Journal of Universal Computer Science, vol. 15, no. 12 (2009), 2311-2329
submitted: 15/12/08, accepted: 25/6/09, appeared: 28/6/09 © *J.UCS*

Meeting Warming-up: Detecting Common Interests and Conflicts among Participants before a Meeting

Zhiyong Yu
(School of Computer Science, Northwestern Polytechnical University, Xi'an, P. R. China
yuzhiyong@mail.nwpu.edu.cn
and
Academic Center for Computing and Media Studies, Kyoto University, Japan)

Zhiwen Yu, Xingshe Zhou
(School of Computer Science, Northwestern Polytechnical University, Xi'an, P. R. China
{zhiwenyu, zhouxs}@nwpu.edu.cn)

Daqing Zhang
(Institut TELECOM - TELECOM & Management SudParis, France
Daqing.Zhang@int-edu.eu)

Yuichi Nakamura
(Academic Center for Computing and Media Studies, Kyoto University, Japan
yuichi@media.kyoto-u.ac.jp)

Abstract: In order to boost both efficiency and effectiveness of meetings, we propose a novel Meeting Warming-up system to detect common interests and conflicts among participants before a meeting. The basic idea of the proposed approach is: firstly, modelling user preference by extending the attribute concept tree with additional relations both in and across attributes; secondly, determining common interests and conflicts through preference propagation and merging; thirdly, visualizing the detected results via a group preference graph. As a result, each participant can intuitively understand the group's opinions as a whole and warm up for discussions around potential outcomes. In particular, the meeting may have an easy and friendly start with commonly agreed outcomes, the commonly disagreed items may be ignored to save time, and participants may be mentally prepared to discuss the possible conflicts carefully and sufficiently. The experimental results showed that our approach is feasible.

Keywords: Smart meeting, group dynamics, user preference, preference propagation, visualization
Categories: H.5.3

1 Introduction

The smart meeting room is a typical example of intelligent environments [Yu, 09]. It aims to facilitate human-human and human-computer interaction, and eventually improve meeting productivity. As [Huang, 04] pointed out that performing a preference task [McGrath, 84], in which participants make their decisions purely based on personal preferences, GSS (group support system) is not a good choice because it can dampen desired exchange of personal preferences and values. In this case, face-to-face meeting with ubiquitous support could be a better choice.

However, most of preference task meetings are either not efficient or not effective [Yu, 09] [Romano, 01]. [Tab. 1] lists two common meeting styles as well as the comparison with our system, with descriptions as follows:

- *Meeting Style 1*: Since different people are likely to have different or even conflicting personal preferences, if the floor is open for discussion, it would require more exchanges to reach a consensus or compromise, so-called low efficiency. Also fierce debates undermine group cohesiveness [Shaw, 71].
- *Meeting Style 2*: Besides reaching a group decision, some meetings prefer to maintain group cohesiveness. Thus, meeting participants often hide their own opinions, e.g., succumb to the authority or the majority, which may leads to group thinking [Nass, 04], so-called low effectiveness.
- *Meeting Warming-up*: We expect that the group dynamics can benefit from the exposition of user preferences, the same as social behavior feedbacks [DiMicco, 06] [Kulyk, 05] [Kim, 08]. Our system intends to achieve both high freedom of expression and group cohesiveness by enhancing mutual awareness [Jameson, 03], i.e., detecting common interests and conflicts among participants before a meeting.

	Freedom of Expression	*Group Cohesiveness*	*Time Efficiency*
Meeting Style 1	High	Low	Low
Meeting Style 2	Low	Medium	High
Meeting Warming-up	High	High	Medium

Table 1: Different results of different meeting styles

To show how the Meeting Warming-up system works, we take a trip-planning meeting in a temporary group with several (e.g., 3 to 15) participants as an example. Before the actual meeting, everyone knows nothing about others' opinions. The meeting organizer presents all related concepts according to the topic, such as the tourist attractions, the possible time slots, the choices of activities, various transportation means, etc. Extended with additional relations both in and across attributes, attribute concept trees [Schickel, 05] are used to model this domain knowledge. Then each participant separately points out some preferences as he likes with a graphical interface, e.g., giving scores for 10% of all concepts, which reduces the work load of participants. Because of the sparsity of explicit preferences, it is not enough to get the common interests and conflicts merely through these preferences. Meeting Warming-up can speculate a participant's unknown preferences according to his already known ones, and thus can determine the common interests and conflicts of the group, e.g., everyone like to travel by air, or someone like summer trip while others like winter trip. Finally, a group preference graph is presented to all participants for visualization. With this functionality, the organizer can design a dedicated agenda, and participants can warm up for discussion around potential outcomes.

In the rest of this paper, we first discuss the related work in Section 2. Then we introduce our preference model in Section 3, followed by the overview of our Meeting Warming-up system in Section 4. Sections 5 and 6 describe common interest and conflict detection and visualization, respectively. The experimental results are presented in Section 7. Finally, in Section 8, we conclude the paper.

2 Related Work

There has been a lot of work done on smart meeting room (such as AmI [Huang, 04], SCR [Jaimes, 05], EasyMeeting [Chen, 04], IMR [Mikic, 00], and MeetingAssistant [Yu, 07]), and a number of studies on user preference in the fields of personalized service [Kim, 07] [Yu, 06], e-commerce [Sarwar, 01] [Schickel, 06] [Aydogan, 07], etc. Next we discuss related work about group dynamics in smart meetings and ontology-based preference modelling.

2.1 Group Dynamics in Smart Meetings

Group dynamics, i.e., developing process of social relationships among meeting participants, have been recognized as a fundamental aspect of the meetings' efficacy since the seminal work of [Tang, 91]. It has been verified that the awareness of group dynamics is able to improve the effectiveness of interaction. For example, In [Pianesi, 08]'s study, meeting participants receive multimedia feedback on their relational behavior to increase self-awareness. [DiMicco, 06] developed a visualization system for turn-taking patterns in a face-to-face meeting. [Kulyk, 05] showed that visualizing the speaking time and gaze behavior can result in more balanced participation in a meeting. [Rienks, 06] determined who dominated a meeting based on the number of floor grabs and turn takes. Our work differs from these studies in several respects:

First, we deal with different aspect of social dynamics in meeting, i.e., user preference, specifically what they agree on and how diverse their opinions are. It is at the semantic level of group dynamics. Previous works, such as [Pianesi, 08], [DiMicco, 06], [Kulyk, 05], and [Rienks, 06], focus on physical-level group dynamics in terms of turn-taking, speaking time, gaze behaviour, and so on. We think semantic-level group dynamics is the inner reason of physical-level.

Second, our study offers pre-meeting support and uses the result as assistance for meeting discussion, while most other studies address on in-meeting (e.g., [Kulyk, 05]) or post-meeting (e.g., [Pianesi, 08], [DiMicco, 06], and [Rienks, 06]) support. As a result, our system does not suffer the reflection delay as others.

In addition, a conflict avoiding meeting assistant [Kernkamp, 06] was developed to help the chairperson avoid unwanted conflicts during a meeting. However, detecting potential conflicts without a chairman before a real conflict occurs is not resolved. Our work explores this issue.

2.2 Ontology-based Preference Modelling

User preferences have been widely studied in fields other than smart meetings. We use an ontology-based approach to model user preferences that is also adopted by other researchers and verified superior to flat preference model. Among others, [Kim, 07] proposed a user preference description language constructed by various domain ontologies. [Thomopoulos, 04] concluded that the use of viewpoints allows one to simplify user interface on the "kind of" ontology. [Middleton, 01] showed that using an ontology during the profiling process of paper classification outperforms using a flat list of topics. However, these ontology-based models seldom support preference propagation or merging. And they all aim at *items* instead of *features* (possible values of an item's attributes), with further discussion in Section 3.1.

Propagation means inferring the missing preferences from explicitly given ones. It is usually used to restore the uncomplete user preference. Topics in different meetings vary. It is not practical to collect opinions of a person on all topics. Thus predicting methods based on previous choices, i.e., collaborative filtering [Sarwar, 01] [Chen, 05] cannot be used. Preference propagation based on already builded knowledge such as domain ontology can avoid suffering this situation. There are various methods of propagation across different concepts in the ontology. In heterogeneous attribute utility model [Schickel, 05], the structure of user preference is assumed to follow an ontology of product attributes, which provides an inductive bias that allows learning of preference to succeed even with very few ratings. [Schickel, 06] proposed to fill in missing elements of a user's preference using the knowledge captured in an ontology. [Aydogan, 07] estimated the relative distance in a taxonomy between two concepts using some intuitions of the structure. [Stefanidis, 06] discussed several ways to compute the appropriate score for a concept to which no explicit score is assigned. In a word, these studies relied on regular relations of ontology, i.e., "is a" or "part of" relation from a concept to its parent. Estimating semantic similarity is done by calculating the length of the path between the concept nodes [Schickel, 05] [Aydogan, 07] or cardinality of the set indicated by a concept [Schickel, 06] [Stefanidis, 06]. In our preference propagation method, besides these regular relations, we adopt additional relations between values of different attributes and non-uniform relations between siblings in the same attribute. With these new features, propagation can be conducted in broader condition and more precisely as verified by our experimental results.

3 Preference Model

The purpose of the preference model is to answer the questions such as "what do you prefer?" and "how much do you prefer it?". Thus, a preference statement specified by a user is in the form of (*concept*, *score*), where *concept* is the object of preference and *score* refers to the degree of preference. We basically follow the ontology-based preference model in existing researches [Schickel, 05] [Schickel, 06] [Aydogan, 07] [Stefanidis, 06]. The difference is that more expressive relations are incorporated in our model. The overview of our preference model is illustrated in [Fig. 1]. The domain ontology represents knowledge about the meeting topic, with three different kinds of relations between concepts. The preferences are aggregated by statements in which all concepts come from the domain ontology.

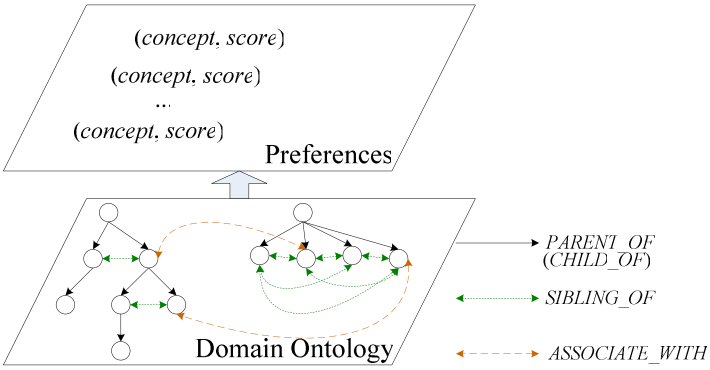

Figure 1: Ontology-based preference model

3.1 Object of Preference

For the object of preference, there are mainly two approaches: one is item based [Sarwar, 01] and the other is feature based [Schickel, 05]. An item is an element of the set of choice, also called object, outcome, or tuple. It has several attributes, and there are many possible values for an attribute. A feature is to describe which value a particular attribute holds. If preferences for certain features are stated accurately, multi-attribute decision theory [Keeney, 93] provides methods to determine the preferred items even when the set of alternatives is extremely large or volatile. This approach does not suffer from cold start, latency, or scalability problems as the number of items increase [Schickel, 06]. Considering the openness of meeting discussion, we adopt the feature based approach.

The topic of a meeting indexes a domain, which contains all common knowledge that possibly referred to in the meeting. We use ontology to organize domain knowledge, because ontology is effective for representing both concepts and relations between concepts. This provides a common glossary and comprehension of this domain [Sarwar, 01], which are fundamental for interpersonal discussion.

The domain ontology consists of several attribute concept trees [Schickel, 05], in which each concept is a possible value that an attribute can hold. According to the hierarchical structure of the trees, there is an "is a" or "part of" relation from a child to its parent. Let *PARENT_OF* and *CHILD_OF* denote the relation from a concept to its child and to its parent respectively. These regular relations reflect the implicit knowledge of the person who wrote the ontology [Schickel, 06], and we can take advantage of it to conduct preference propagation. We assume that the concept trees already exist, e.g., collected from Open Directory Project (http://www.dmoz.org) or Word Net (http://wordnet.princeton.edu).

Besides these relations, we discovered that there are extra two types of relations which are almost ignored by existing studies. One is the non-uniform relation between siblings, because the proximities between any two siblings are not the same. Let *SIBLING_OF* denote the relation from a concept to its sibling. Another is the underlying relation between concepts that span across different attributes. For example, in the "Trip" domain, "Kyoto" in the attribute "Destination" and "Temple" in the attribute "Activity" are tightly correlated because the domain knowledge includes the information that there are many famous temples in Kyoto. We use *ASSOCIATE_WITH* to represent this type of relation.

3.2 Degree of Preference

Two common techniques are widely used to describe the degree of preference: quantitative and qualitative [Chomicki, 02]. The quantitative approach assigns each choice a score that reflects its desirability, while the qualitative approach describes the ordering of candidate choices directly, usually in pair wise fashion. A major problem with the qualitative approach is the difficulty for preference composition [Henricksen, 06]. Therefore, we choose the quantitative approach.

We use a score ranging from −1 to 1 to denote the degree of a person's preference about a particular concept. For instance, if a user states a preference like: (Summer, 0.8), it means he likes a trip in summer very much. Because the concept comes from the domain ontology, it is clear that "Summer" is the value of "Time" attribute.

4 Meeting Warming-up

With the preference model, we design the Meeting Warming-up system to detect and visualize common interests and conflicts among participants before a meeting. The architecture of the Meeting Warming-up system is depicted in [Fig. 2]. It consists of three layers: preparation, detection and visualization.

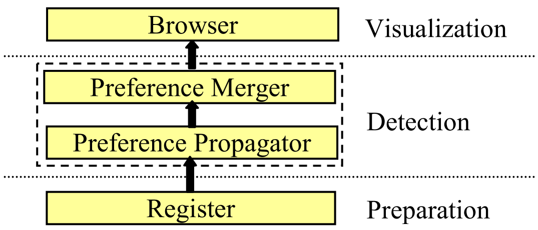

Figure 2: System architecture of Meeting Warming-up

In the preparation layer, domain knowledge is collected and constructed to form the base of preference model. *Seed preferences* (already known partial preferences) are the precondition of succeeding process. The register provides a GUI for meeting participants to input their seed preferences explicitly. As shown in [Fig. 3], users can easily select concepts from the hierarchical domain knowledge and rate them by using a scroll bar. In the detection layer, the preference propagator uses the seed preferences to infer the other unknown preferences through propagating in and across attributes, and then the preference merger merges the propagation results to generate a group user preference that contains the participants' common interests and conflicts. Finally in the visualization layer, the browser presents a group preference graph for visualizing common interests and conflicts. The details of common interest and conflict detection and visualization are described in Sections 5 and 6 respectively.

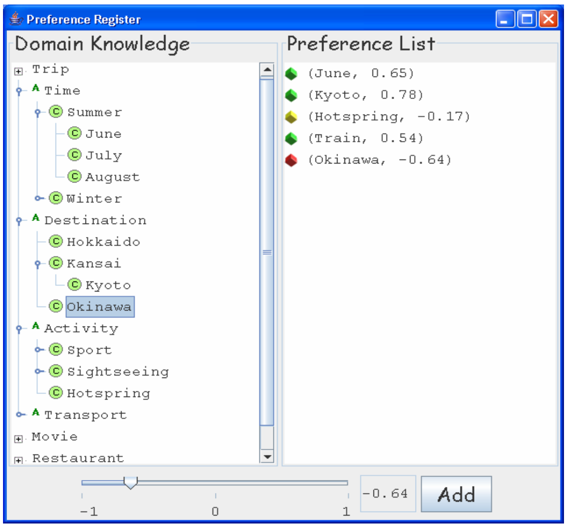

Figure 3: A GUI for seed preference registration

5 Common Interests and Conflicts Detection

5.1 Preference Propagation

Manually inputting a large number of preferences is time-consuming and cumbersome. The preference propagator uses the seed preferences to infer the other unknown preferences through propagating in and across attributes. In this section we first present the methods to calculate the influence factors which reflect how much two concepts in a relation are semantically related. And then we use a propagation algorithm to infer a user's preferences over all features.

5.1.1 Influence Factor

Intuitively, if a person likes concept C_1, and C_1 has a relation to C_2, we may assume that he also likes C_2 to some degree. This is the basic idea of all types of preference propagation. Hence, we adopt this general propagation function:

$$score(C_2) = score(C_1) \times factor(C_1, C_2)\ . \qquad \mathbf{(1)}$$

Then, the problem is how to calculate the influence factor from C_1 to C_2, denoted by $factor(C_1, C_2)$. In our study, we use different methods to obtain the influence factors for different relations.

***PARENT_OF* and *CHILD_OF*.** For this type of relation we use constant factors, which have also been adopted by other works because it is a simple and effective approach [Aydogan, 07]. But the difference is that we use an asymmetric relation, so the two factors are not equal:

$$factor(\mathrm{C}, \mathrm{child}) = \mathrm{a}\ . \quad (2)$$

$$factor(\mathrm{C}, \mathrm{parent}) = \mathrm{b}\ . \quad (3)$$

We have used some personal preference data to tune these factors, and discovered that a = 0.5 and b = 0.8 will provide the best propagation result. For example, when a person gives "December" the score −0.7, and we know "December" is a child of "Winter" from the domain ontology, we can use Functions 1 and 3 to infer that his preference to "Winter" will be $(-0.7) \times 0.8 = -0.56$.

SIBLING_OF. Most concepts connecting with *SIBLING_OF* relation can be arranged as a chain. For example, "Hokkaido", "Kansai", and "Okinawa" are siblings (from domain ontology), but range from north to south according to their latitudes. In order to comply with the transitive property of the propagation function, the factor is calculated as

$$factor(\mathrm{C}, \mathrm{sibling}) = (1 - \mathrm{maxDistance}^{depth(\mathrm{C})})^{diff(\mathrm{C},\mathrm{sibling})}\ . \quad (4)$$

$$factor(\mathrm{sibling}, \mathrm{C}) = factor(\mathrm{C}, \mathrm{sibling})\ . \quad (5)$$

where *depth*(C) is the number of edges in the path from root to the concept C. *diff*(C, sibling) is the percentage of difference between C and this sibling over the largest difference among their siblings. In our experiment (described in Section 6), maxDistance is set to 0.7. Equation 4 comes from the idea that the deeper and closer two siblings are, the more common they are semantically. Apparently, the *SIBLING_OF* relation is symmetric; thus we simply use Equation 5 to get the opposite factor.

ASSOCIATE_WITH. This type of underlying relation between concepts that span across different attributes does not have fixed semantics, thus cannot be recognized in a straightforward way as former two kinds of relation. Inspired by the method of determining inter-topic association of images [Fan, 08], we adopted a statistics-based approach, depicted as follows:

1) Form all ordered pairs of concepts across attributes. For each pair $<C_1, C_2>$, search "C_1", "C_2", "C_1 C_2" in a popular search engine (e.g., Yahoo, http://www.yahoo.com) to get the web page numbers $n(C_1)$, $n(C_2)$, $n(C_1\ C_2)$, then calculate factors as

$$factor(C_1, C_2) = n(C_1\ C_2) / n(C_1)\ . \quad (6)$$

$$factor(C_2, C_1) = n(C_1\ C_2) / n(C_2)\ . \quad (7)$$

2) For all siblings of C_1 (including itself), say C_i, choose only the pair to make that maximizes the $factor(C_i, C_2)$. Vice versa, i.e., for all siblings of C_2, say C_j, only choose the pair to make the $factor(C_1, C_j)$ the largest.

3) If both $<C_1, C_2>$ and $<C_2, C_1>$ are chosen in the above step, then determine the *ASSOCIATE_WITH* relations between them.

4) Normalize factors of determined pairs into a suitable range (e.g., [0.3, 1]).

In the example of the "Trip" domain (for simplicity, it only has four attributes: "Time", "Destination", "Activity", and "Transport"), [Tab. 2] shows part of the *ASSOCIATE_WITH* relations found by this approach. For example, the factor from "To Kyoto" to "Temple" is 1.00, and in the reverse direction, the factor from "Temple" to "To Kyoto" is 0.31. In the searching step, in order to make the meaning of a concept more accurate, and also easier for the search engine to understand, we add a preposition before the main word, e.g., add "To" before concepts in "Destination" attribute, "In" before "Time", and "By" before "Transport".

Pair	Factor	Factor (reverse order)
<To Kyoto, Temple>	1.00	0.31
<To Okinawa, In July>	0.92	0.30
<In Summer, Swimming>	0.85	0.36
<In July, Swimming>	0.61	0.43
<To Hokkaido, Skiing>	0.59	0.30
<To Okinawa, Swimming>	0.49	0.30
<To Okinawa, By Air>	0.36	0.30

Table 2: Partial influence factors of ASSOCIATE_WITH relations

5.1.2 Multi-Seed Propagation Algorithm

When a person specifies more than one preference, the following issues should be considered in propagation:

- For those to-be-decided preferences, how should influences from different seeds be coordinated?
- How can it be assured that different orders of propagation will lead to the same result?
- How can all unknown preferences be covered by propagation?
- When should propagation be stopped?

[Thomopoulos 06] also encountered this kind of problem. In this paper, we propose a multi-seed propagation algorithm to solve it. The main idea is a kind of heuristic: the score with the greatest absolute value of a preference will survive, because the greater absolute value indicates more noticeable interest, as well as provide more information about that person's opinion. One preference stops propagating when it cannot influence its neighbors, which reduces lots of redundant computing steps. As we know, Dijkstra's algorithm can be used as a single-seed propagation algorithm. But in the case of multiply seeds, simply looping the Dijkstra's algorithm will lead much higher complexity than the one we proposed.

[Fig. 4] shows the pseudo code of the multi-seed propagation algorithm. Before propagating, factors of all relations and scores of seed preferences have been determined. Temporarily, we set the scores of to-be-decided preferences to 0. There are two types of preferences, SEED and INFERRED. Each preference stays in ACTIVE or INACTIVE state at any given moment. This recursive procedure uses a flag to indicate when to stop (lines 1, 13, 14); the stop condition is reached when all preferences are INACTIVE (lines 2-4). In line 5, "toConcept" denotes the neighbours of a concept. At the beginning, only SEED preferences are ACTIVE. Each ACTIVE preference tries to update its neighbour's score through the propagation function (Formula 1 given in Section 4.1). The ACTIVE preference succeeds in updating only if the absolute value of the new score is larger than the previous INFERRED preference (SEED preference scores will never be changed) (lines 5-10). When a neighbour is given a new score, its state is changed to ACTIVE (line 11). After it finishes attempting to update all its neighbour's, the ACTIVE preference changes to INACTIVE (line 12).

```
     Input: P (only seed preferences have scores)
    Output: P (all preferences have scores)
 Procedure: propagate

isEnd ← true
For each preference
  If preference is ACTIVE
    isEnd ← false
    For each toConcept of preference.concept
      If toConcept.preference is INFERRED
        newScore ← score(preference) *
factor(preference.concept, toConcept)
        If |newScore|>|score(toConcept.preference)|
          score(toConcept.preference) ← newScore
          set toConcept.preference to ACTIVE
    set preference to INACTIVE
If isEnd is false
  call propagate
```

Figure 4: Pseudo code of the multi-seed propagation algorithm

For example, we already have two seed preferences: *score*(C_1) = −0.7, *score*(C_2) = 0.88, and influence factors of relations between C_1/C_3 and C_2/C_3: *factor*(C_1, C_3) = 0.8, *factor*(C_2, C_3) = 0.5. The problem is how to determine *score*(C_3). There are two ways that can infer *score*(C_3): *score*(C_1) × *factor*(C_1, C_3) = −0.56; *score*(C_2) × *factor*(C_2, C_3) = 0.44. Since |−0.56| > |0.44|, we set *score*(C_3) = −0.56.

5.2 Preference Merging

The system then merges the propagation results to generate a group user preference. It first normalizes participants' scores, because different persons might use different measures to denote their preferences. Then it uses the average score and the standard deviation for group preference calculation. The average score represents the degree of common interest, and the standard deviation of scores indicates the degree of conflict. Thus the common interests and conflicts of the participants are detected.

Different persons might use different measures to denote their scores of preference. That is to say, scores in different personal preferences are not obtained by universal measurement. Thus, we must normalize these scores before merging.

For each meeting participant, we normalize his/her original score of each concept according to the following general normalization equation [Yu, 06]:

$$s_C' = \frac{s_C - s_{\min}}{s_{\max} - s_{\min}} \times (U_{\max} - U_{\min}) + U_{\min} \quad \textbf{(8)}$$

where s_C is the original score of concept C, s_C' is the normalized score of C, and s_{max} and s_{min} are the maximum and minimum scores in the preference, respectively. [U_{min},

$U_{\max}$] is the universe of scores. Here, $U_{\max} = 1$ and $U_{\min} = -1$, i.e., the normalized score is always on a standard universe of [−1, 1].

The objective of our study is to obtain important social contexts, i.e., common interests and conflicts, by analyzing the personal preferences of all meeting participants. Compared to other measures, i.e., median and mode, mean is more reliable and stable. Therefore, we use the average score to indicate the common degree of interest. And we take the standard deviation of scores, a most commonly used method, as the degree of conflict for different participants.

$$averageScore(C_i) = \frac{1}{m}\sum_{k=1}^{m} score_{\mathrm{person}_k}(C_i) \cdot \quad \textbf{(9)}$$

$$standardDeviation(C_i) = \sqrt{\frac{1}{m}\sum_{k=1}^{m}(score_{\mathrm{person}_k}(C_i) - averageScore(C_i))^2} \cdot \quad \textbf{(10)}$$

where m is the number of participants.

6 Visualization

Our system offers a group preference graph for visualization of group common interests and conflicts [see Fig. 5]. The graph is generated by using ZGRViewer (http://zvtm.sourceforge.net/zgrviewer.html), which is widely adopted for interactive visualization of social structures. An ellipse node denotes a concept, the label is the name of the concept, and the left numeral underneath is the averages score, which determines the size of that node. The numeral in the square bracket is the standard deviation. An edge denotes two opposite relations of a pair of concepts, with the factors on the edge. Different styles of edges indicate different types of relations. It highlights conflicts and common interests as follows:

- Top 3 conflicts, the nodes with the greatest three standard deviations, are represented by red background color; they are “Okinawa”, “Bicycle”, and “June” in this example.
- Top 3 common likes, the nodes with the greatest three average values (except the nodes in the conflicts set), are represented by green background color; they are “Summer”, “August”, and “Temple” in this example.
- Top 3 common dislikes, the nodes with the smallest average values (except the nodes in the conflicts set), are represented by purple background color; they are “Swimming”, “January”, and “February” in this example.

With the group preference visualized via a public display, meeting participants can intuitively understand the global state of their opinions. Through this, a participant can compare his own opinion with others’ in private, find his position in the group, predict the possible results, or prepare key points to persuade others, etc. The common disliked outcomes may be ignored to save time, and the conflicted outcomes can be retrieved for sufficiently analysis and discussion. Furthermore, preference can be also clustered to display groupings of opinion, allowing within-group and between-group debate [Eagle, 03]. In general, the preference visualization tool would be helpful to facilitate discussion and decision making.

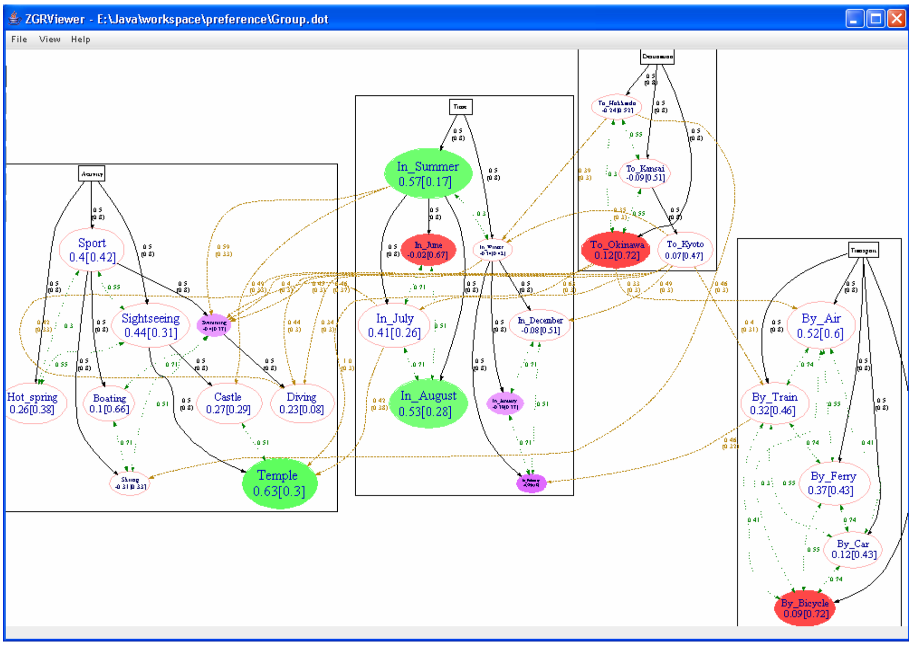

Figure 5: Common interests and conflicts visualization

7 Evaluation

To evaluate the effectiveness of our approach for detecting common interests and conflicts, we conducted an experiment. Expected results include: subjective acceptance of our system, the accuracy of preference propagation, and the accuracy of preference merging. Seven participants (students and staff in the media center of Kyoto University) discussed a plan for a “Trip” in the next year. According to this topic, we chose 4 attributes and 26 concepts (i.e., possible values) to form an experimental ontology (a complete domain ontology will be too large to do the experiment): “Time” (with 8 concepts), “Destination” (with 4 concepts), “Activity” (with 9 concepts), and “Transport” (with 5 concepts).

7.1 Experimental Settings

The experiment included the following steps:

(1) Each participant specified preference scores to all concepts using a graphical interface. Then we calculate the group preferences in two ways: one merging all specified scores (referred to as “pure merging results”), and the other merging propagated scores which take partial specified scores as seed preferences (referred to as “propagated merging results”). There are 3 different groups, which are discussed below.

(2) The participants were first divided into 2 groups (one including 3, and the other 4 participants). Each group chose these concepts by discussion: top 3 common likes, top 3 common dislikes, and top 3 conflicts. Then, they merged into a single group and repeated the procedure. These results are referred to as "actual discussion results".

(3) All participants viewed the group preference graph [see Fig. 5], which depicts the pure merging results of the largest group, and then they were asked to answer a questionnaire [see Tab. 3].

Note that the procedure of this experiment is different from actual use of the system. In practical use meeting participant only need to specify partial preferences, view the group preference graph before discussion, and exchange opinions to reach a consensus or compromise on a final group decision.

Question	Average rate*
Q1. *I would like to express my opinions about the topic privately before meeting.*	3.86
Q2. *The system was easy to understand and use.*	3.86
Q3. *The system could help me to express my preferences correctly.*	3.57
Q4. *I was glad to give preference values for all the concepts about the topic.*	3.50
Q5. *I was satisfied with the results of group discussion I attended.*	2.79
Q6. *It was easy to draw the results by group discussion.*	1.79
Q7. *The group graph exactly demonstrated the common interests and conflicts.*	3.57
Q8. *This tool was helpful and I would like to use it for a real meeting.*	4.21

*Table 3: Questionnaire and rating results (*5 = strongly agree; 4 = agree; 3 = neutral; 2 = disagree; 1 = strongly disagree)*

7.2 Evaluation Results

7.2.1 User Survey

We conducted a user survey to evaluate user acceptance of our system and user feeling about the meeting discussion. The result also supplied the basis for analysis of the other two dedicated results. The questionnaire and results are reported in [Tab. 3].

The ratings suggest that the outcomes of the questionnaire are encouraging. The subjects expressed their positive opinions on system interfaces and the way of declaring themselves before meeting (*Q1*, *Q2*, and *Q8*). The capability of the interface to express personal preferences is acceptable but needs improvement (*Q3*). It seems difficult to draw out the common interests and conflicts by discussion (*Q5* and *Q6*). The participants were more satisfied with the pure merging results than the actual discussion results (*Q7*). The answers of these three questions (*Q5-7*) verified that usually meetings are not effective. The participants were not very pleased about giving preferences for all the concepts (*Q4*). We can therefore conclude that propagation is important, because it could reduce the number of concepts that need to be explicitly specified. Merging is also helpful because it can detect the common interests and conflicts more easily and precisely. In general, our system could be used to facilitate the process of making decisions.

7.2.2 Propagation Result

We used the cosine similarity to calculate the accuracy of preference propagation. For each person, there are a total of 26 specified preference scores. N (from 4 to 16) number of these scores were randomly chosen as seeds and the other 26 − N were used as standards for comparison. After propagating, we obtained 26 − N inferred preference scores, and then compared these with the standards. For consistency sakes, we always chose 10 pairs to compare. Then, the cosine similarity between the inferred and standard scores was calculated as follows:

$$cosineSimilarity = \frac{\sum_{i=0}^{10} score_{\text{SEED}}(C_i) \times score_{\text{INFERRED}}(C_i)}{\sqrt{\sum_{i=0}^{10} score_{\text{SEED}}(C_i)^2 \times \sum_{i=0}^{10} score_{\text{INFERRED}}(C_i)^2}} \quad (11)$$

The value range of cosine similarity is [−1, 1]; therefore, as long as the similarity is greater than 0, it suggests a success of propagation to some degree, i.e., that the inferred results were similar to specified ones.

We compared our propagation approach with [Aydogan, 07]'s method that sets the factor of *PARENT_OF* and *CHILD_OF* to 2/3, all *SIBLING_OF* to 4/7, and does not include an *ASSOCIATE_WITH* relation. We ran the propagation process 20 times for each participant and got the average of the cosine similarity. [Fig. 6] shows the cosine similarities for different numbers of seeds.

The result shows that our approach outperforms [Aydogan, 07]'s method especially when the number of seed preferences (N) is small. We observe that the cosine similarity (cos) increases with N using both methods. If N>16, the difference between the two methods is very small, and the similarities continuously approach to 1. This is reasonable, because if the system collects more information from a person, it may achieve higher inference accuracy. In real use, N is usually small because users do not intend to input a large number of seed preferences. Another superiority of our approach comes from the ability of propagation across attributes, thus it can successfully cover all concepts in the domain even if a person fails to specify some attributes.

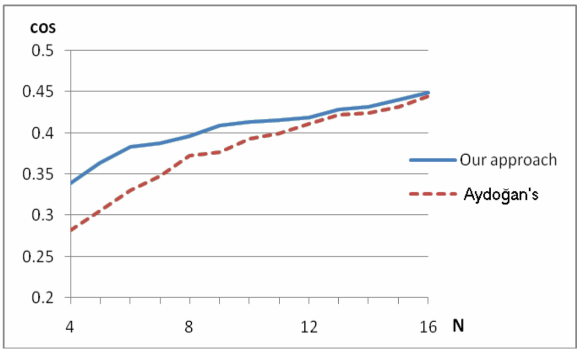

Figure 6: Cosine similarity vs. number of seeds

7.2.3 Merging Result

We use the percentage of hits to calculate the accuracy of preference merging. Suppose that the standard result of a type of top 3 is (C_1, C_2, C_3), and the merging result is (C_1', C_2', C_3'). Then we count the number of hits, i.e., to see how many items in merging result exist in the standard result. For example, if $C_1' = C_2$, $C_2' = C_3$, and $C_3' \neq C_1$, the number of hits is 2. As we are interested in only the top 3 concepts, the hit percentage can be calculated by

$$hitPercentage = \frac{|\,intersection(standardTop3, mergedTop3)\,|}{3} \quad . \tag{12}$$

According to the result of the user survey, the participants were more satisfied with the pure merging results than the actual discussion results; thus, we adopted the pure merging results as the standard. We compared the propagated merging results with actual discussion results [see Tab. 4]. We ran the propagated merging process 20 times for each group, selected the three kinds of top 3 (conflicts, common likes, and common dislikes), and then calculated the average of the hit percentages.

Number of seeds	Hit percentage
4	23.1%
8	30.4%
12	36.7%
16	42.7%
Actual discussion	29.6%

Table 4: Propagated merging vs. actual discussion

The comparison shows that if the number of seeds is more than 8 (i.e., every participant specifies preference scores for more than 8 concepts out of the total 26 concepts), the hit percentage of getting the common interests and conflicts will be higher than by actual discussion. The reasons why the discussion cannot get a high hit percentage are, possibly, that people are not accustomed to talk about their dislikes; that during discussion, some participants did not keep the same opinions as that inputted into our system; and, that the discussion is influenced by many other factors, e.g., the leader of the group, the rules about making decisions, the number of members, and so forth.

7.3 Remarks

According to the evaluation results, we are convinced that traditional meetings are not satisfactory for the users, whereas our approach is able to increase their satisfaction by demonstrating the pure merging results. Although there is an error between the propagated merging results and the pure merging results, they are more accurate than the actual discussion results. This proves that our work is promising. Through the group preference graph, which reflects the group’s opinion by showing common interests and conflicts, a participant can compare his own opinion with others’ in private, find his position in the group, predict the possible results, or prepare key points to persuade others, etc.

The present approach has several shortcomings, as displayed in our current experiment. Some participants commented that it would be better if only 3 or 5 levels were offered for rating as preferences rather than using a score from −1 to 1. This might be true, because it is difficult to quantify a person's desires so precisely. Another issue is that a participant may like a parent concept very much, but give low scores to all its children. This is caused by incompleteness of our experimental ontology of trip domain. For instance, besides "Castle" and "Temple", there are lots of other children for "Sightseeing", but are not in the experimental ontology. It is worth noting that in different languages or cultures, the *ASSOCIATE_WITH* relationship are different. For example, in Japanese common sense, despite June is in summer, it is not hot but a rainy season. This knowledge apparently has not been captured by analyzing English language webs. Besides, people's statement of preferences is often erroneous [Decker, 07], i.e., inconsistent. Finally, the method used to find proper parameters of our algorithm is not theoretically validated. They were tuned by partial persons' experimental data, but may be not suitable for other persons. These issues will be addressed further in our future work.

8 Conclusion

To facilitate decision-making process in meetings, we have proposed the Meeting Warming-up system that detects and visualizes common interests and conflicts among meeting participants. Using the system, participants can warm up for discussions, leading to an easy and friendly start with the commonly agreed outcomes, ignoring the potentially commonly disagreed outcomes to save time, and preparing to discuss the possible conflicts in advance.

We have introduced the user preference model and the three layers of Meeting Warming-up, i.e., preparation, detection and visualization. Despite some shortages remain, the experiment results show that ontology-based preference model enhanced with additional semantic relations is helpful to express one's opinion in a compact way. The propagation algorithm has acceptable accuracy and low computation complexity. The merging mechanism can aggregate the group preference more easily and precisely than actual discussion. And the users are satisfied with the graphical user interface, which intuitively demonstrates the group members' opinions as a whole.

In the future, more experiments will be conducted to improve our current system. Also we plan to explore other preference elicitation methods and real-time supports for smart meetings.

Acknowledgements

This work is partially supported by the Doctorate Foundation of Northwestern Polytechnical University of China under Grant No. CX200814, and the Ministry of Education, Culture, Sports, Science and Technology, Japan under the project of "Cyber Infrastructure for the Information-explosion Era".

References

[Aydogan, 07] Aydogan, R., and Yolum, P.: "Learning Consumer Preferences using Semantic Similarity"; In Proc. of the 6th international Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'07), ACM Press (2007), 1-8.

[Chen, 04] Chen, H., Perich, F., Chakraborty, D., Finin, T. and Joshi, A.: "Intelligent Agents Meet Semantic Web in a Smart Meeting Room"; In Proc. of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, IEEE Computer Society, New York (2004), 854-861.

[Chen, 05] Chen, A.: "Context-Aware Collaborative Filtering System: Predicting the User's Preference in the Ubiquitous Computing Environment"; LoCA 2005, LNCS 3479 (2005), 244-253.

[Chomicki, 02] Chomicki, J.: "Querying with Intrinsic Preferences"; In Proc. of the 8th international Conference on Extending Database Technology: Advances in Database Technology, Springer-Verlag (2002), 34-51.

[Decker, 07] Decker, R., Meißner, M., and Scholz., S.W.: "Detecting and Debugging Erroneous Statements in Pairwise Comparison Matrices"; Operations Research Proceedings (2007), 277-282.

[DiMicco, 06] DiMicco, J.M., Hollenbach, K.J. and Bender, W.: "Using visualizations to review a group's interaction dynamics"; In CHI '06 extended abstracts on Human factors in computing systems, ACM, Montreal, Quebec, Canada (2006), 706-711.

[Eagle, 03] Eagle, N., and Pentland, A.: "Social Network Computing"; In Proc. UbiComp'03 (2003), 289-296.

[Fan, 08] Fan, J., Keim, D.A., Gao, Y., Luo, H., and Li, Z.: "JustClick: Personalized Image Recommendation via Exploratory Search from Large-Scale Flickr Image Collections"; IEEE Transactions on Circuits and Systems for Video Technology, Vol. 18, No. 8 (2008).

[Henricksen, 06] Henricksen, K., Indulska, J., and Rakotonirainy, A.: "Using Context and Preferences to Implement Self-adapting Pervasive Computing Applications: Experiences with Auto-adaptive and Reconfigurable Systems"; Softw. Pract. Exper. 36 (2006), 1307-1330.

[Huang, 04] Huang, W., and Zhang, P.: "An Empirical Investigation of the Effects of GSS and Group Process on Group Outcome in Small Group Decision-makings"; Journal of Computer Information Systems, Vol. XLV, No. 1 (2004), 23-29.

[Jaimes, 05] Jaimes, A. and Miyazaki, J.: "Building a Smart Meeting Room: from Infrastructure to the Video Gap (Research and Open Issues)"; In Proc. of the 21st International Conference on Data Engineering Workshops, IEEE Computer Society (2005), 1173-1182.

[Jameson, 03] Jameson, A., Baldes, S., and Kleinbauer, T.: "Enhancing Mutual Awareness in Group Recommender Systems"; In Proc. of the IJCAI'03 Workshop on Intelligent Techniques for Web Personalization, AAAI (2003).

[Keeney, 93] Keeney, R., and Raiffa, H.: "Decisions with Multiple Objectives: Preferences and Value Tradeoffs"; Cambridge University Press (1993).

[Kernkamp, 06] Kernkamp, R.: "Functional Requirements for a Conflict Managing Meeting Assistant"; In Proc. of the 4th Twente Student Conference on IT (2006).

[Kim, 07] Kim, J., Kim, J., and Kim, C.: “Ontology-Based User Preference Modeling for Enhancing Interoperability in Personalized Services”; Universal Access in HCI, HCII 2007, LNCS 4556 (2007), 903-912.

[Kim, 08] Kim, T., Chang, A., Holland, L., and Pentland, A.: “Meeting mediator: enhancing group collaboration with sociometric feedback”; In Proc. of CHI '08, ACM, New York (2008), 3183-3188.

[Kulyk, 05] Kulyk, O., Wang J., and Terken J.: “Real-Time Feedback on Nonverbal Behavior to Enhance Social Dynamics in Small Group Meeting”; MLMI 2005, LNCS 3869 (2006), 150-161.

[McGrath, 84] McGrath, J.E.: “Groups: Interaction and Performance”; Englewood Cliffs, NJ: Prentice Hall (1984).

[Middleton, 01] Middleton, S.E., Roure, D.C.D. and Shadbolt, N.R.: “Capturing knowledge of user preferences: ontologies in recommender systems”; In Proc. of the 1st international conference on Knowledge capture, ACM, Victoria, British Columbia, Canada (2001), 100-107.

[Mikic, 00] Mikic, I., Huang, K. and Trivedi, M.: “Activity Monitoring and Summarization for an Intelligent Meeting Room”; In Proc. of the Workshop on Human Motion (HUMO'00), IEEE Computer Society, Los Alamitos, CA, USA (2000), 107-112.

[Nass, 04] Nass, C., and Brave, S.: “Voice Activated: How People are Wired for Speech and How Computers will Speak with Us”; MIT Press (2004).

[Pianesi, 08] Pianesi, F., Zancanaro, M., Not, E., Leonardi, C., Falcon, V. and Lepri, B.: “Multimodal support to group dynamics”; Personal Ubiquitous Comput., 12 (3), 2008, 181-195.

[Rienks, 06] Rienks, R., and Heylen, D.: “Dominance Detection in Meetings Using Easily Obtainable Features”; MLMI 2005, LNCS 3869 (2006), 76-86.

[Romano, 01] Romano, Jr., and Nunamaker, Jr.: “Meeting Analysis: Findings from Research and Practice”; In Proc. of the 34th Hawaii International Conference on System Sciences (HICSS’01), IEEE Press (2001), 1072-1085.

[Sarwar, 01] Sarwar, B., Karypis, G., Konstan, J., and Reidl, J.: “Item-based Collaborative Filtering Recommendation Algorithms”; In Proc. of the 10th international Conference on World Wide Web (WWW’01), ACM Press (2001), 285-295.

[Schickel, 05] Schickel-Zuber, V., and Faltings, B.: “Heterogeneous Attribute Utility Model: A New Approach for Modeling User Profiles for Recommendation Systems”; In Proc. of the WebKDD’05 Workshop at the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD’05), ACM Press (2005), 80-91.

[Schickel, 06] Schickel-Zuber, V., and Faltings, B.: “Inferring User’s Preference using Ontologies”; In Proc. of AAAI 2006 (2006), 1413-1418.

[Shaw, 71] Shaw, M.E.: “Group Dynamics: The Psychology of Small Group Behavior”; McGraw Hill (1971).

[Stefanidis, 06] Stefanidis, K., Pitoura, E., and Vassiliadis, P.: “Modeling and Storing Context-Aware Preferences”; ADBIS 2006, LNCS 4152 (2006), 124-140.

[Tang, 91] Tang, J.C.: “Findings from Observational Studies of Collaborative Work”; Int J Man-Mach Stud 34 (1991), 143-160.

[Thomopoulos, 04] Thomopoulos, R.: “Expressing Preferences in a Viewpoint Ontology”; In CoopIS, DOA, and ODBASE, LNCS 3761 (2005), 1596-1604.

[Thomopoulos, 06] Thomopoulos, R., Buche, P., and Haemmerle, O.: "Fuzzy Sets Defined on a Hierarchical Domain"; IEEE Transactions on Knowledge and Data Engineering, Vol. 18, No. 10 (2006), 1397-1410.

[Yu, 06] Yu, Z., Zhou, X., Hao, Y., and Gu, J.: "TV Program Recommendation for Multiple Viewers Based on User Profile Merging"; User Modeling and User-Adapted Interaction (UMUAI) 16(1), Springer Press (2006), 63-82.

[Yu, 07] Yu, Z., Ozeki, M., Fujii, Y. and Nakamura, Y.: "Towards smart meeting: enabling technologies and a real-world application"; In Proc. of the 9th international conference on Multimodal interfaces, ACM, Nagoya, Aichi, Japan (2007), 86-93.

[Yu, 09] Yu, Z., and Nakamura, Y.: "Smart Meeting Systems: A Survey of State-of-the-Art and Open Issues"; ACM Computing Surveys, Vol. 41, No. 3 (2009).