

Behavior Modeling with Probabilistic Context Free Grammars

Sahin Cem Geyik, Jierui Xie and Boleslaw K. Szymanski

Department of Computer Science and Center for Pervasive Computing and Networking
Rensselaer Polytechnic Institute
Troy, New York 12180
{geyiks,xiej2,szymansk}@cs.rpi.edu

Abstract – Identifying the behavioral patterns in a social network setting is beneficial to understand how people behave in certain application domains. Such patterns can also be utilized to characterize social signals such as social roles from interactions. In this work, we examine how probabilistic context free grammars (PCFGs) can be utilized to model interactions and role taking in a social network. We describe how to automatically build a PCFG given a set of interactions as the training data. Our experiments on the Mission Survival Corpus 1 (MSC-1) dataset show that PCFGs are a concise way of modeling social entity behaviors and are useful in understanding the probability distribution of interactions as well as the behavior types that are observed.

Keywords: Social Networks, behavior modeling, behavioral patterns, PCFGs.

1 Introduction

Social networks analysis [1] includes examining the actions of entities in a social setting. These actions can be either interactions between entities (e.g. talking, exchanging items etc.), or actions which do not include interactions, but nevertheless in the social context, hence resulting from the social setting. Such actions often contain patterns which are specific to the application domain. For example, it is natural to assume that a brainstorming session often include group discussions which are not dominated by one person but rather involves dyadic communications. In the eyes of a single entity however, this application domain means that the opposing person (hence the interacting entity) changes rapidly.

It is important to understand these patterns to realize the characteristics of a social setting. A simple example can be given in a company where there is a certain order in the interactions. If a customer has to go through certain department tables to accomplish a certain task, the placement of these departments in the

commonly used order saves time and energy for both the customer and the worker. Another example can be given in the military context. A large community has a certain set of interaction patterns between its members which can be observed in the long run. A potential dangerous activity can be recognized from the abnormal patterns that occur in a community that is being monitored. Such a recognition ability can also be used in the work environment to predict failures as well improve performance in a business process.

In this paper, we introduce a novel method for modeling social node behaviors. We propose utilization of probabilistic context free grammars (PCFGs) which are probabilistic versions of regular context free grammars. Given a set of action sequences in a network setting, an automatic generation method, which we also present here, can build a PCFG which concisely holds probabilities for action patterns. This PCFG can either be used to predict future behaviors, or to understand these patterns due to its concise nature and appropriateness for manual inspection.

The rest of the paper is as follows. In the next section we go over the details of our method, which include the construction algorithm as well as the essence of our behavior modeling idea. Next we give the results of our PCFG modeling on the Mission Survival Corpus 1 (MSC-1) dataset [2] owned by Project FBK (Fondazione Bruno Kessler). We provide grammars constructed for socio and task label categories defined within the dataset. Following the results, the related work section discusses the recent developments in social network analysis. We finalize the paper with the summary, conclusions and future work.

2 Methodology

In this section we give the details of our modeling method. We will first give the definition of the PCFG as well as how it can model the set of actions being taken by a social entity. We finish the section with the overview of our automated PCFG construction algo-

rithm.

2.1 PCFGs and Modeling Idea

A Probabilistic Context Free Grammar [3] consists of a five-tuple $\langle S_{nt}, S_t, R_g, \text{Prob}, \text{Start} \rangle$ where:

- Start is the initial nonterminal symbol for any production of a string from the grammar,
- S_{nt} is a list of nonterminal symbols, each of which points to further production rules,
- S_t is a list of terminal symbols which are the symbols actually seen in the sentences,
- R_g is a list of production rules that define how a string of terminal and nonterminal symbols can be immediately produced from a nonterminal symbol,
- Prob is a list of probabilities, each of which is assigned to a rule and defines the likelihood with which this rule is used in parsing or generating a sentence.

To put it simply, a PCFG is an extension of the ordinary context free grammar in which the rules of each nonterminal are assigned probabilities of use. These probabilities are required to sum up to 1.0 for each nonterminal. Probability of generating a string, given a grammar G , is the product of the probabilities of productions taken at each branch of its parsing tree. If there is more than a single parsing tree, a summation over all parsing tree probabilities must be performed. A simple grammar that generates strings of the form a^n is given below.

$$\text{Start} \rightarrow a \text{ (0.6)} | a \text{ Start (0.4)}$$

For the above grammar, the two-symbol string a^2 has the probability $0.4 \times 0.6 = 0.24$ which can also be seen as $P(\text{Start} \rightarrow a \text{ Start} | \text{Start}, G) \times P(\text{Start} \rightarrow a | \text{Start}, G)$.

A PCFG constructed from the interaction sequences basically has the ability to reproduce these patterns according to the probabilities integrated into the grammar. These probabilities are inferred from the training data containing these patterns.

To apply PCFGs to a specific application, two problems must be solved. The first problem is the choice of meaningful terminals. In other words, we need to define the alphabet whose letters will be used to record interaction sequences. For a meeting setting for example, the attendant tag identifiers could be such tokens. In some applications, however, a more meaningful token could be a category to which an attendant tag holder belongs, instead of the specific id of this attendant (e.g. *meeting director* instead of *John Doe*).

Second problem is the extraction of sequences. It is not easy to know what is an appropriate time frame to

collect a sequence and conclude that it is a meaningful pattern. Such a decision depends on the application domain, for example, if we try to model what set of interactions occur for a certain task in a company, the sequence begins from the time the request for the task is submitted. Another example can be given in a meeting scenario, if we would like to model the interactions of the meeting organizer with the meeting attendants, we can either mark the beginning and end of this duty, or separate the interactions into meaningful bursts.

2.2 Automated PCFG Construction

In our previous work [3], we have described in detail how a PCFG can be constructed given a set of sentences (hence strictly from the positive data that show only the correctly formed sentences). This algorithm was an extension of the works done in [4] and [6] with improvements on the time complexity. Although we will not go into details of the grammar inference algorithm in this paper, we will summarize its methodology.

Inference algorithm consists of two stages: (i) data incorporation, and (ii) application of operators. In the first stage, all sentences are introduced to the initial grammar as rules of the *START* nonterminal and probabilities are assigned according to the sentence frequencies. Each terminal symbol (token) is introduced to the PCFG by a nonterminal symbol which has a single rule (that terminal symbol) which is selected with probability 1.

In the second stage, the grammar is generalized and made more compact using two operators:

- **Chunking** that creates a new nonterminal which is assigned a string of nonterminals and which replaces all the occurrences of this string in other productions with that new nonterminal. Frequency of this nonterminal (hence its single rule) is set to the number of replacements made,
- **Merging** that creates a new nonterminal defined as a combination of two nonterminals. The right hand sides of productions of both nonterminals form the productions of this new nonterminal and probabilities are assigned according to their respective frequencies. The merged two nonterminals are removed and occurrences of any of these nonterminals are replaced by this new nonterminal.

Since grammar inference is a search for operands for two possible operations, an evaluation method is needed to measure the quality of a grammar which results from an application of each possible operation. A Bayesian posterior probability $P(G|D)$ of the grammar G given the data D is used for this purpose is defined as:

$$P(G|D) = \frac{P(G)P(D|G)}{P(D)}. \quad (1)$$

For maximization purposes, $P(D)$, the probability of the given training data D can be omitted from the formula above. $P(G)$, the probability of grammar G is calculated by using the length of grammar description, $l(G)$. The simple description method proposed in [3] allows for restricting the search space for operand of a possible chunk operation to strings of length at most 5. $P(D|G)$, the probability of training data D given the grammar G , is calculated as the product of probabilities of separate sentences (d_i) in the training data, so

$$P(G) = 2^{-l(G)} \quad \text{and} \quad P(D|G) = \prod_{i=1}^{|D|} P(d_i|G).$$

Formulation of $P(D|G)$ as above helps the algorithm to avoid re-parsing after merging operation and also reduces the search space for the operands of merging operation. In [3] we established that the time complexity of the algorithm is $O(D^2 \log(D))$.

3 Results

We have used Mission Survival Corpus 1 (MSC-1) dataset [2] to show the applicability of our methods. This dataset basically contains time-stamped annotations of 11 meetings of people deciding on how to proceed in a disaster scenario. The annotation includes the *socio role label* and *task role label* of each meeting attendant as well as an indication of who speaks at the time-stamp. Socio roles (*supporter*, *protagonist*, *attacker*, *neutral*) mainly represent the attendant's attitude toward the group's function while the task roles (*giver*, *seeker*, *orienteer*, *neutral*) represent the individual's function and technical skills. Although the role names are self-explanatory, the interested reader can look up detailed role descriptions in [2]. Furthermore, we will give brief explanations while we are presenting our results.

In the MSC-1 dataset, we modeled two sets of action patterns (which we also call the first and second metric):

- **Which roles go together:** Given a social role (socio label) of an attendant, what task roles are taken (sequence-wise) by this attendant during this social role? Or vice versa,
- **Attendants of which roles are speaking while a certain role is undertaken:** When an attendant of a meeting plays a certain social role, what are the social roles of the other attendants that speak (hence we assume, interactions while the attendant plays) during this role? The same question is valid for the task roles.

The modeling results for the first set (*Which roles go together*) can be seen in the grammars in Figures 1-4. The second set (*Who Speaks while a Certain Role is Undertaken*) is presented in Figures 5-8.

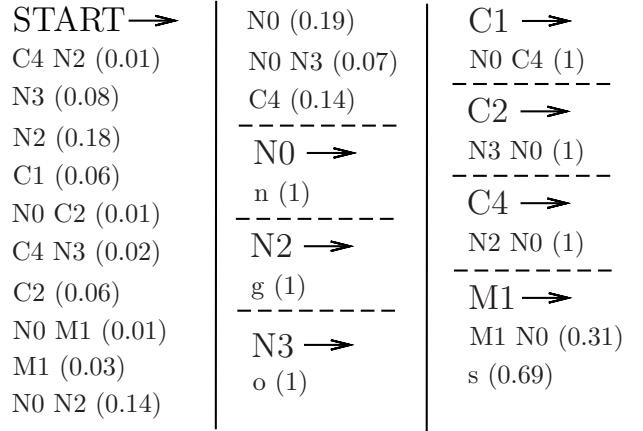


Figure 1: PCFG for Modeling ***Which Roles Go Together*** for Supporter Socio Label.

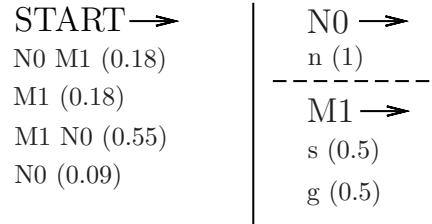


Figure 2: PCFG for Modeling ***Which Roles Go Together*** for Attacker Socio Label.

Figure 1 gives the grammar constructed to model first metric (*Which roles go together*) for the *supporter* socio label. The supporter has a cooperative attitude and, furthermore, provides resource support to other attendants. This mentality is clear in the given grammar. Orienteer (*o*) and giver (*g*) terminals represent mainly the supporting attitude of the attendant. Among these, giver is probability-wise the most dominant task-label that can be seen in the grammar. It can also be observed that seeker (*s* in the nonterminal *M1*) is another task-label that can appear in the supporter interaction sequence. This basically represents the cooperative attitude, since the conversation (cooperation) between two individuals in a meeting proceeds by questions (*s - seeker terminal*) and answers (*g - giver terminal*).

The grammar that models first metric for the *attacker* socio label can be seen in Figure 2. As the name suggests, the attacker *attacks* the opinion of another attendant. Such a behavior can be definitely observed in the grammar since both seeker (terminal *s*) and giver (terminal *g*) has equal probability (as given by *M1*). Such information can be easily observed thanks to the concise representation provided by the PCFG that is automatically constructed from the dataset.

Next, let us examine the grammar in Figure 3 constructed for modeling the first metric for *giver* task label. A giver basically provides factual information.

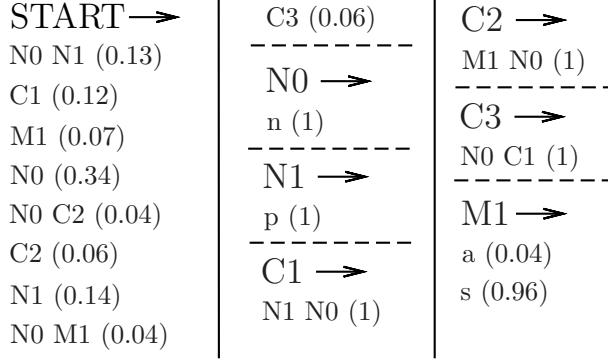


Figure 3: PCFG for Modeling **Which Roles Go Together** for Giver Task Label.

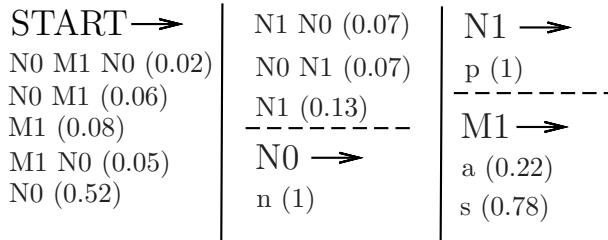


Figure 4: PCFG for Modeling **Which Roles Go Together** for Seeker Task Label.

By the nonterminal $M1$, a supporter (s) is much more dominant than an attacker (a) in this situation. The probability of a giver being a protagonist (the attendant that drives the conversation, denoted by p in the grammars) is quite high (~40%). Please note that although such results can be obtained through statistical means, grammars provide much more compact and manually examinable complete information about the meeting data. Furthermore, certain grammar processing algorithms (e.g. backward chaining from the terminal to the $START$ nonterminal) can be utilized to estimate the probabilities with which certain terminals occur in the sentences as well as the transition probabilities of symbols [5]. This way, such analysis of behaviors from the grammars does not have to be manual at all. Some extra information about the giver grammar is the fact that protagonist and attacker or supporter do not ever occur together. This means that the protagonist socio-label does not usually change its opinion and then becomes the supporter or attacker of another person. Instead, it is rather he who is attacked or supported. Furthermore, an attacking or supporting giver does not try to take control of the conversation. The grammar's power comes into attention in such situations, because such comments require the examination of sequences.

Figure 4 shows the grammar that models the first metric for the *seeker* task label. Seeker asks questions. However, from nonterminal $M1$ (created by a merge operation), we see that a person asking a question is

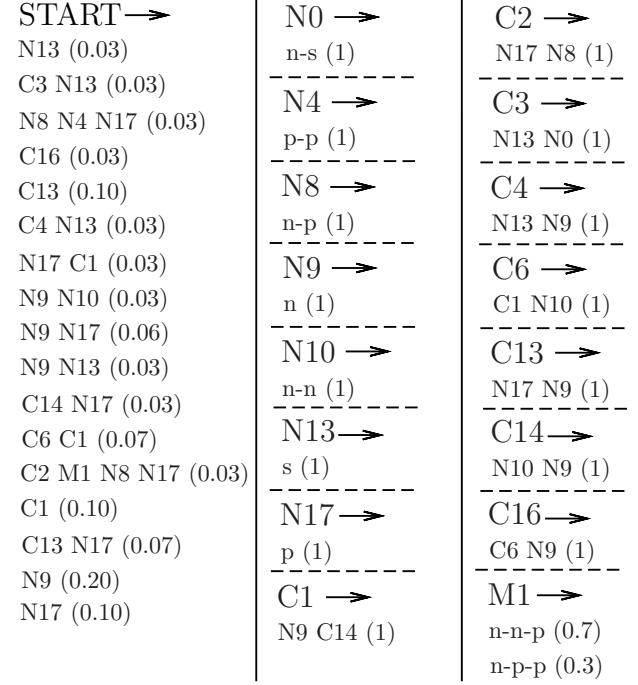


Figure 5: PCFG for Modeling **Who Speaks while a Certain Role is Undertaken** for Supporter Socio Label.

more likely to be a supporter (s) than an attacker (a). The most dominant socio-label is the neutral (n). The next one is a supporter who is the main socio-role taken by seekers. It is also remarkable any significant socio-label (i.e. not neutral) is either followed or preceded (mostly) by the neutral socio-label. This shows two things. For the proceeding case, we see that seeker either gets satisfied or his supporting status ends after his confirmation question. For the preceding case however, it can be concluded that for the annotators (the people that annotated this data by manually observing attendants), it takes a while to understand whether a person asking question is a supporter or attacker. Also, it can be seen that the seeker could be a person who answers his own questions, or drives the conversation by his questions, hence acting as the protagonist (p).

Using the grammar in Figure 5, let's examine who speaks while an attendant undertakes the socio label role of *supporter*. As expected of a supporter, the other attendants that speak while a supporter plays his role is mostly protagonists (p). For example, from the $START$ nonterminal, it can be seen that $N17$ is a really (probabilistically) dominant nonterminal which points to a single speaker, protagonist (p). We can see that there can be multiple protagonists speaking at the same time, given by the nonterminal $M1$ (e.g. $n-p-p$ means a neutral and two protagonists speaking at the same time). However, even from that nonterminal, not looking at the rules of $START$ nonterminal, we can see that it has

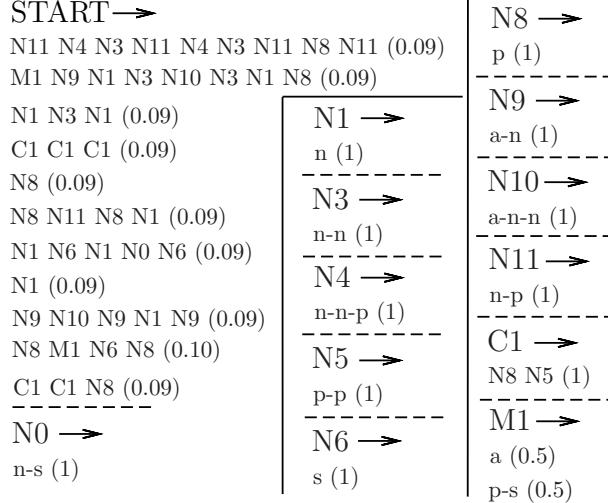


Figure 6: PCFG for Modeling ***Who Speaks while a Certain Role is Undertaken*** for Attacker Socio Label.

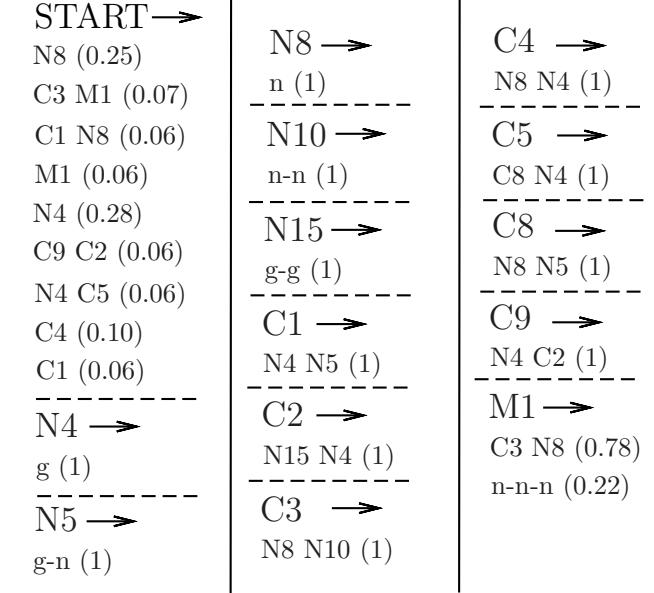


Figure 8: PCFG for Modeling ***Who Speaks while a Certain Role is Undertaken*** for Seeker Task Label.

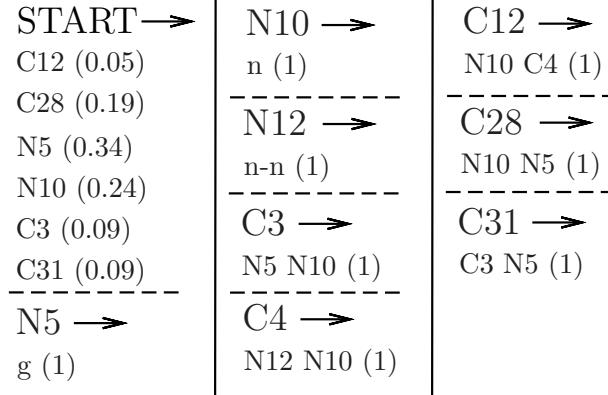


Figure 7: PCFG for Modeling ***Who Speaks while a Certain Role is Undertaken*** for Giver Task Label.

a lower probability as compared to a single protagonist. This is given by the probabilities of rules being 0.3 to 0.7 in the nonterminal *M1*.

Figure 6 shows the grammar that models the second metric (*Attendants of which roles are speaking while a certain role is undertaken*) for the attacker socio label. It is worthwhile to notice in the attacker’s grammar that it is either another attacker (*a*) or a protagonist (*p*) who speaks. A protagonist can speak with a supporter (*s*) at the same time (as in nonterminal *M1*). However, we do not see a supporter and an attacker speaking at the same time. From this information, we can say that arguments take turns. What may be even more interesting is the fact that we do not see a protagonist and another attacker speaking in the same sentence. This is an important finding which strengthens our deduction that people with opposite ideas take turns.

For Figure 7, we can briefly state that the only sig-

nificant speaker during a *giver*’s turn is another giver, this means a cooperative information providing. While constructing the grammars, we remove the rules with a probability below 0.001, which we assumed are insignificant. In the original (uncut) grammar, we were able to see a seeker (*s*) as well as orienteer (*o*) when somebody played a role of a giver in the meeting. These rules were removed when we set the threshold for presentation purposes.

Finally, the second metric grammar for the *seeker* task label is given in Figure 8. A seeker asks questions; hence it is only natural to see that these questions are satisfied by a giver (*g*) who provides factual information. This can also be seen in the grammar as the only significant terminals are including givers. Among the deleted rules (original grammar), we were able to see orienteers and seekers (since an orienteer can also contribute to solving problems by organizing meetings, and a seeker can be the further questioner), but these were removed in the thresholding process.

4 Previous Work

The field related to our work is group behavior modeling, e.g., face-to-face group meetings [7], conversation between patients and caregivers [8] etc. A large body of research aims at modeling *physical* interaction phenomena, such as utterances, gestures, eye gaze, head position and body motion, from multimodal sensors streams. Instead of providing straight-forward semantic meaning of the behavior, most of these outcomes serve as higher level features for algorithms attempting to infer semantic interactions between individuals. For example, [9] discussed an approach to estimate who

was talking to whom based on head positions of the participants. [10] used gazes, head gestures, and utterances to determine interactions regarding who responds to whom in multiparty conversations. [11] proposed a classifier to recognize agreement or disagreement utterances, utilizing both word-based and prosodic cues. In [12], audio and body motion features are used to assess interest and attraction level in conversational dyads.

By taking the individual interaction patterns into account, there are quite a few papers focusing on recognizing the group activities as a whole. [13] investigated a statistical framework for automatic meeting analysis based on various hidden Markov models. A set of meeting actions (including monologue, presentation, whiteboard etc.) are defined based on turn-taking patterns. A range of audio-visual features (e.g. speech activity, pitch, speaking rate and head and hand blobs) from each participant are extracted and combined to investigate the group-wise nature of the actions. More recent work is presented in [14] by utilizing a two-layer HMM framework. [15] introduced a dynamic Bayesian network based model, characterized by multiple stream processing to segment the group states similar to [13], but with features including prosody from audio signals, speaker activity and lexical features from textual transcripts. [16] defined the group states as discussion, presentation and briefing. A decision tree classifier was trained to estimate group states using four frequency-based features (including number of overlaps in speech, number of changes of speakers, number of participants who had spoken and average length of overlaps). By defining the states as presentation, discussion and break, [17] presented event-driven multilevel dynamic Bayesian networks (DBNs) to perform online detection of multilevel events, which coordinates both the multicue-based bottom-up reasoning and context-based top-down guidance.

In addition to modeling through interactions, the work on automatically identifying the roles (including functional, social and dominating) in a group has drawn a lot of attention. [18] used Bayesian methods to recognize functional roles (e.g., the anchorman, the second anchorman, the guest) in broadcast data. [19] recognized roles in movies (e.g., hero), based on social network analysis. By combining both the lexical feature and the social networks describing the interactions between the meeting participants, [20] discussed an approach to identifying the project manager, the marketing expert, the user interface expert and the industrial designer. [21] introduced a machine learning approach based on multiclass SVMs with radial based kernels to recognize roles such as giver, seeker and recorder in meetings by employing audio-visual features. [22] investigated a new framework for functional role detection based on the *influence model* [23], which takes into account the features of other participants. [24] proposed an DBN based approach for discovering influence in a

lounge where people played interactive debating games, which automatically determines how much influence a member has on others on a pair-wise basis. [25] and [26] described models for automatically detecting the team members who play a dominating role in a meeting using both support vector machines and a dynamic Bayesian network with a two-level structure.

Different from the above-mentioned methods, we present the utilization of grammars to model actions based on the social role and the interactions between group members, which provides straightforward but more detailed interpretation of the group behaviors.

5 Conclusions and Future Work

In this paper, we have proposed the use of PCFGs to model actions in the social network context. We made a brief overview on our automatic PCFG construction method, which takes as input the sequence of social entity interactions as well as the action patterns. This method provides the user with a PCFG which encapsulates the frequency of social actions. To evaluate the methodology presented, we have modeled social interactions as well as role taking in Mission Survival Corpus 1 [2]. From the modeling results, we can conclude that grammars are a concise way of storing social action patterns and help with inferences about the application domain's social properties.

Our future work will include the modeling of other application domains as well utilizing the constructed grammars for prediction purposes. For example, in a marketing scenario, the significant patterns that lead to an interest in an item can be modeled. Using an appropriate grammar processing method can help with formulating recommendations. Various other application domains can be listed where the PCFGs can be used for prediction, such as role recognition and performance evaluation of business processes.

Acknowledgment

This research was sponsored by US Army Research laboratory and the UK Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001 and under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors, and should not be interpreted as representing the official policies, either expressed or implied, of the US Army Research Laboratory, the U.S. Government, the UK Ministry of Defense, or the UK Government. The US and UK Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

- [1] Wasserman, S., Faust, K., *Social Network Analysis: Methods and Applications*, Cambridge Univ Press, New York, 1994.
- [2] Pianesi, F., Zancanaro, M., Not, E., Leonardi, C., Falcon, V., Lepri, B., *Multimodal Support to Group Dynamics*, Personal and Ubiquitous Computing, vol. 12(3), January 2008, Page(s): 181-195.
- [3] Geyik, S. C., Szymanski, B., *Event Recognition in Sensor Networks by Means of Grammatical Inference*, IEEE INFOCOM 2009, Rio de Janeiro, Brazil, March 2009, Page(s): 900-908.
- [4] A. Stolcke (1994), *Bayesian Learning of Probabilistic Language Models*, Doctoral dissertation, Dept. of Electrical Engineering and Computer Science, University of California at Berkeley.
- [5] Stolcke, A., Segal, J., *Precise n-gram probabilities from stochastic context-free grammars*, in Proc. 32nd ACL, Page(s): 7479, Las Cruces, NM, 1994.
- [6] S. F. Chen (1996), *Building Probabilistic Models for Natural Language*, Doctoral dissertation, Dept. of Computer Science, Harvard University.
- [7] Wen Dong, W., Mani, A., Pentland, A., Lepri, B., Pianesi, F., *Modeling Group Discussion Dynamics*, submitted to IEEE Transactions on Autonomous Mental Development.
- [8] Chen, D., Wactlar, H., Malkin, R., Yang, J., *Detecting social interaction of elderly in a nursing home environment*, in ACM Transactions on Multimedia Computing, Communication and Application, October, 2006.
- [9] Stiefelhagen, R., Chen, X., Yang, J., *Capturing Interactions in Meetings with Omnidirectional Cameras*, International Journal of Distance Education Technologies 3(3), Page(s): 34-47, 2005.
- [10] Otsuka, K., Sawada, H., Yamato, J., *Automatic Inference of Cross-modal Nonverbal Interactions in Multiparty Conversations*, In Proc. ICMI 2007, Page(s): 255-262, 2007.
- [11] Hillard, D., Ostendorf, M., Shriberg, E., *Detection of Agreement vs. Disagreement in Meetings: Training with Unlabeled Data*, In Proc. HLT-NAACL, Page(s): 34-36, 2003.
- [12] Pentland, A., Madan, A., *Perception of social interest*, in Proc. IEEE Int. Conf. on Computer Vision, Workshop on Modeling Peopleand Human Interaction, Beijing, Oct. 2005.
- [13] McCowan, I., Gatica-Perez, D., Bengio, S., Lathoud, G., Barnard, M., Zhang, D., *Automatic analysis of multimodal group actions in meetings*, IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 3, Page(s): 305-317, 2005.
- [14] Zhang, D., Gatica-Perez, D., Bengio, S., McCowan, I., *Modeling individual and group actions in meetings with layered HMMs*, IEEE Trans. Multimedia, vol. 8, no. 3, Page(s): 509-520, Jun. 2006.
- [15] Dielmann, A., Renals, S., *Automatic meeting segmentation using dynamic Bayesian networks*, IEEE Trans. Multimedia, vol. 9, no. 1, Page(s): 25-36, Jan. 2007.
- [16] Banerjee, S., 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, Page(s): 2189-2192, 2004.
- [17] Dai, P., Di, H., Dong, L., Tao, L., Xu, G., *Group Interaction Analysis in Dynamic Context*, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, v.39, n.1, Page(s): 34-42, 2009.
- [18] Vinciarelli, A., *Role recognition in broadcast news using Bernoulli distributions*, in Proceedings of IEEE Int. Conference on Multimedia and Expo, Page(s): 1551-1554, 2007.
- [19] Weng, C., Chu, W., Wu, J., *Movie analysis based on roles social network*, In proceedings of IEEE International Conference on Multimedia and Expo, Page(s): 1403-1406, 2007.
- [20] Garg, N. P., Favre, S., Salamin, H., Hakkani-Tur, D. Z., Vinciarelli, A., *Role recognition for meeting participants: an approach based on lexical information and social network analysis*, in ACM Multimedia, 2008, Page(s): 693-696.
- [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.
- [22] Dong, W., Lepri, B., Cappelletti, A., Pentland, A., Pianesi, F., Zancanaro, M., *Using the influence model to recognize functional roles in meetings*, Proceedings of the 9th international conference on Multimodal interfaces, November 12-15, 2007, Nagoya, Aichi, Japan.
- [23] 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.

- [24] Basu, S., Choudhury, T., Clarkson, B., and Pentland, A., *Towards measuring human interactions in conversational settings*, in Proc. IEEE CVPR Int. Workshop on Cues in Communication (CVPR-CUES), Kauai, Dec. 2001.
- [25] Rienks, R., 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, 2005, Lecture Notes in Computer Science, volume 3869, Page(s): 76-86, 2006.
- [26] Rienks, R., Zhang, D., Gatica-Perez, D., Post, W., *Detection and Application of Influence Rankings in Small Group Meetings*, In Proceedings of International Conference of Multimodal Interfaces, ICMI 06, 2006