

[12] K. Laskowski, M. Ostendorf, and T. Schultz, “Modeling vocal interaction for text-independent participant characterization in multi-party conversation,” in *Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue*. Association for Computational Linguistics, 2008, pp. 148–155.

[13] ——, “Modeling vocal interaction for text-independent classification of conversation type,” in *Proc. SIGdial*, vol. 194201, 2007.

[14] A. Sapru and F. Valente, “Automatic speaker role labeling in ami meetings: recognition of formal and social roles,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 5057–5060.

[15] M. Zancanaro, B. Lepri, and F. Pianesi, “Automatic detection of group functional roles in face to face interactions,” in *Proceedings of the 8th international conference on Multimodal interfaces*. ACM, 2006, pp. 28–34.

[16] F. Pianesi, M. Zancanaro, B. Lepri, and A. Cappelletti, “A multimodal annotated corpus of consensus decision making meetings,” *Language Resources and Evaluation*, vol. 41, no. 3-4, pp. 409–429, 2007.

[17] F. Valente and A. Vinciarelli, “Language-independent socio-emotional role recognition in the AMI meetings corpus.” in *Proc. INTERSPEECH*, 2011, pp. 3077–3080.

[18] J. Carletta, “Unleashing the killer corpus: experiences in creating the multi-everything AMI meeting corpus,” *Language Resources and Evaluation*, vol. 41, no. 2, pp. 181–190, 2007.

[19] T. Wilson and G. Hofer, “Using linguistic and vocal expressiveness in social role recognition,” in *Proceedings of the 16th international conference on Intelligent user interfaces*. ACM, 2011, pp. 419–422.

[20] A. Sapru and H. Bourlard, “Automatic social role recognition in professional meetings using conditional random fields,” in *Proceedings of Interspeech*, 2013.

[21] ——, “Investigating the impact of language style and vocal expression on social roles of participants in professional meetings,” in *Humaine Association Conference on Affective Computing and Intelligent Interaction (ACII).* IEEE, 2013, pp. 324–329.

[22] ——, “Detecting speaker roles and topic changes in multiparty conversations using latent topic models.” in *Proc. INTERSPEECH*, 2014, pp. 2882–2886.

[23] A. Vinciarelli, F. Valente, S. H. Yella, and A. Sapru, “Understanding social signals in multi-party conversations: Automatic recognition of socio-emotional roles in the AMI meeting corpus,” in *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2011, pp. 374–379.

[24] W. Dong, B. Lepri, F. Pianesi, and A. Pentland, “Modeling functional roles dynamics in small group interactions,” *IEEE Transactions on Multimedia,*, vol. 15, no. 1, pp. 83–95, 2013.

[25] J. Zhang and S. Gong, “Action categorization by structural probabilistic latent semantic analysis,” *Computer Vision and Image Understanding*, vol. 114, no. 8, pp. 857–864, 2010.

[26] J. W. Pennebaker, M. R. Mehl, and K. G. Niederhoffer, “Psychological aspects of natural language use: Our words, our selves,” *Annual review of psychology*, vol. 54, no. 1, pp. 547–577, 2003.