MIT Media Laboratory Technical Note 609 Ninth Int'l Conf on Multimodal Interfaces, Nov 12-15, 2007, Nagoya Japan

Using the Influence Model to Recognize Functional Roles in Meetings

Wen Dong
The MIT Media Laboratory
20 Ames Street
02139-4307 Cambridge, MA, USA
wdong@media.mit.edu

Bruno Lepri
FBK-Irst
Via Sommarive
38050 Povo-Trento, Italy
lepri@itc.it

Alessandro Cappelletti
FBK-Irst
Via Sommarive
38050 Povo-Trento, Italy
cappelle@itc.it

Alex (Sandy) Pentland
The MIT Media Laboratory
20 Ames Street
02139-4307 Cambridge, MA, USA
sandy@media.mit.edu

Fabio Pianesi
FBK-Irst
Via Sommarive
38050 Povo-Trento, Italy
pianesi@itc.it

Massimo Zancanaro
FBK-Irst
Via Sommarive
38050 Povo-Trento, Italy
zancana@itc.it

ABSTRACT
In this paper, an influence model is used to recognize functional roles played during meetings. Previous works on the same corpus demonstrated a high recognition accuracy using SVMs with RBF kernels. In this paper, we discuss the problems of that approach, mainly over-fitting, the curse of dimensionality and the inability to generalize to different group configurations. We present results obtained with an influence modeling method that avoid these problems and ensures both greater robustness and generalization capability.

Categories and Subject Descriptors
H.5.3 [INFORMATION INTERFACES AND PRESENTATION]: Group and Organization Interfaces: *Computer-supported cooperative work - Synchronous interaction*

I.2.10. [ARTIFICIAL INTELLIGENCE]: Vision and Scene Understanding – *Perceptual Reasoning*

General Terms
Design, Experimentation, Human Factors.

Keywords
Group Interaction, Support Vector Machines, Intelligent Environments.

1. INTRODUCTION
The complexity of social dynamics occurring in small group interactions often hinders the performance of teams. The availability of rich multimodal information about what is going on during the meeting makes it possible to explore the possibility of providing various kinds of support to dysfunctional teams, from facilitation to training sessions addressing both the individuals and the group as a whole. A necessary step in this direction is that of automatically capturing and understanding group dynamics.

In order to improve performance of meetings, external interventions by experts such as facilitators and trainers are commonly employed. Facilitators participate in the meetings as external elements of the group and their role is to help participants maintaining a fair and focused behavior as well as directing and setting the pace of the discussion. Training experiences aim at increasing the relational skills of individual participants by providing an offline (with respect to meetings) guidance—or coaching—so that the team eventually will be able to overcome or to cope with its disfunctionalities.

In [17], the absence of any detectable difference in the acceptability of reports about own relational behaviour according to whether they had been produced by a human expert or by an automatic system was reported. Clearly, crucial to any such an automatic system is that it be capable of understanding people social behaviour, e.g., by abstracting over low level (visual, acoustic, etc.) information to produce medium-/coarse-grained one about the social/relational roles members play in the group. The latter is the kind of information that most coaches and group facilitators (implicitly or explicitly) use while doing their job. In [21;18], sliding windows multiclass SVMs with radial kernels were used to recognize functional relational roles in meetings. The results were very positive, with the macro F scores for the different roles above 80%. However this approach suffers from two limitations. First, the observation vector included not only the features of the participant whose role had to be detected but also those of all the other participants and it is therefore very sensible to the curse of dimensionality [5], which might artificially inflate results. The second limitation is that radial kernels might turn out to have an infinite VC dimensionality and that can easily lead to over-fitting [8].

In this paper, we investigate a new framework for functional role detection in meetings, the 'influence model' [4; 12], and compare this approach with multi-class SVMs based on linear and RBF kernels, and Hidden Markov Models. Among its advantages, the

influence model overcomes problems such that the curse of dimensionality that easily affect alternative framework (e.g., SVM). We will show that when the features from a single participant only are used both the SVM and HMM approaches achieve similar accuracy at role classification. When features from all the participants are included the SVM performance increases dramatically, but at the cost of a loss of robustness and generalizability. The influence model, on the other hand, can take into account the features of other participants in a more robust and generalizable manner, with an intermediate increase in accuracy over the single-person role classification methods. Both all-person SVM and influence model results at labeling roles are comparable to the human inter-rater reliability, but show different patterns of error.

This paper is structured as follows: in the next section, previous work relevant to the topic of the automatic detection of social phenomena in groups is presented and discussed. Section 3 introduces the corpus used for our experiments, the particular phenomenon we are tackling—namely, functional relational roles—and discusses some previous results on their automatic detection. Section 4 introduces the influence modelling, while the results of our experiments are discussed in section 5. Finally, section 6 draws the conclusions and hints at future work.

2. PREVIOUS AND RELATED WORK

Multimodal analysis of group behavior is a relatively recent research area compared to the large body of studies focusing on multimodality as a flexible, efficient, and powerfully expressive mean for human-computer interaction (see [Oviatt, 2002] for a survey of multimodal input). Despite this, some research groups have started producing important results. For instance, McCowan et al. [14] developed a statistical framework based on different Hidden Markov Models to recognize the sequences of group actions using observations provided by a set of audio-visual features obtained by monitoring the individuals' actions. For example, "discussion" is a group action which can be recognized from the verbal activity of individuals. Rienks and Heylen [19] used Support Vector Machines to automatically detect the team members who play a dominating role in a meeting, by relying on a few basic features. In a more recent work, Rienks et al. [20] addressed the problem of automatically detecting participant's influence levels in meetings using static models (i.e. SVMs) and a dynamic model, the team-player influence model (a dynamic Bayesian network with a two-level structure: the player level and the team level). Banerjee and Rudnick [3] proposed a simple taxonomy of participant roles and meeting states, and then trained a decision tree classifier to learn them from simple speech-based features. The classifier takes as input a feature representation of a short time window during the meeting (meeting history) and classifies the roles and the states at the end of the window. Finally, Brdiczka and colleagues [7] developed a real-time detector for configurations of interaction groups.

3. THE MISSION SURVIVAL CORPUS

For the experiments discussed in this paper, we have used the Mission Survival Corpus [18], a multimodal annotated corpus based on the audio and the video recordings of eight meetings that took place in a lab setting appropriately equipped with cameras and microphones. Each meeting consisted of four people engaged in the solution of the "mission survival task". This task is frequently used in experimental and social psychology to elicit decision-making processes in small groups. Originally designed by National Aeronautics and Space Administration (NASA) to train astronauts, the Survival Task proved to be a good indicator of group decision making processes [13]. The exercise consists in promoting group discussion by asking participants to reach a consensus on how to survive in a disaster scenario, like moon landing or a plane crash in Canada. The group has to rank a number (usually 15) of items according to their importance for crew members to survive. In our setting, we used the plane crash version. This consensus decision making scenario was chosen for the purpose of meeting dynamics analysis mainly because of the intensive engagement requested to groups in order to reach a mutual agreement, thus offering the possibility to observe a large set of social dynamics and attitudes. In our setting, we retained the basic structure of the Survival Task with minor adjustments: a) the task was competitive across groups/team, with a price being awarded to the group providing the best survival kit. b) the task was collaborative and based on consensus within the group, meaning that a participant's proposal became part of the common sorted list only if he/she managed to convince the other of the validity of his/her proposal.

Figure 1. A picture of the experimental setting.

The recording equipment consisted of five Firewire cameras—four placed on the four corners of the room and one directly above the table— and four web cameras installed on the walls surrounding the table. Speech activity was recorded using four close-talk microphones, six tabletop microphones and seven T-shaped microphone arrays, each consisting of four omni-directional microphones installed on the four walls in order to obtain an optimal coverage of the environment for speaker localization and tracking.

Each session was automatically segmented labeling the speech activity recorded by the close-talk microphones every 330ms [9]. The fidgeting—the amount of energy in a person's body and hands—was automatically tracked by using skin region features and temporal motion [10]. The values of fidgeting for hands and body were extracted for each participant and normalized on the fidgeting activity of the person during the entire meeting.

3.1 The Functional Role Coding Scheme

The Functional Role Coding Scheme (FRCS) was partially inspired by Bales' Interaction Process Analysis [2]. It consists of

ten labels that identify the behavior of each participant in two complementary areas: the Task Area, which includes functional roles related to facilitation and coordination tasks as well as to technical experience of members; the Socio Emotional Area, which is concerned with the relationships between group members and the functional roles "oriented toward the functioning of the group as a group". Below we give a synthetic description of the FRCS (for more information, see [18].

The Task Area functional roles consist or: the Orienteer (o)—she orients the group by introducing the agenda, defining goals and procedures, keeping the group focused and on track and summarizing the most important arguments and the group decisions. The Giver (g): she provides factual information and answers to questions, states her beliefs and attitudes about an idea, expresses personal values and factual information. The Seeker (s), who requests information, as well as clarifications, to promote effective group decisions. The Procedural Technician (pt); she uses the resources available to the group, managing them for the sake of the group. The follower (f), who just listens, without actively participating in the interaction.

The Socio-Emotional functional roles: the Attacker (a); she deflates the status of others, expresses disapproval, and attacks the group or the problem. The Gate-keeper (gk), who is the group moderator, mediates the communicative relations, encourages and facilitates the participation and regulates the flow of communication. The Protagonist (p); she takes the floor, driving the conversation, assuming a personal perspective and asserting her authority. The Supporter (su), who shows a cooperative attitude demonstrating understanding, attention and acceptance as well as providing technical and relational support. The Neutral Role (n), played by those who passively accept the ideas of the others, serving as an audience in group discussion.

Of course, participants may—and often do—play different roles during the meeting, but at a given time each of them plays exactly one role in the Task Area and one role in the Socio-Emotional one.

The FCRS was showed to have a high inter-rater reliability (Cohen's statistics $\kappa = 0.70$ for the Task Area; $\kappa = 0.60$ for the Socio-Emotional Area).

3.2 Predicting the functional roles

In [21 and 18], an SVM-based approach was discussed that predicts the functional roles taken by the participants of a meeting from information such as the speech activity and the fidgeting of each participant in a time window. The bound-constrained SV classification algorithm with a Gaussian RBF kernel was used. The cost parameter C and the kernel parameter γ were estimated through the grid technique by means of cross-fold validation using a factor of 10.

In the first attempt ([21]), only the features of the participant himself were used to detect his role and different window' sizes were tested. In this case, the accuracy for the Task Area roles reached 0.65 for the 14-second window seconds, and the accuracy for the Socio-Emotional roles reached 0.70 for the 12-second window.

In the second attempt [18], the features of all the four participants were used to predict the role of a single participant; again different window's sizes were tested. Note that this approach relies on a very large feature vector and risks problems of overfitting and robustness. The accuracy at 9-second window reached 0.90 for Task area roles and 0.92 for Socio-Emotional roles with high precision and recall for all the roles (see Table 1). In both attempts, the precision and recall on the individual roles were not homogeneous

Table1. Precision and recall of the different roles in the second attempt using an SVM-based approach.

	Neutral	*Supporter*	*Protagonist*	*Attacker*
Precision	0.89	0.89	0.91	0.83
Recall	0.92	0.81	0.91	0.74
F	0.91	0.85	0.91	0.78

	Follower	*Orienteer*	*Giver*	*Seeker*
Precision	0.84	0.93	0.93	0.89
Recall	0.90	0.87	0.91	0.68
F	0.87	0.90	0.92	0.77

4. INFLUENCE MODELING

The influence modeling approach is a method that can effectively deal both with the curse of dimensionality and the over-fitting problem. It has been developed in the tradition of the N-heads dynamic programming on coupled hidden Markov models [15], the observable structure influence model [1], and the partially observable influence model [4]. It extends, though, these previous models by providing greater generality, accuracy, and efficiency.

The influence modeling is a team-of-observers approach to complex and highly structured interacting processes. In this model, different observers look at different data, and can adapt themselves according to different statistics in the data. The different observers find other observes whose *latent state*, rather than observations, are correlated, and use these observers to form an estimation network. In this way, we effectively exploit the interaction of the underlying interacting processes, while avoiding the risk of overfitting and the difficulties of observations with large dimensionality.

Mathematically speaking, a latent structure influence process is a stochastic process $\{S_t^{(c)}, Y_t^{(c)} : c \in \{1,...,C\}, t \in N\}$. In this process, the latent variables $S_t^{(1)},...,S_t^{(C)}$ each have finite number of possible values $S_t^{(c)} \in \{1,...,m_c\}$ and their (marginal) probability distributions evolve as the following:

$$\Pr(S_t^{(c)} = s) = \pi_s^{(c)} \quad (1)$$

$$\Pr(S_{t+1}^{(c)} = s) = \sum_{c_1=1}^{C} \sum_{s_1=1}^{m_{c_1}} h_{s_1,s}^{(c_1,c)} \Pr(S_t^{(c)} = s_1) \quad (2)$$

where $1 \le s \le m_c$ and $h_{s_1,s}^{(c_1,c)} = d^{(c_1,c)} a_{s_1,s}^{(c_1,c)}$ ($a_{s_1,s}^{(c_1,c)}$ represent the relations of different states for the interacting processes, and $d^{(c_1,c)}$ represent the influence among the processes). The observations $\vec{Y}_c = (Y_t^{(1)},...,Y_t^{(C)})$ are coupled with the latent states $\vec{S}_c = (S_t^{(1)},...,S_t^{(C)})$ through a memory-less channel:

$$P(\vec{S}_c)P(\vec{Y}_c \mid \vec{S}_c) = \prod_{c=1}^{C} P(S_t^{(c)})P(Y_t^{(c)} \mid S_t^{(c)}) \qquad (1)$$

We give the forward-backward algorithm of the latent structure influence process (for latent state inference), and the maximum likelihood algorithm (for parameter estimation) below. A detailed discussion of this model, as well as its algorithms, can be found in [11; 12].

Given the parameters of the influence model, as well as the observation sequences, the marginal probability distributions on latent states for individual interacting processes can be computed as follows.

$$\tilde{\alpha}_1^{(c)}(s) = \pi_s^{(c)} p(y_1^{(c)} \mid s)$$

$$\tilde{\alpha}_{t+1}^{(c)}(s) = p(y_t^{(c)} \mid s) \sum_{c_1=1}^{C} \sum_{s_1=1}^{m_{c_1}} \alpha_t^{(c)}(s_1) h_{s_1,s}^{(c_1,c)}$$

$$\text{scale}_t^{(c)}(s) = \sum_{s=1}^{m_c} \tilde{\alpha}_t^{(c)}(s)$$

$$\alpha_t^{(c)}(s) = \tilde{\alpha}_t^{(c)}(s) \Big/ \text{scale}_t^{(c)}(s)$$

$$\beta_T^{(c)}(s) = 1$$

$$\beta_{t<T}^{(c)}(s) = \sum_{c_1=1}^{C} \sum_{s_1=1}^{m_{c_1}} h_{s,s_1}^{(c,c_1)} \beta_{t+1}^{(c_1)} p(y_{t+1}^{c_1} \mid s_1) \Big/ \text{scale}_{t+1}^{(c_1)}$$

$$\gamma_t^{(c)}(s) = \alpha_t^{(c)}(s)\beta_t^{(c)}(s)$$

$$\xi_{t-1\to t}^{(c_1,c_2)}(s_1,s_2) = h_{s_1,s_2}^{(c_1,c_2)} \alpha_{t-1}^{(c_1)}(s_1) \beta_t^{(c_2)}(s_2) p(y_{t+1}^{c_2} \mid s_2)$$

Given the observation sequences, as well as the inferred latent state sequences, the parameters can be re-estimated as the following:

$$a_{s_1,s_2}^{(c_1,c_2)} = \frac{\sum_{t=2}^{T} \xi_{t-1\to t}^{(c_1,c_2)}(s_1,s_2)}{\sum_{t=2}^{T} \sum_{s_2=1}^{m_{c_2}} \xi_{t-1\to t}^{(c_1,c_2)}(s_1,s_2)}$$

$$d^{(c_1,c_2)} = \frac{\sum_{t=2}^{T} \sum_{s_1=1}^{m_{c_1}} \sum_{s_2=1}^{m_{c_2}} \xi_{t-1\to t}^{(c_1,c_2)}(s_1,s_2)}{\sum_{t=2}^{T} \sum_{c_2=1}^{C} \sum_{s_1=1}^{m_{c_1}} \sum_{s_2=1}^{m_{c_2}} \xi_{t-1\to t}^{(c_1,c_2)}(s_1,s_2)}$$

$$\pi_s^{(c)} = \frac{\gamma_1^{(c)}(s)}{\sum_{s=1}^{m_c} \gamma_1^{(c)}(s)}$$

The parameters that couple the latent states with the observations are computed as usual.

The following imaginary power plant network example illustrates the interacting dynamic processes we are referring to, and how the influence modeling simultaneously exploring and exploiting the structure among them. In this example, we have a network of six power plants, and our task is to estimate whether a power plant is normal or overheated from noisy observations of it. A natural approach is to make estimations of the states of a power plant from the observations of not only this power plant, but also the related power plants, in a short window around the time of the estimated states. The estimation is a chicken and egg problem: the more we know about the structure, the better we can estimate the latent states, and vice versa.

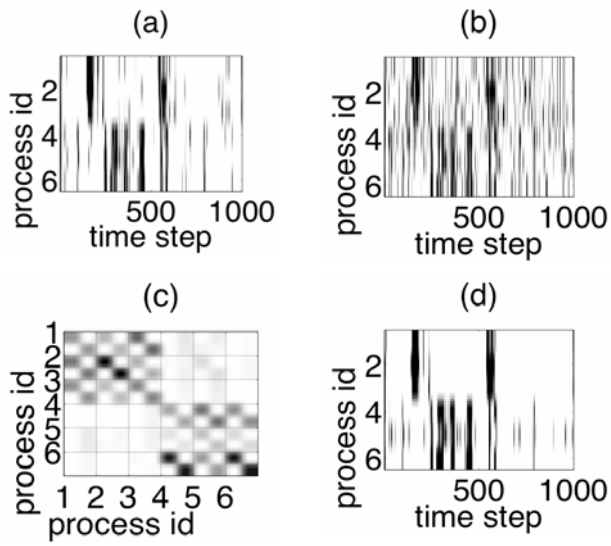

Figure 2: Estimating network structure and latent states simultaneously from noisy observations with the influence model. The task is to estimate the true states, as well as the interaction, from noisy observations shown in (b). The recovered interaction structure in (c) has 90% accuracy, and the estimated the latent states in (d) have 95% accuracy.

An important consideration in choosing a multi-class classifier is whether the classifier, after it is trained from a training data set, can generalize to future applications. With increased dimensionality and without regularization, even a linear classifier, which is considered stable, can overfit. The latent structure influence modeling of interacting processes avoids the curse of dimensionality problem with the team of observers approach. In this approach, the individual observers only look at the latent states of the other related observers, rather than looking at the raw observations, and thus are less likely to be overfit and more likely to be generalizable.

Figure 3 compares the performances of several dynamic latent structure models (the influence model, the hidden Markov model with 16 latent states and 10-dimensional Gaussian observations, the hidden Markov model with 64 latent states and 10-dimensional Gaussian observations, and 10 hidden Markov models, each on one dimensional data). Of the 1000 samples, we use the first 250 for training and the rest 750 for validation.

Judged from Figure 3, the logarithmically scaled number of parameters of the influence model allows us to attain high accuracy based on a relatively small number of observations. This is because the influence model preserves the asymptotic marginal probability distributions of the individual "bits", as well as the linear relationship among them. Hence, the influence model shrinks the number of parameters of the original hidden Markov

model logarithmically and in an efficient way, while still preserving the principal dynamics of the process.

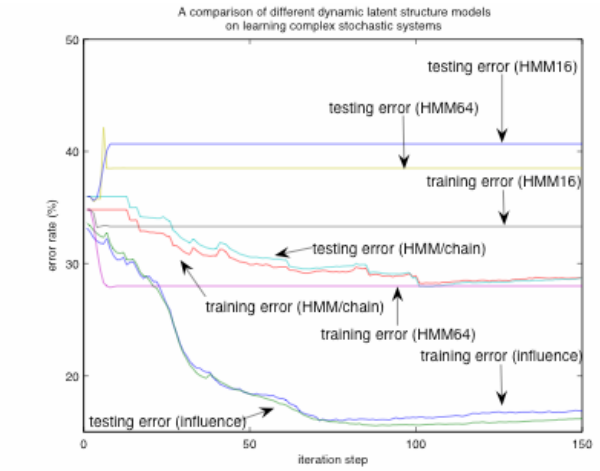

Figure 3: Latent state influence process is immune to overfitting.

5. Detection of functional roles

In this section we compare the results obtained from three different approaches: Support Vector Machines with linear kernel, Hidden Markov Models and Influence modeling.

A simple multi-class SVM approach, although powerful, has several limitations in its generalization capability. The first issue is related to finding general features applicable to all speakers. Different speakers might have different ways to fulfill their functional roles in a group discussion. Having a speaker specific implementation is a nontrivial task for support vector classifiers. The second issue is related to the curse of dimensionality. When we make use of the observations of other speakers for our classification task, the length of the observation vector grows linearly with a large multiplication constant. For instance, using 9-seconds windows the length of the observation vector is 432 (9 seconds x 3 samples/second x features x 4 speakers). The third issue is about the assessment of the trained classifier, as well as how the speakers interact with each other. Extracting an intuitive understanding of group interactions among the speakers from the trained support vector classifiers, as well as how the individuals fulfill their function roles is not easy. A final issue is generalizability to different numbers of speakers. The SVM approach is not modular in the number of participants, whereas a network approach can be scaled to different sizes of groups. As a result, a natural next step is to use a Bayesian hierarchical dynamic model, and compare the performance with that of a standard multi-class SVM.

Moreover, as already hinted at, the results by [18] might have another weakness related to the generalization capability. The RBF kernel, in fact, might have an infinite Vapnik-Chernovenkis dimension and might be subject to over-fitting and to a poor generalization capacity. To overcome this problem, we changed the partition between train-set and test-set in order to try out the robustness of our approach with respect to over-fitting. We used four meetings for train-set (040805_0930, 040805_1100, 040805_1400, and 180805_1400) and the other meetings for test-set (030805_1100, 090805_1100, 180805_0930, and 180805_1130). In all these experiments we modeled role assignment as a multi-class classification problem and used Support Vector Machines as classifier. A linear kernel was used in order to reduce the risk of overfitting. The highest accuracy score obtained is 70%. The macro precision and the macro recall for Task area roles are 48% and 52%. The performance is worst for Socio-emotional area roles: the macro precision is 39% and the macro recall 48%. Table 1 and Table 2 show the confusion matrices for Task area roles and Socio-Emotional area roles respectively. The observation vector is composed of the smoothed version of speaking/non-speaking, hand movement, and body movement of the speaker under investigation, as well as the number of simultaneous speakers in a fixed-length window centered around the moment of interest. We take this window to be from 10 seconds before till 10 seconds after the moment of interest.

Table 1: Confusion matrix between the ground truth and the typical classification result for task roles with Support Vector Machine and linear kernel (G=giver, N=neutral, O=orienteer, and S=seeker)

		SVM classification on test data				
		Giver	Neutral	Orienteer	Seeker	Total
Ground truth	G	8468	4049	1624	635	14776
	N	2517	29304	520	899	33240
	O	1385	527	205	74	2191
	S	35	18	535	717	1305
	Total	13571	28364	2416	7161	51512

Table 2: Confusion matrix between the ground truth and the typical classification result for socio-emotional roles with Support Vector Machine and linear kernel (A=attacker, N=neutral, P=protagonist, and S=supporter)

		SVM classification on test data				
		Attacker	Neutral	Protagonist	Supporter	Total
Ground truth	A	74	70	20	21	185
	N	460	32766	3936	1309	38471
	P	322	2463	5777	818	9380
	S	146	1351	1748	231	3476
	Total	124	35386	8048	7954	51512

A major goal of our work was to provide for a fair comparison between SVM approach and the Influence Model approach trying to avoid over-fitted results with SVMs.

In the influence modeling of the speakers' functional roles, we used $2n$ number of interacting processes to model the task roles and the social roles of the n individual speakers in a meeting. The observations for the individual processes are the corresponding speakers' raw features (speaking/non-speaking, body movement, hand movement, and number of simultaneous

speakers) averaged over short fixed-length time windows centered around the observation times. The latent states for the individual processes are the corresponding labels. In the training phase of influence modeling, we find out the observation statistics of different functional role classes, as well as the interaction of different speakers with the EM (expectation maximization) algorithm, based on the training data. In the application phase, we infer the individual speakers' social/task roles based on the observations about the individual speakers, as well as their interactions, using the parameters previously trained.

We partitioned the data set of the eight meetings into two parts, as in the SVM experiments, and estimated the generalization capability of the trained classifier by two fold cross validation.

With the influence modeling, we can generally get 75% accuracy in classifying both the task roles and the social roles. We are satisfied with this performance since identifying the functional roles without semantics is normally considered as a very hard problem.

The trained influence matrix (Figure 4) gives an intuition on how the "team of observers" cooperates to gain the maximum classification accuracy. In this matrix, an entry at row x and column y indicates how the state corresponding to row x is indicative of the state corresponding to column y. The eight observers (for the eight speaker/functional-class combinations) tend to vote for a neutral state on a speaker's functional role, with regard to their decisions at the next observation time. The observers also model the interaction of the different functional roles among the speakers in the training phase, and use the model for classification.

What is striking about this influence matrix is its modularity. The influence on each speaker is similar to that of all the other speakers. This means that we should be able to generalize the influence model to groups with different numbers of participants by replicating the appropriate portions of the influence matrix.

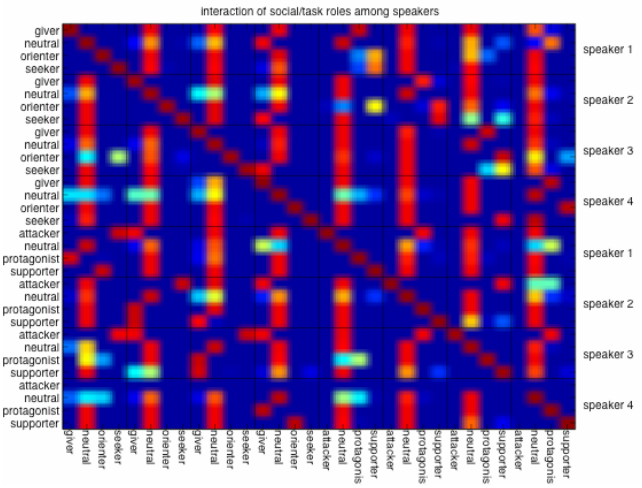

Figure 4: Interaction of social/task roles among speakers. An entry at row x and column y indicates how large the state corresponding to row x is indicative of the state corresponding to column y (red means more indicative, and blue means less indicative).

Turning to the hidden Markov model, we trained eight hidden Markov models for the four speakers' social/task roles. Without the information about the other participants' functional roles, HHM typically yields 60% accuracy for task roles, 70% accuracy for social roles, and 65% overall accuracy. This is similar to the accuracy of the SVM approach. The typical confusion matrices for task/social roles are given in Table 3 and 4.

Table 3: Confusion matrix between the ground truth and the typical classification result for task roles with one hidden Markov model per speaker (G=giver, N=neutral, O=orienteer, S=seeker)

		HMM per speaker classification on test data				
		Giver	Neutral	Orienteer	Seeker	Total
Ground truth	G	7126	3637	1370	2643	14776
	N	4728	23740	809	3963	33240
	O	1212	310	195	474	2191
	S	505	677	42	81	1305
	Total	13571	28364	2416	7161	51512

Table 4: Confusion matrix between the ground truth and the typical classification result for socio-emotional roles with one hidden Markov model per speaker (A=attacker, N=neutral, P=protagonist, and S=supporter)

		HMM per speaker classification on test data				
		A	N	P	S	Total
Ground truth	A	54	91	40	0	185
	N	20	30874	3017	4560	38471
	P	0	2825	4695	1860	9380
	S	50	1596	296	1534	3476
	Total	124	35386	8048	7954	51512

Putting these results together, it can be seen that by including the influence modeling to capture connections between speaker roles, we can achieve approximately 10% increase in accuracy, to about 75% overall accuracy. This is similar to the inter-rater accuracy of the human labeling of this corpus, and therefore may be close to an upper limit for modeling accuracy without risking overfitting. The macro precision and the macro recall of the Influence Model are 40% and 39% for Task area roles. The performance is better for Socio-Emotional area roles: the macro precision is 41% and the macro recall 50%. By comparing the confusion matrices for the influence model and the hidden Markov models, one can see that most of the improvements are in the majority classes, and are due to the fact that influence modeling uses the functional roles of other speakers. However, for the Socio-Emotional area roles the macro precision (52%) and the macro recall (51%) of the hidden Markov model are better than the macro precision (41%) and macro recall (50%) of the Influence Model. These results of the Influence Model are due to two different reasons: the high number

of false positives in the Attacker role classification and the not good performance at classifying the Supporter role.

Table 5: Confusion matrix between the ground truth and the typical classification result for task roles with influence modeling speaker (G=giver, N=neutral, O=orienteer, S=seeker)

		Influence model classification on test data				
		Giver	Neutral	Orienteer	Seeker	Total
Ground truth	G	8059	4225	1858	634	14776
	N	2535	29362	406	937	33240
	O	1304	500	320	67	2191
	S	526	714	64	1	1305
	Total	12424	34801	2648	1639	51512

Table 5: Confusion matrix between the ground truth and the typical classification result for socio-emotional roles with influence modeling (A=attacker, N=neutral, P=protagonist, and S=supporter)

		Influence model classification on test data				
		A	N	P	S	Total
Ground truth	A	74	72	19	20	185
	N	341	32767	3521	1842	38471
	P	269	2536	5290	1285	9380
	S	127	1281	1455	613	3476
	Total	811	36656	10285	3760	51512

6. CONCLUSION

In this paper, we have used the Influence Model for recognizing group functional roles played during meetings. In previous works we used Support Vector Machines with Gaussian RBF kernel and sliding windows. Both approaches produce a similar, medium level of classification accuracy (roughly 65%) when using only features from one individual.

When using features from all participants, the SVM approach obtains higher recognition accuracy (Pianesi et al., in press), but suffered from two problems related to generalization capability: (a) the curse of dimensionality (if we make use of the observations/features of other speakers for our classification task, the length of the observation/feature vector grows linearly with a large multiplication constant); (b) overfitting (the Gaussian RBF kernel might have infinite VC-dimension).

The Influence Model is a good technique to deal with these weaknesses. In fact, the latent structure influence modeling of interacting processes avoids the curse of dimensionality problem using the "team of observers" approach. In this approach, the individual observers only look at the latent states of the other related observers, which best summarize the observations from the perspectives of the latter, thus are less likely to suffer from overfitting and lack of generalization.

The performance obtained using Influence Model for recognizing group functional roles is comparable to the inter-rater reliability on this corpus of data: we can generally get 75% accuracy in classifying both the Task area roles and the Socio-Emotional area roles.

One interesting observation is that the Influence Model seems to be generalizable to different numbers of participants in the group, since the influence between participants was very similar for all subjects and all experiments. The ability to automatically adapt to different sized groups without retraining would allow a great increase in the flexibility and applicability of automatic role classification technology.

One important area for future work is that current training algorithm for the influence model does not do well at classifying the low-frequency classes (Orienteer/Seeker for Task area roles, and Attacker/Supporter for Socio-Emotional area roles). A direction for improvement is adding more features, and hierarchical training. In the future works, we plan to add some novel features starting from vocal energy, 3D postures and focus of attention.

7. ACKNOWLEDGMENTS

This work was partially supported by the UE under the CHIL (FP6) project.

8. REFERENCES

[1] Asavathiratham, C., Roy, S., Lesieutre, B., and Verghese, G. The influence model. In *IEEE Control Systems Magazine, Special Issue on Complex Systems*, (12) 2001.

[2] Bales, R.F. *Personality and interpersonal behavior*. New York: Holt, Rinehart and Winston, 1970.

[3] Banerjee, S., and Rudnicky, A.I. Using Simple Speech-based Features to Detect the State of a Meeting and the Roles of the Meeting Participants. In Proceedings of the 8th International Conference on Spoken language Processing (Interspeech 2004-ICSLP), Jeju Island, Korea, October 2004.

[4] Basu, S., Choudhury, T., Clarkson, B., and Pentland, A., Learning human interactions with the influence model. Technical report, MIT Media Laboratory vision and modeling technical report #539, 2001. URL .

[5] Bellman, R., 1961. *Adaptive Control Processes: A guided Tour*. Princeton University Press.

[6] Benne, K.D., Sheats, P. Functional Roles of Group Members, *Journal of Social Issues* 4, 41-49. (1948)

[7] Brdiczka, O., Maisonnasse, J., and Reignier, P. Automatic Detection of Interaction Groups. In Proceedings of the 7th International Conference on Multimodal Interface. Trento, Italy, October 2005.

[8] Burges, C.J.C., A Tutorial on Support Vector Machines for Pattern Recognition. In Data Mining and Knowledge Discovery, Vol. 2, No. 2, pp. 121-167. (1998)

[9] Carli, G., Gretter, G. A Start-End Point Detection Algorithm for a Real-Time Acoustic Front-End based on DSP32C VME Board. In Proceedings of ICSPAT, Boston, USA. 1992.

[10] Chippendale P 7th International Conference on Automatic Face and Gesture Recognition - FG2006 (IEEE) Southampton, UK, April 2006.

[11] Dong W., Influence Modeling of Complex Stochastic Processes, Masters thesis, MIT, 2006

[12] Dong, W. and Pentland, A. Modeling Influence between Experts. In *Lecture Notes on Artificial Intelligence, Special Volume on Human Computing*. Springer-Verlag, 2007

[13] Hall, J. W., Watson, W. H. (1970) The Effects of a normative intervention on group decision-making performance. In Human Relations, 23(4), 299-317.

[14] McCowan I., Bengio S., Gatica-Perez D., Lathoud G., Barnard M., and Zhang D. Automatic Analysis of Multimodal Group Actions in Meetings. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 27 (3), pp. 395-317, 2005

[15] Oliver, N., Rosario, B., and Pentland, A., A Bayesian computer vision system for modeling human interactions. In *IEEE transactions on pattern analysis and machine learning*, 22(8) 831-843, 2000.

[16] Oviatt, S. Multimodal Interfaces. In Handbook of Human-Computer Interaction, (ed. by J. Jacko & A. Sears), Lawrence Erlbaum: New Jersey, 2002.

[17] Pianesi, F., Zancanaro M., Not E., Leonardi C., Falcon V., and Lepri B. Multimodal Support to Group Dynamics. In *Personal and Ubiquitous Computing*. Vol. 12, No. 2, 2008.

[18] Pianesi F., Zancanaro M., Lepri B., Cappelletti A. *in press* Multimodal Annotated Corpora of Consensus Decision Making Meetings. To appear in *The Journal of Language Resources and Evaluation*

[19] Rienks R., and Heylen D., Dominance Detection in Meetings Using Easily Obtainable Features. In Revised Selected Papers of the 2nd Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms. Edinburgh, Scotland, October 2006.

[20] Rienks R., Zhang D., Gatica Perez D., and Post W. Detection and Application of Influence Rankings in Small Group Meetings. In Proceedings of International Conference of Multimodal Interfaces ICMI-06, 2006

[21] Zancanaro M., Lepri B., Pianesi F. Automatic Detection of Group Functional Roles in Face to Face Interactions. In Proceedings of International Conference of Multimodal Interfaces ICMI-06, 2006.