INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 4 ISSUE 5 (2018) PAGES 4054 - 4062
RECEIVED : 20.09.2018 PUBLISHED : 24.10.2018

Oct 24, 2018

Tree-Based Interaction Pattern Mining for Human Interaction in Meetings

[1]Anette Regina.I, M.phil Research Scholar, Department of computer Science Science, Muthurangam Government Arts College, Vellore District

[2] Janani.K, Associate Professor, Department of computer, Muthurangam Government Arts College, Vellore District

ABSTRACT:

Mining Human Interaction flow in meetings or general representation of any interaction face to face to meetings is useful to identify the person reaction in dissimilar situation. In this paper, we propose face-to-face meetings for mining method to extract frequent interaction patterns of human interaction. Discussion with Human interaction Meetings, such as proposing an idea, giving comments, and expressing a positive opinion, Negative opinion indicate user intention toward a topic or role. Human interaction flow in a discussion meeting is represented as much a tree. Tree based interaction pattern mining algorithms are designed to analyze the structures of the trees and to extract interaction flow mining patterns. The experimental results show that we can successfully extract several interesting interaction patterns that are useful for the interpretation of human behavior in meeting discussions, such as determining frequent pattern interactions, typical pattern interaction flows, and relationships between different types of interactions.

Keywords: Data mining, Knowledge Extraction, Decision making

I. INTRODUCTION

Face-to-face conversation is one of the most basic forms of communication in our life and is used for conveying/sharing information, understanding others' intention/emotion, and making decisions. The process of data mining is to extract knowledge from a dataset in a human understandable structure. In the societal dynamics such as person interaction is the one of the most important thing for considerate, how a human's behavior or human actions under the assembly and determining whether the assembly was well organized or not is one of the main issues in meetings. To further understand and interpret human interactions in meetings, we need to discover higher level semantic knowledge about them, such as which interactions often occur in a discussion, what interaction flow a discussion usually follows, and what relationships exist among interactions. This knowledge likely describes important patterns of interaction. We also can regard it as a grammar of meeting discussion. Data mining, which is a powerful method of discovering new knowledge, has been widely adopted in many fields, such as bioinformatics, marketing, and security. In this study, we investigate data mining techniques to detect and analyze frequent interaction patterns; we hope to discover various types of new knowledge on interactions.

Human interaction meeting flow in a discussion to represented as a tree pattern. The tree-based interaction pattern mining algorithms to analyze tree-based structures and extract interaction flow patterns. An interaction flow that appears frequently reveals relationships between many types of interactions. For instance, if one type of interaction pattern appears, what is the probability of another type following it? Human interactions mining are important for accessing and understanding meeting content for flow. First,

INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 4 ISSUE 5 (2018) PAGES 4054 - 4062
RECEIVED : 20.09.2018 PUBLISHED : 24.10.2018
Oct 24, 2018

the tree based mining results can be used for indexing table for meeting semantics, also existing meeting capture systems could use this technique as a smarter indexing tool to search and access particular semantics of the meetings. Next, the extracted patterns are useful for interpreting human interaction pattern in meetings. The researchers could use them as domain knowledge for further analysis of human interaction.

2. RELATED WORK

Tree based interaction in meetings has attracted much research in the fields of speech processing and human-computer interaction. Human interaction is one of the most significant individuality of group social dynamics in meetings. Unlike corporal interactions such as, turn taking and addressing the human communications are incorporated with semantics. Adopt a collaborative approach for capturing interactions [5] mostly focus on detecting physical interactions among participants without any relationships with topics. Therefore they cannot obviously conclude participant's attitude. Every increasing amount of recorded gathering data is motivating the need for the completion of tools to proficiently access and quickly retrieves significant pieces. The AMI project [8] was proposed for studying human interaction issues in meetings, such as turn-taking, gaze behavior, influence, and talkativeness. Otsuka et al. [9] used gaze, head gestures, and utterances in determining interactions regarding who responds to whom in multiparty face-to-face conversations.

DiMicco et al. [14] presented visualization systems for reviewing a group's interaction dynamics, e.g., speaking time, gaze behavior, turn-taking patterns, and overlapping speech in meetings. In general, the above-mentioned systems aim at detecting and visualizing human interactions in meetings, while our work focuses on discovering higher level knowledge about human interaction. To discover the frequent patterns in a tree H. Aoyama [1] introduces a novel algorithm to discover all frequent patterns subtrees in a tree-plant with a novel data structure called scope-list. In this scheme TREEMINER through a pattern matching tree mining algorithm were contrasted by H. Aoyama. It also present a relevance of tree mining to examine real web logs for usage patterns. In earlier work XSpanner [6] systematically expand the two algorithm pattern growth methods for drawing out frequent tree patterns.

3 HUMAN INTERACTION AND INTERACTION FLOW

Discovering semantic information is important for sympathetic and interpret how people can act together in a meeting discussion. Capturing all of this informal assembly information has been a topic of research in several communities over the past decade. In this paper describe the human interaction flow and discovering frequent patterns of human interaction in meeting discussions at same meeting by using graph based substructure frequent. The removal results would be useful for summarization, indexing, and comparison of meeting records. Human interactions in a meeting discussion are defined as social behaviors or communicative actions taken by meeting participants corresponding to the current topic

3.1 Human Interaction

Human interaction categorizes obviously vary according to the practice of the meeting or the type of the meetings. In this investigation mainly focus on the face to face meetings that are online purchase through ecommerce interactions. For a numerical structure for conceptualizing, investigate and scheming interaction flow, generate a set of interaction types based on category scheme propose, comment, acknowledgement, requestInfo, askOpinion, posOpinion, negOpinion. The detailed meanings are as follows: propose a user proposes an idea with respect to a topic; comment—a user comments on a proposal, or answers a question; acknowledgement—a user confirms someone else's comment or explanation, e.g., "yeah," "uh huh," and "OK;" requestInfo—a user requests unknown information about a topic; askOpinion-a user asks someone else's opinion about a proposal; posOpinion—a user expresses a positive opinion, i.e., supports a proposal; and negOpinion—a user expresses a negative opinion, i.e., disagrees with a proposal.

3.2 Human Interaction Flow

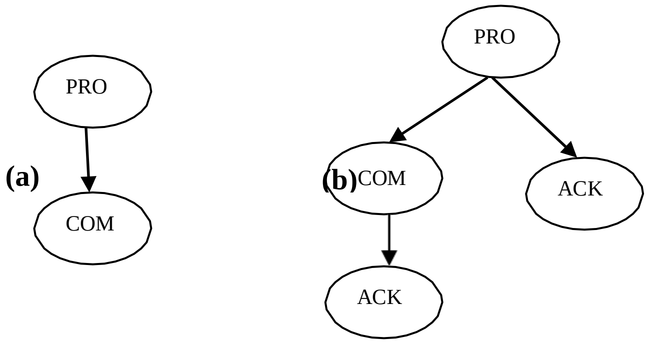


Fig. 1. Tree representation of interaction flow.

Human interaction flow in face to face meeting is premeditated as tree .An interaction flow is a list of all interactions in a conversation with the relationship between them. An interaction flow is a list of all interactions in a conversation session with triggering relationship between them. L = {PRO; COM; ACK; REQ; ASK; POS; NEG} Labels are abbreviated names of interactions, i.e., PRO-Propose, COM—Comment, ACK—acknowledgement, REQ— request Info, ASK—askOpinion, POS—posOpinion and NEG-negOpinion .Examples of interaction trees shown in the figure. Fig. 1b, a PRO triggers two interactions, COM and ACK (specifically COM is performed earlier than ACK), and then an ACK responses to the COM interaction. Labels in the interaction tree are not sorted, because the edges reflect temporal relationship between the siblings. Hence, sorting, e.g., alphabetically, would likely break this relationship.

4. TREE-BASED INTERACTION PATTERN

4.1Tree Based Pattern Mining Algorithm

Human interactions such as propose an idea, generous remarks and expressing a positive opinion, point out user intention in the direction of a theme or role in a discussion. Human interaction flow in a conversation session is represented as a tree. Treebased relations mining algorithms are considered to examine the structures of the trees and to extort interaction flow patterns. Designed a tree based pattern mining algorithm for interaction flow mining.

For instance, the tree PRO-COM *ACK can be transformed into two trees, PRO-COM *ACK and PRO-ACK8COM. If the siblings of a tree are the same (e.g., PRO-COM*COM), this process can be omitted. The purpose of generating isomorphic trees is toease string matching. It formulate the frequent tree pattern mining algorithm for every node in the tree .For every tree in TD the algorithm first interactions the places of siblings to produce the full set of isomorphic trees (ITD). The principle of generating isomorphic trees is

INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 4 ISSUE 5 (2018) PAGES 4054 - 4062
RECEIVED : 20.09.2018 PUBLISHED : 24.10.2018

to easiness string matching. Following generate the isomorphic trees then calculates support values of alltrees at Steps 2-3. In Step 4, it selects the trees whose supports are larger than and detect isomorphic trees inside them. If m trees are isomorphic, it selects individual of them and rejects the others. It finally outputs all frequent tree patterns with respect to σ.

Algorithm 1: FITM (TD, σ) (Frequent interaction tree pattern mining)

Input: Tree database (TD) and a support threshold σ

Output: Frequent tree configurations with respect to σ

Procedure:

(1) Scan database TD and generate its full set of isomorphic trees (ITD)

(2) Scan database ITD and count the numeral of occurrences for each tree t

(3) Calculate the support of each one tree

(4) Select the trees whose supports are higher than σ also detect isomorphic trees; if m trees are isomorphic, select one of them and discard the others.

(5) Output the frequent trees

Where, TD: A dataset of interaction trees in the flow, ITD: The full set of isomorphic trees to TD

T:Tree, t^k: The subtree with k nodes, i.e K-subtree ,

C^k: Candidate set with k-nodes.

F^k: Frequent set with k-subtrees σ -support threshold minsup

(Pattern). Patterns are frequent trees or subtrees in the tree database.

(Support). Given a tree or subtree T and a data setof trees TD, the support of T is defined as

$$Supp(T)=\frac{\text{Number of occurrences of T}}{\text{Total number of tree in TD}}$$

4.2 Frequent Interactiion Subtree Pattern Mining

It primary calculates the support of every node and selects the nodes whose supports are larger than σ to form the set of frequent nodes, F^1 from Steps 2-3. It then adds a frequent node to accessible frequent i-subtrees to make the set of candidates with i + 1 node at Steps 4-8. If there are any trees whose supports are larger than , it selects them to form F^{i+1} and repeats the procedure from Step 4, otherwise it stops to output of frequent subtrees. In Step 7we join t^i and t^1 to generate the candidate subtree set of size. First, t1 can be joined as the parent of the root of the original tree t^i. Fig. 3a shows this case. Then, t^1 may be placed as one child of each node of t^i as depicted. Step 8 calls a subprocedure (Subtree_Support_Calculating) to calculate the support of each tree in C^{i+1}.

Algorithm 2: FISTM (TD,σ) (Frequent interaction subtree pattern mining)

Input: Tree database (TD) and a support threshold σ

Output: Frequent subtree configurations with respect σ to

Procedure:

(1) i<-0.

(2) Scan database TD, calculate the support of each node.

(3) Select the nodes whose chains are longer than σ to form $F^{1.}$

(4) i<- i +1 .

(5) For each tree t^i in F^i, do .

(6) For each node t^1 in F^1, do.

(7) Join t^i and t^1 to generate C.

(8) Subtree Support Calculating (TD; t^{i+1})//calculate the support of each tree in C^{i+1}

(9) if there are any trees whose supports are larger than ,then select them to form F^{i+1} and return to Step (4).

(10) Else output the recurrent subtrees whose supports are higher than σ.

4.3 Subtree_Support_Calculating Method

The subprocedure, Subtree_Support_Calculating, first creates subtrees of each tree t in TD with the size of subtrees the same as that of st (Steps 3-4). Then, for each subtree of t, it generates its isomorphic trees and compares their string codes with that of st. If it matches, the number of occurrences of st is increased by 1 (Steps 5-14). It finally calculates and returns the support of st (Steps 15-16).

Subprocedure. Subtree Support Calculating (TD, st)

```
(1) count ← 0
(2) supp(st) ← 0
(3) for each tree t € TD do
(4)      create subtrees S of t with any item s € S;
(5)      flag false
(6)           for each item s € S do
(7)                generate isomorphic trees IS of s
(8)           for each item is € IS do
(9)           if tsc(is)=tsc(st) then
(10)                count ← count + 1
```

INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 4 ISSUE 5 (2018) PAGES 4054 - 4062
RECEIVED : 20.09.2018 PUBLISHED : 24.10.2018
Oct 24, 2018

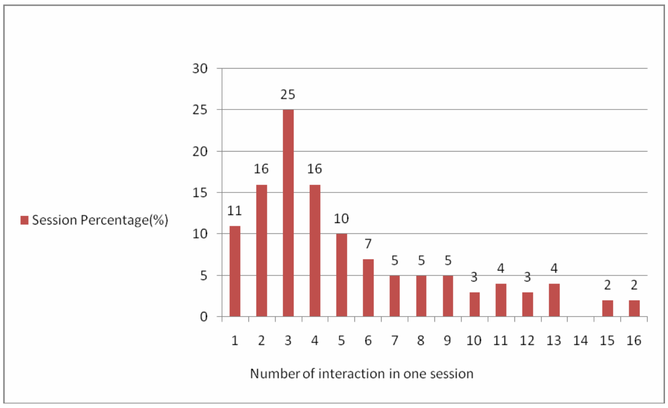
```
                        flag ← true
                        break
                if flag = true then
                 break
        supp(st) count/TD
return supp(st)
```

5. RESUTS AND DISCUSSION

5.1 Data Set

Our data set four real meetings, lasting 15 minutes on average. Multiple devices, such as video cameras, microphones, and motion sensors, were used for capturing the meetings. The four meetings include one PC purchase meeting (22 min, discussing PCs to be ordered for the laboratory, such as types, configuration, size, weight, manufacturer, etc.), one trip-planning meeting (14 min discussing time, place, activities, and transportation for a summer trip), one soccer preparation meeting (20 min, talking about the players and their roles and positions in an upcoming match), and one job selection meeting (08 min, talking about factors that will be considered in seeking a job, such as salary, working place, employer, position, interest, etc.).

The longest sessions (those with the most interactions) are composed of 16 interactions. Sessions containing one to four interactions, which account for around 80 percent of all sessions, could be regarded as ordinary sessions. Other sessions, with six interactions, and especially those with 11 to 16 interactions, are considered special. They might be hot sessions in which participants engaged in emphatic and heated discussion. Actually hotspots in meetings are proofed to be associated with high degree of crossover of utterances and large number of back-channel feedbacks. The three longest (or hottest) sessions, with 20 interactions, are “PRO-PRO-ACK*(COM-(COMACK *(COM-COM-COM-(COM-COM -COM)*REQ))* COM),” “COM-(COM-REQ*(COM-ACK-COM-COM- (COM-COM-COM*(NEG-COM))*COM))* ACK,” and “COM-(COM-ACK*ACK*ACK*ACK*ACK)*ACK* represented with our tree string codes.



5.2 Results

From these patterns, we can observe that sessions were often started with comments, proposals, or questions. Although the most frequent tree is a one-interaction session, most of the frequent trees consist of two or three interactions. Within the six frequent trees, the interaction of comment appears five times whereas acknowledgement occurs four times, which means they are very common in meeting discussion.

INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 4 ISSUE 5 (2018) PAGES 4054 - 4062
RECEIVED : 20.09.2018 PUBLISHED : 24.10.2018

For frequent interaction subtree mining, we first examine the running time required for different values of support threshold. The running time decreases as the support threshold increases. In particular, when the support threshold increases from 1 to 2 percent, the running time decreases dramatically. It takes about 8.5 s to discover frequent subtrees when the support threshold is set to 1 percent.

Rule No	**Associate rule**	**Subs tree pattern**	**Sup p(A)**	**Sup p(C)**	**Supp(A→C)**	**Confidence**
1	RER-COM	2-3	0.3258	0.8230	0.2809	0.8622
2	COM-ACK	2-1	0.8230	0.6096	0.4916	0.5973
3	{COM-COM-COM}-ACK	4-1	0.1067	0.6096	0.0618	0.5792
4	{COM-COM}-ACK	3-1	0.2893	0.6096	0.1489	0.5147
5	{REQ-COM}-ACK	3-2	0.2809	0.6096	0.1348	0.4799
6	{COM-COM-COM}-COM	4-2	0.2067	0.8230	0.0506	0.4742
7	PRO-POS	2-4	0.3539	0.1882	0.1517	0.4287
8	REQ-{COM-ACK}	3-2	0.3258	0.4916	0.1348	0.4138

Table1 Association Rules

Setting the support threshold σ to 4 percent, we obtain 34 frequent subtrees, of which seven trees have one node, 12 have two nodes, 10 have three nodes, and five have fournodes. depicts the top five patterns for each number of nodes ranked by the value of support. The patterns of one node show the distribution of different interaction types. The comment interaction appears the most frequently. The interactions of acknowledgement and propose are also very common. The most infrequent interactions are askOpinion and negOpinion, which are not included in the top five. The two-node patterns are basically combinations of one-node patterns.

6. CONCLUSION

This paper have proposed an interaction based tree mining for human interaction flow from face to face meetings , They also can be used for interpretation of human interaction in meetings. In the future, we will develop several applications based on the discovered patterns. We also plan to explore embedded tree mining for hidden interaction pattern discovery. Embedded subtrees are a generalization of induced subtrees, which allow not only direct parentchild branches, but ancestor-descendant branches . For example, when there is an interaction of propose, there always follows a comment, directly or indirectly. Finally, we plan to incorporate more meeting content in both amount and category. The current meetings are all task oriented. It is valuable to capture various categories of meetings for analysis such as panel, debate, interview, etc. There would be some differences in the frequent interaction patterns for different meeting styles.

REFERENCES

[1] Z.W. Yu, Z.Y. Yu, H. Aoyama, M. Ozeki, and Y. Nakamura, “Capture, Recognition, and Visualization of Human Semantic Interactions in Meetings,” Proc. Eighth IEEE Int’l Conf. Pervasive Computing and Comm. (PerCom ’10), pp. 107-115, Mar.-Apr. 2010.

[2] Q. Yang and X. Wu, “10 Challenging Problems in Data Mining Research,” Int’l J. Information Technology and Decision Making, vol. 5, no. 4, pp. 597-604, 2006.

[3] M.J. Zaki, “Efficiently Mining Frequent Trees in a Forest: Algorithms and Applications,” IEEE Trans. Knowledge and Data Eng., vol. 17, no. 8, pp. 1021- 1035, Aug. 2005.

[4] C. Wang, M. Hong, J. Pei, H. Zhou, W. Wang, and B. Shi, “Efficient Pattern-Growth Methods for Frequent Tree Pattern Mining,” Proc. Pacific- Asia Conf. Knowledge Discovery and Data Mining (PAKDD ’04), pp. 441- 451, 2004.

[5] Christian Becker and Zhiwen Yu, ‘Tree-Based Mining for Discovering Patterns of Human Interaction in Meetings,’ Proc. Eighth IEEE Int’l Conf. Knowledge Discovery and Data Mining (PAKDD’10), pp. 107-115,2012.

[6] M. Barnard, S. Bengio,D. Gatica-Perez, G. Lathoud, I. Mccowan, D. Zhang, ‘Automatic Analysis of Multimodal Group Actions in Meetings,’ IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 27, no. 3, pp. 305-317,2005.

[7] P. Chou, R. Hegde, S. Junuzovic, Z. Liu, C. Zhang, and Z. Zhang, ‘Requirements and Recommendations for an Enhanced Meeting Viewing Experience,’ Proc . ACM Int’ lConf. Multimedia, pp. 539- 548,2008.

[8] A. Nijholt, R.J. Rienks, J. Zwiers, and D. Reidsma, "Online and Off-Line Visualization of Meeting Information and Meeting Support," The Visual Computer: Int'l J. Computer Graphics, vol. 22,no. 12, pp. 965-976, 2006.

[9] K. Otsuka, H. Sawada, and J. Yamato, "Automatic Inference of Cross-Modal Nonverbal Interactions in Multiparty Conversations," Proc. Int'l Conf. Multimodal Interfaces (ICMI '07), pp. 255- 262, 2007.