Toward Intelligent Meeting Agents

Hsinchun Chen
Andrea Houston
Jay Nunamaker
University of Arizona

Jerome Yen
*Hong Kong University
of Science and Technology*

An experiment with an

AI-based software agent

shows that it can help users

organize and consolidate ideas

from electronic brainstorming.

The agent recalled concepts as

effectively as experienced

human meeting facilitators

and in a fifth of the time.

roupware—computer aids designed for collaborative work—has produced measurable productivity gains for major corporations in recent years. Agent software enhances productivity even more by helping groupware perform convergent tasks, thus freeing users for more creative work. The result is a significant time savings over groupware without agents.

Agents represent a fundamental shift in the human-computer interaction paradigm, however. Broadly defined, an agent is a program that performs unique tasks without direct human supervision. As such, it transforms the user from a worker into a manager who delegates tasks to that agent. Agents must be able to engage and help all types of end users—whether that is to act as "spiders" on the Internet searching for relevant information, schedule meetings based on an executive's constraints, or filter articles based on learned user profiles.[1]

In this article, we describe an experiment to validate the performance of an intelligent agent. An intelligent agent differs from an agent in part because of its ability to reason about a task and learn from task performance. The sidebar "What makes an agent intelligent?" describes the differences in more detail. Our intelligent agent supports the GroupSystems electronic meeting system developed by the University of Arizona. The agent organizes ideas from brainstorming sessions into category lists for further discussion, a task often difficult for human electronic meeting facilitators.

The goal of our experiment was to verify the agent's performance against that of human meeting facilitators. The experiment addresses a weakness in intelligent agent research: Many projects involving intelligent agents have rarely experimentally or empirically tested or verified performance against humans.[1] In our experiment, we used 12 human facilitators and four sessions of varying sizes. We were able to verify that the agent performed as well as experienced human facilitators in identifying important meeting concepts—and it took a fifth of the time. The agent was less effective in generating precise and relevant concepts. However, since identifying concepts is a much more time-consuming task for meeting facilitators, the agent is still extremely practical. The experiment facilitators have used it in many actual sessions to create a strawman category list, which they then refine to be more precise.

CONSOLIDATING IDEAS

Electronically supported meetings can help reduce the time required for managers to complete complex projects by 90 percent, according to *Fortune*.[2] Electronic meeting systems can improve meeting quality by permitting

anonymous comments, by providing instant access to information, by offering a structure for making decisions and evaluating alternatives, and by encouraging equal participation.

An electronic meeting has several stages. One is *idea generation*. During this stage, the group feels energized as it realizes the positive contribution to the question posed. Group members' overall satisfaction increases. The *idea organization* stage condenses all the comments into a manageable list of categories. As group members perceive that categorizing ideas is going to be difficult and time-consuming, satisfaction begins to decrease. When the group senses they have reached a manageable list of categories (typically 10 to 15 items), satisfaction starts to increase again. As the group settles on a set of consensus categories, satisfaction continues to increase and the group proceeds to complete its task.

A major advantage of electronic meetings is that members can brainstorm in parallel. Indeed, electronic meeting systems are generally very effective during idea generation. A major disadvantage is that all the ideas from brainstorming—typically several hundred comments—must be organized. For this stage, the meeting system often requires additional tools. Idea-organizing tools let participants identify and consolidate ideas, typically by separately suggesting categories that merit further consideration.

The process of identifying crucial ideas embedded in meeting comments and generating a category list is a convergent task that must meet several challenges:

- *Information overload.* Idea generation tends to produce many ideas, which results in information overload.[3] In a typical meeting of 10 to 20 participants, several hundred brainstorming comments can be generated in less than an hour, making it extremely difficult for participants to browse and consolidate comments. The participants are often impressed with the number of ideas generated but become overwhelmed with the task of organizing them into categories.

- *Lack of a collaborative vocabulary.* During electronic brainstorming, the goal is to generate creative, uncensored ideas. A natural consequence is that different participants use different vocabularies to convey those ideas. In fact, research shows that in a spontaneous word choice for objects in five domains, the probability that two people would use the same term is less than 20 percent.[4] During idea organization, these vocabulary differences can cause problems, and there is little system support to alleviate them.[5] Human meeting facilitators often try to address the vocabulary problem, but the overwhelming demand to monitor group dynamics and operate in a complex meeting environment makes this task difficult, especially when they are not familiar with the subject.

- *Pressure to synthesize tasks.* To be useful, ideas from electronic brainstorming must be consolidated and organized in a short time. Participants may take up to an hour to browse and understand the ideas generated, judge their merits, merge similar ideas, eliminate redundant or irrelevant ideas, consult other members, and so on. Many comments are raw or unpolished and often require special synthesizing. This process can be frustrating and ineffective, causing group satisfaction levels and productivity to diminish significantly. Some unique ideas may also be lost during this cognitively demanding process.

- *Sensitive topics and lack of trust.* Sensitive topics may emerge during idea organization. Participants may be reluctant to suggest such topics as categories because they are afraid to cause disagreements. An idea-organizing tool must address this reluctance by letting participants comment anonymously.

What makes an agent intelligent?

The basic idea of agent research is to develop programs that have some special skill and that engage and help users in complex tasks such as planning trips, recommending houses, and facilitating meetings. Agent research is new and broad. In software engineering, it covers diverse applications such as interface design, communication and coordination, and adaptation and learning algorithms.

Various researchers have used different names for their agents, including autonomous agents, adaptive interfaces, intelligent interfaces, knowbots, and intelligent agents.

However, we believe—and many researchers concur—that to be called "intelligent," an agent must satisfy several interrelated criteria. Daniel Weld offers five attributes,[1] which we believe capture the essence of an intelligent agent:

- *Integrated.* The agent must support an understandable, consistent interface.

- *Expressive.* The agent must accept requests in different modalities.
- *Goal-oriented.* The agent must determine how and when to achieve a goal.
- *Cooperative.* The agent must collaborate with the user.
- *Customized.* The agent must adapt to different users.

In summary, an intelligent agent must be capable of autonomous goal-oriented behavior in some environment that acts as a personal assistant to the user (such as electronic meeting software). Our meeting agent is able to satisfy these criteria, and so is truly "intelligent."

Reference
1. D. Weld, "The Role of Intelligent Systems in the National Information Infrastructure," *AI Magazine*, Fall 1995, pp. 45-64.

MEETING AGENT

Figure 1 shows some output from a typical electronic brainstorming session. The constraints of meeting time and variations in participants' typing skills often make the comments quite "noisy." Typos and incomplete or ungrammatical sentences are common. Comments may vary from a few words to multiple paragraphs.

After a one-hour brainstorming session, meeting participants typically take a break of 10 to 15 minutes before moving to idea organization. Thus, an agent must analyze and classify comments within this brief time using the electronic meeting system's platform. It must also accommodate the general-purpose, domain-independent nature of the meeting software, which is designed to support any kind of meeting, task, participant, and institution.

What problems must our company address to improve product development and the engineering-support interface?

1.1 Vision
1.2 Market selection
1.3 Product selection
1.4 Individuals who should be focused on market or product selection full time are burdened with day-to-day activities that greatly reduce their effectiveness.
...
1.10 LACK OF STREAMLINED BACK END EXTENDING FROM SILICON TO PRODUCTION. CURRENTLY, DESIGNERS ARE RESPONSIBLE FOR MAINTAINING AND COORDINATING THE ENTIRE PROCESS.
...
2.3 When a development is undertaken, it must be accompanied by commitment not only from the design engineer, but also from test, fab, qa, marketing for PDSs, etc. As it is only the design enginer has accountability.
...
2.5 re 2.3 - everybody has accountability, but to whom, and with what measures/rewards/punishments for not attaining goals - what goals, they're seldom defined or stated clearly

Figure 1. Comments from part of an electronic brainstorming session. A typical session can generate several hundred comments, usually unpolished and often cryptic.

"What problems must our company address to improve product development and the engineering/support interface?"

Topics classified:
** DESIGN/PRODUCT DEVELOPMENT
** SALE/PRODUCT ENGINEER/SUPPORT GROUP
** TEAM/PRODUCT ENGINEER/SUPPORT GROUP
** MARKET/PRODUCT DEFINITION/PLANNING/REQUIREMENT
** PROJECT
** CUSTOMER
** TESTING
** PROCESS
** SYSTEM
** PEOPLE
** MANAGEMENT
** SCHEDULE/CYCLE TIME/DEVELOPMENT CYCLE/CONTROL
** TEAM
** CHANGE
** GOAL

Figure 2. System-generated categories for comments generated during electronic brainstorming.

In our experiments we used the GroupSystems electronic meeting system,[2,6] which is now at 500 sites, including business, government, and universities. In our implementation, GroupSystems runs on a 486 66-MHz PC with Microsoft Windows 3.1. The agent uses files from electronic brainstorming sessions extracted directly from GroupSystems as input.

The GroupSystems configuration we used required that we develop the agent in Borland C++. We added two graphical user interfaces, one for generating concepts and one for analyzing comments. We describe these in more detail later. Basically, a meeting facilitator can invoke the agent at any time to produce a category list of ideas. Each category is linked to specific comments, which users can browse. Figure 2 shows our agent's category list for the entire session excerpted in Figure 1, for example.

Algorithms

Several natural-language parsing techniques that are domain-dependent and based on AI concepts work extremely well in narrow topics with well-defined vocabularies.[5] However, the noisy data, diversity of vocabularies, and jargon in meetings make it infeasible to use these techniques for organizing the ideas generated in electronic brainstorming. The agent uses three techniques to produce the list, all of which are described in detail elsewhere:[3]

- *Automatic indexing.* The agent's first task is to identify the content of each comment. Automatic indexing is domain-independent and computationally efficient. The technique identifies subject descriptors in each comment and computes the number of times each descriptor occurs in the entire session. A *descriptor* can be a one-, two- or three-word phrase; a *document* is one comment. The user can remove incidental or noisy descriptors from consideration by setting a term-frequency threshold (typically two or three occurrences of a term) for a document.
- *Cluster analysis.* The importance of each descriptor in representing the content of the entire document (comment) varies. Using term frequency and inverse document frequency, cluster analysis assigns weights to each term in a document to represent the term's level of importance. Term frequency measures how often a particular term occurs in the entire collection. Inverse document frequency indicates the specificity of the term and allows terms to get higher weights during cluster analysis. Cluster analysis then coverts raw data (indexes and weights) into a matrix that shows similarities and dissimilarities among terms. Using an asymmetric cluster function,[7] we generated a network-like concept space of terms and their weighted relationships.
- *Hopfield net classification.* The Hopfield net is a neural net that groups similar term pairs.[8] We chose the Hopfield net over others because it has a content-addressable memory and has been successfully used in other AI applications. The net stores information in single-layer, interconnected neurons (nodes) and weighted synapses (links). In applying the net to idea organization, we treated each term as a neuron and the asymmetric weight between any

Computer

two terms as the unidirectional, weighted connection between neurons. Using each term as an individual input pattern, the Hopfield algorithm activates neighbors, combines weights from all associated neighbors, and repeats this process until convergence occurs. The resulting output reveals all concepts semantically relevant to the input term. By repeating this process for all the terms according to their occurrence (most frequent to least frequent), we identified the underlying clusters of ideas in the EBS comments.

User interaction

When meeting participants invoke the agent, they see the screen in Figure 3. The first thing they do is select the electronic brainstorming session they want the agent to organize. They are most often interested in the session at hand, but they may want to use the agent to analyze previous sessions in the meeting (a meeting can have multiple brainstorming sessions) to compare and refine the category lists. The meeting participant in Figure 3 is looking at a large session (30-60 Kbytes). The agent can also handle small and medium sessions.

The participant has set the document frequency to 4. This means that if a term or phrase appears less than four times, the agent does not consider it as a category descriptor. The facilitator can set the threshold from 1 to 5. If no threshold is set, the agent will default to a value appropriate to the session size. The facilitator can also set a threshold for the co-occurrence weight. If the relationship between two terms is below the threshold, the agent will not pair the terms into a phrase. The higher the weight, the stronger the relationship must be. Again, if the facilitator does not set a threshold, the agent will default to the session size.

The four buttons beneath co-occurence weight give the facilitator the option of starting the concept gener-

ator, analyzing topics (which calls up another screen), setting defaults, or exiting the agent. The facilitator in Figure 3 has selected Generate Topics, and the list appears below.

If the facilitator wants to analyze the topic list, he selects Analyze Topics and sees the screen in Figure 4. He can then highlight a topic for review or editing. He can also add or delete entire topics. The facilitator has highlighted Product/Design/Support. The agent displays all the comments related to that topic. The summary shows the distribution of topic terms in the comments. The agent saves whatever topics the facilitator deletes in an uncategorized
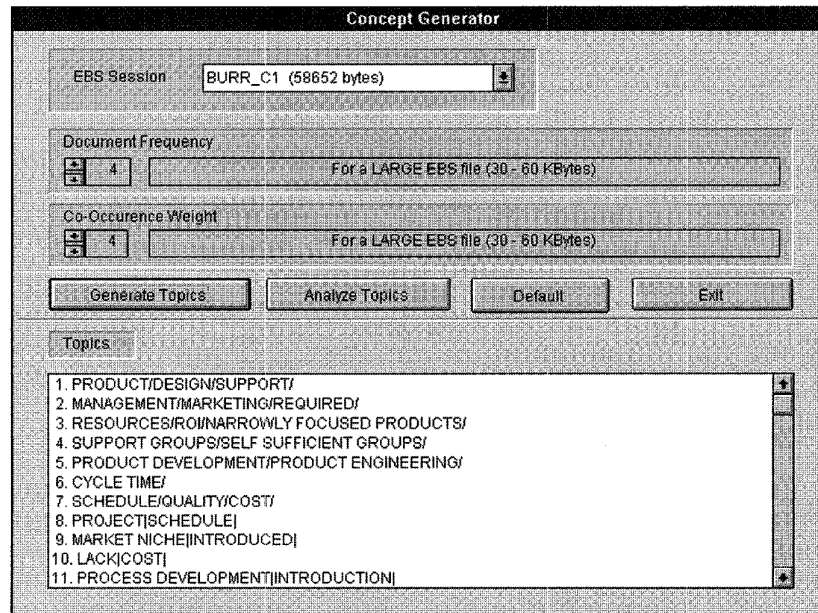

Figure 3. Invoking the agent. The user can start the concept generator or move on to the next screen to analyze comments.
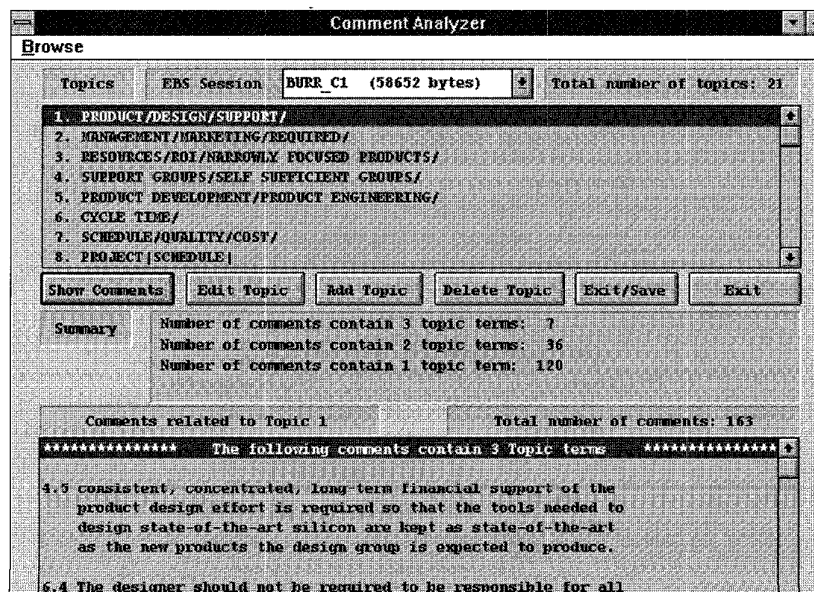

Figure 4. Analyzing the topic list.

Table 1. Results by individual subject and session.							

Session 1: Large session

	Experienced			Inexperienced			Session	Agent
Lists	A1	A3	A6	A2	A4	A5	A7	A8
Recall	.54	.60	.89**	.85	.80	.79*	.87	.67
Precision	.94**	.72*	.85**	.96**	.71	.92**	.93**	.49

Session 2: Small session

	Experienced			Inexperienced			Session	Agent
Lists	A1	A3	A6	A2	A4	A5	A7	A8
Recall	.73	.59	.83	.74	.90	.74	.80	.72
Precision	.98**	.89**	.90**	.92**	.76*	.92**	.73*	.51

Session 3: Large session

	Experienced			Inexperienced			Session	Agent
Lists	B1	B2	B5	B3	B4	B6	B7	B8
Recall	.83	.86	.96	.86	.86	.94	.88	.83
Precision	.83**	.91**	.84**	.94**	.82*	.88**	.97**	.61

Session 4: Small session

	Experienced			Inexperienced			Session	Agent
Lists	B1	B2	B5	B3	B4	B6	B7	B8
Recall	.78	.84	.94	.61	.79	.90	.85	.77
Precision	.88**	.91**	.77	.83	.83*	.73	.82*	.58

where ** indicates that $P \leq 0.05$ and * indicates that $0.05 < P \leq 0.1$ (vs. agent)

comment bin (not shown). When the facilitator is ready to exit the agent, he has a choice of saving or not saving changes to the topic list. He may decide not to save changes and have the agent regenerate a topic list using different co-occurrence weights, for example.

THE EXPERIMENT

Experienced facilitators were able to consistently achieve about 80 percent concept recall and concept precision. We wanted to see how well our agent compared with these figures.

Parameters and procedure

We chose four GroupSystems electronic brainstorming sessions from the files at the University of Arizona. Two of the sessions were small (less than 250 comments) and two were large (more than 400 comments). We omitted medium sessions in the experiment because it would make the parameters unnecessarily complex. Each session had to meet two criteria:

- The category list generated by the meeting participants was available.
- Domain knowledge was not required to adequately categorize the session.

We grouped the four sessions into two sets, each of which contained one large and one small session. We selected 12 facilitators from the university's Center for the Management of Information and from Ventana Corp., a spin-off company that develops GroupSystems technologies. We classified the facilitators according to their years of experience and the number of sessions they had facilitated in the last two years. Experienced facilitators had at least two years experience and had conducted, on average, more than three sessions a month. Inexperienced facilitators had up to two years experience and had conducted three or fewer sessions.

We designated six facilitators as experienced and six as inexperienced. To each set, we assigned three experienced facilitators and three inexperienced facilitators. Each facilitator also reviewed one large session and one small session (in random order). Thus, three experienced facilitators and three inexperienced facilitators reviewed each session set.

We conducted the experiment in three phases.

LIST GENERATION. In this phase, we asked each facilitator to evaluate both sessions in the assigned session set and generate a category list for each session. At the same time, we ran the agent for that session. We recorded the time it took for all facilitators and the agent to generate each category list. The agent took three minutes for the small sessions and five minutes for the large sessions. The facilitators took from five to 81 minutes. The average times for large sessions were 25.8 and 42 minutes (two sets); for the small session, they were 20.5 and 25.7.

LIST EVALUATION. We next asked each facilitator to evaluate and modify eight category lists for each categorized session. The facilitators could not tell which category list was from the agent because the experimenters used the same font for all the lists. The eight lists consisted of

- three lists generated by experienced facilitators,
- three lists generated by inexperienced facilitators,
- one list generated by the meeting participants during the session, and
- one list generated by the agent.

We used the list generated by the meeting participants as a basis for comparison. We asked the facilitators to modify each list until it captured all relevant categories and eliminated all nonrelevant ones. We also asked them to rank each list according to how well it represented the session.

CODING AND ANALYSIS. In the final phase, two experimenters coded the results generated in phase 2. The experimenters independently categorized the facilitator's modifications to each list according to seven variables.

They also calculated intercoder reliability as one of the measures, using a well-known formula to ensure coding consistency.[9] Intercoder reliability for the two experimenters in four sessions was 93.8, 96.3, 94.5, and 93 percent, all exceeding the 80 percent significance threshold adopted in past research.[9]

Results

We used three measures—identified, target, and relevant—to calculate concept recall and concept precision. *Identified* represents the items on the original list. *Target* represents the total items on the list after all additions, deletions, and merges, and so represents the categories the facilitator felt should be on each list. Differences between lists were commonly due to variations in the terms used and the granularity of ideas. *Relevant,* the intersection of the other two measures, indicates the items on the identified list that were also on the target list.

We calculated concept recall and concept precision using a variation[3] of the document recall and precision measures popular in information science applications. Concept recall—the relevant items divided by the target items—represents the percentage of relevant meeting ideas that were properly captured in the original list. Concept precision—relevant items divided by identified items —represents the percentage of concepts on the original list that the facilitators deemed relevant to the meeting topics (taken from phase 2). Values for these two measures were between 0 and 1, with 1 being the most desirable.

BY FACILITATOR AND SESSION. Table 1 summarizes the experimental results by session (within sets A and B) and list generator—experienced facilitators, inexperienced facilitators, original session, and agent. Within a given session, we compared each facilitator-generated list with the agent's list using a two-sample *t*-test. The statistically significant results (10 percent level) are marked with * in the table; the very significant results (five percent level) are marked with **.

In 26 of the 28 cases (the four cases with the original session are not counted), the agent's mean concept recall was not significantly different from any of the individual lists generated by the facilitators, although the scores for the agent's lists were generally poorer. Overall, the agent was able to produce 75 percent concept recall.

As Table 1 shows, however, the agent performed poorly in concept precision, with a mean of 55 percent. In only four cases, the agent's precision level was not significantly different from that of a facilitator-generated list. In all the other cases, the agent lists were significantly worse. Facilitators also consistently ranked the agent's lists lower than all other lists, probably as a result of this poor performance.

BY FACILITATOR EXPERIENCE. Figures 5 and 6 are comparisons of concept recall and concept precision, respectively, grouped by facilitator experience. We had 23 list
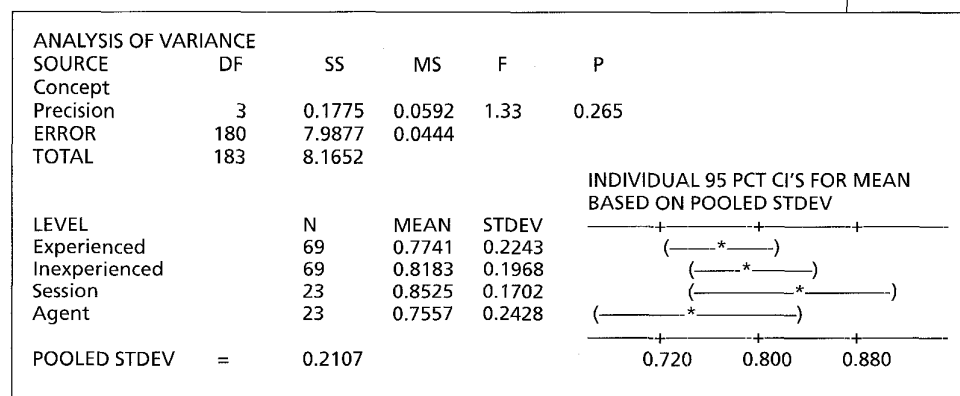
```
ANALYSIS OF VARIANCE
SOURCE           DF      SS      MS      F       P
Concept
Precision        3     0.1775  0.0592  1.33   0.265
ERROR          180     7.9877  0.0444
TOTAL          183     8.1652
                                       INDIVIDUAL 95 PCT CI'S FOR MEAN
                                       BASED ON POOLED STDEV
LEVEL                 N     MEAN   STDEV  ----+--------+--------+--------
Experienced          69    0.7741 0.2243       (------*------)
Inexperienced        69    0.8183 0.1968            (------*------)
Session              23    0.8525 0.1702             (----------*----------)
Agent                23    0.7557 0.2428  (------------*------------)

POOLED STDEV   =    0.2107                ----+--------+--------+--------
                                          0.720    0.800    0.880
```

Figure 5. Concept recall comparison by facilitator experience.

```
ANALYSIS OF VARIANCE
SOURCE           DF      SS      MS      F       P
Concept
Precision        3     1.9765  0.6588  27.72  0.000
ERROR          180     4.2777  0.0238
TOTAL          183     6.2542
                                       INDIVIDUAL 95 PCT CI'S FOR MEAN
                                       BASED ON POOLED STDEV
LEVEL                 N     MEAN   STDEV  ----+--------+--------+--------
Experienced          69    0.8654 0.1479                  (--*-)
Inexperienced        69    0.8533 0.1483                 (--*-)
Session              23    0.8697 0.1453                 (---*---)
Agent                23    0.5480 0.1943  (------*------)

POOLED STDEV = 0.1542                     ----+--------+--------+--------
                                          0.60     0.75     0.90

TWOSAMPLE T FOR Agent vs. Experienced
95 PCT CI FOR MU Agent - MU Experienced: ( -0.4078, -0.2270)
TTEST MU Agent = MU Experienced (vs. NE): T= -7.17 P=0.0000 DF= 30

TWOSAMPLE T FOR Agent vs. Inexperienced
95 PCT CI FOR MU Agent - MU Inexperienced: ( -0.3956, -0.2150)
TTEST MU Agent = MU Inexperienced (vs. NE): T= -6.90 P=0.0000 DF= 31

TWOSAMPLE T FOR Agent vs. Session
95 PCT CI FOR MU Agent - MU Session: ( -0.4240, -0.2195)
TTEST MU Agent = MU Session (vs. NE): T= -6.36 P=0.0000 DF= 40
```
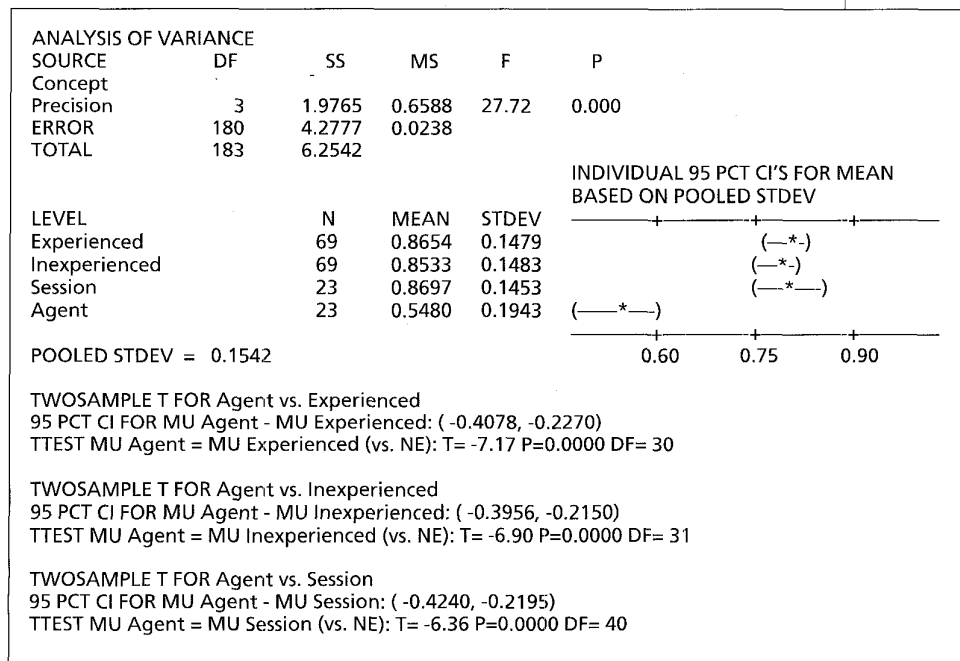
Figure 6. Concept precision comparison by facilitator experience.

evaluations each for the session output and agent output (we had to eliminate one set of evaluations because a facilitator left the company toward the end of the experiment) and 69 evaluations for the facilitators' lists (again because a facilitator departed, we lost three evaluations). We used the Minitab statistical software to perform a one-way analysis of variance on each grouping. Concept recall's *F*-ratio indicated that the results were not significant, so we did not pursue further analysis. However, concept precision's *F*-ratio indicated that the results were significant, so we also performed pairwise, two-sample *t*-tests on the data.

The results confirmed our earlier observations for individual facilitator and session lists. For concept recall, the differences between the agent list and the facilitator lists were statistically insignificant. There was no statistical difference between any of the three kinds of facilitator lists and the agent's lists for concept recall. However, the agent lists were significantly inferior when compared with facilitator lists on concept precision.

We also noted that the performance of experienced and inexperienced facilitators was indistinguishable, both in concept recall and concept precision. This suggests that facilitator experience is not necessarily a good predictor of the ability to categorize ideas.

FOR ALL FACILITATORS. Figures 7 and 8 summarize the performance of all facilitators in concept recall and concept precision, respectively. Once again, the agent's lists were comparable to facilitators' lists in concept recall, but significantly inferior in concept precision. Facilitators were able to achieve 80 percent concept recall (versus the agent's 76 percent) and 86 percent concept precision (versus the agent's 55 percent).

Interpretation

These results bear closer examination.

FACILITATION EXPERIENCE. There are two possible reasons inexperienced facilitators performed as well as experienced facilitators. First, analytical skills and training can influence a person's ability to organize ideas. Second, the four sessions might have been so generic that the idea organization became trivial and did not depend on experience.

LIST GENERATION TIME. The agent generated categories in a fifth of the time a facilitator took. This time savings could be even more significant because the facilitators generated their category lists under ideal conditions—no distractions and no responsibility to manage or facilitate a live meeting. Many subjects commented that they would welcome such a tool in an actual electronic meeting environment. With the agent providing a straw-man category list, meeting facilitators and participants could focus on the more productive and satisfying tasks of generating creative ideas and engaging in group decision making.

QUALITY OF THE SYSTEM LIST. One recurring criticism of the agent's lists was that there were too many general terms that did not contain enough context (meaning) to be considered relevant. Sometimes the categories were too general to be useful and sometimes they were too specific to be considered summaries. The facilitators generally deleted such categories, lowering the agent's concept precision scores. One solution to this problem might be to decrease the length of the agent list, which the user can set at runtime. At present the default is 20 items. Generating a list of 10 to 15 items might improve
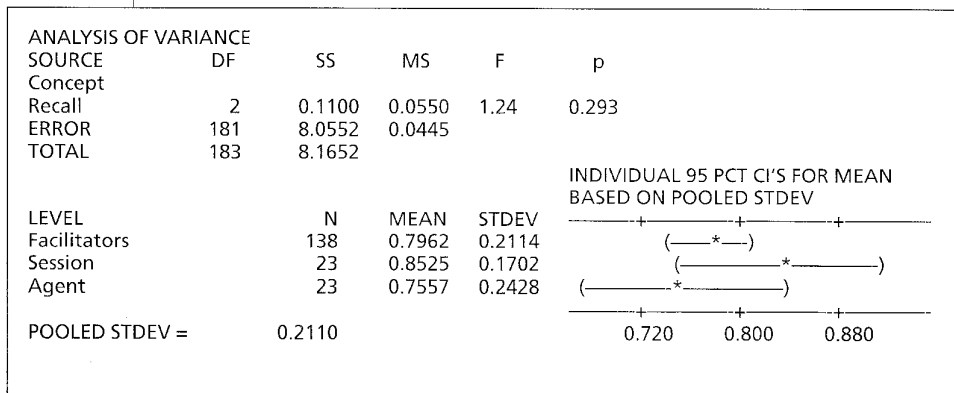
```
ANALYSIS OF VARIANCE
SOURCE          DF      SS       MS       F        p
Concept
Recall          2       0.1100   0.0550   1.24     0.293
ERROR           181     8.0552   0.0445
TOTAL           183     8.1652
                                        INDIVIDUAL 95 PCT CI'S FOR MEAN
                                        BASED ON POOLED STDEV
LEVEL           N       MEAN     STDEV   ------+---------+---------+-------
Facilitators    138     0.7962   0.2114               (----*---)
Session         23      0.8525   0.1702                  (----------*-----------)
Agent           23      0.7557   0.2428   (-----------*------------)

POOLED STDEV =  0.2110                   ------+---------+---------+-------
                                         0.720     0.800     0.880
```

Figure 7. Concept recall comparison for all facilitators.

```
ANALYSIS OF VARIANCE
SOURCE          DF      SS       MS       F        p
Concept
Precision       2       1.9715   0.9857   41.66    0.000
ERROR           181     4.2827   0.0237
TOTAL           183     6.2542
                                        INDIVIDUAL 95 PCT CI'S FOR MEAN
                                        BASED ON POOLED STDEV
LEVEL           N       MEAN     STDEV   -------+---------+---------+------
Facilitators    138     0.8593   0.1477                         (*-)
Session         23      0.8697   0.1453                        (---*---)
Agent           23      0.5480   0.1943   (----*---)

POOLED STDEV =  0.1538                   -------+---------+---------+------
                                         0.60      0.75      0.90
TWOSAMPLE T FOR Agent vs. Facilitators
95 PCT CI FOR MU Agent - MU Facilitators: (-0.3985, -0.2241)
TTEST MU Agent = MU Facilitators (vs. NE): T= -7.34 P=0.0000 DF= 26
```
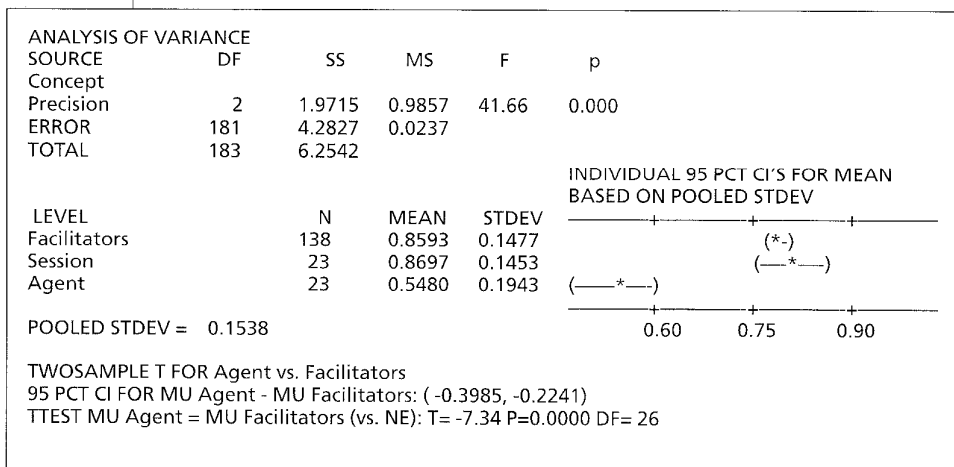
Figure 8. Concept precision comparison for all facilitators.

Computer

precision. On the other hand, recall might suffer. We plan to do more experimentation to determine the effects of system list size.

RECALL VERSUS PRECISON. Although the agent did relatively poorly in concept precision, we do not view this as a major weakness. The reason is that it is easier to refine a list than to generate additional topics for a list that is too short. Concept recall is a far harder and more time-consuming task than concept precision. If we focus on shortening that phase of the meeting, the facilitator and participants can work together to prune and refine the raw list as part of the consensus-building phase of the meeting. The agent's comment analyzer (Figure 4) helps in that work. The facilitators have also indicated that they prefer to have the agent generate long straw-man lists with high recall (but lower precision) as opposed to shorter lists with high precision (but lower recall).

THE AGENT WAS WELL-RECEIVED by facilitators both in our experiments and in subsequent live sessions. In the live sessions, facilitators used the agent to generate a straw-man category list, which the group then inspected, evaluated, and modified. For the most part, the feedback from these sessions has been positive.

We believe the agent addresses all the problems inherent in idea organization:

- *Information overload.* The agent effectively handles large amounts of data, rapidly segmenting the data into smaller, more manageable chunks.
- *Lack of a collaborative vocabulary.* The agent examined participants' vocabularies and was able to produce a consolidated list. The primary problem in arriving at a *collaborative* vocabulary is time. The agent was able to do in minutes what experienced facilitators generally take hours to do.
- *Pressure to synthesize tasks.* The agent produced categories in a fifth of the time it took human facilitators.
- *Sensitive topics and lack of trust.* For better or worse, the agent surfaces any category discussed during the meeting. Participants may also view the agent as an unbiased party. In fact, some participants favored the agent's suggestions over the facilitator's suggestions because of this perception.

Other advantages of the agent are that it is not domain specific and, perhaps most important, that it reduces the cognitive load of organizing ideas. An interesting, unanticipated application of the agent was a modified version of data mining. The client wanted to combine and categorize several months of electronic meeting data to discover ongoing or recurring issues. The agent was able to generate a satisfactory list in just minutes. The facilitators involved had estimated that it would take several hours to accomplish the task.

The agent also has some weaknesses. The three most often given and our suggested improvements are

- *Many categories are too general.* We can give meeting participants access to the stop-word file. This file con-

tains commonly occurring but semantically meaningless words ("a," "or"), which the agent ignores when indexing documents.
- *Single-word descriptors lack content.* We could add another parameter to the agent's program during automatic indexing that would favor multiple-word descriptors over single words.
- *The agent cannot generalize.* We could add access to a domain-specific thesaurus during cluster analysis.

Our immediate work involves designing a graphical, two-dimensional concept clustering agent based on a self-organizing feature map. The agent will analyze meeting output and produce a graphical summary map that users can browse. The system should be more intuitive and thus more suitable for categorizing meeting output. We are also experimenting with applying these techniques to other groupware applications and are examining the use of agent techniques to analyze documents such as Lotus Notes discussions and online newsgroup items. ∎

References

1. P. Maes, "Agents That Reduce Work and Information Overload," *Comm. ACM*, July 1994, pp. 30-40.
2. D. Kirkpatrick, "Here Comes the Payoff from PCs," *Fortune*, Mar. 23, 1992, pp. 93-100.
3. H. Chen et al., "Automatic Concept Classification of Text from Electronic Meetings," *Comm. ACM*, Oct. 1994, pp. 56-73.
4. G. Furnas et al., "The Vocabulary Problem in Human-System Communication," *Comm. ACM*, Nov. 1987, pp. 964-971.
5. H. Chen, "Collaborative Systems: Solving the Vocabulary Problem, *Computer*, May 1994, pp. 58-66.
6. J.F. Nunamaker et al., "Electronic Meeting Systems To Support Group Work," *Comm. ACM*, July 1991, pp. 40-61.
7. H. Chen and K. Lynch, "Automatic Construction of Networks of Concepts Characterizing Document Databases," *IEEE Trans. Systems, Man and Cybernetics*, Sept./Oct. 1992, pp. 885-902.
8. J. Hopfield, "Neural Network and Physical Systems with Collective Computational Abilities," *Proc. National Academy of Science*, National Academy Press, Washington, D.C., 1982, pp. 2,554-2,558.
9. K. Krippendorff, *Content Analysis: An Introduction to Its Methodology*, Sage Press, Newbury Park, Calif., 1980.

Hsinchun Chen is an associate professor of management information systems at the University of Arizona and head of the Artificial Intelligence Group. He has published more than 30 articles covering semantic retrieval, search algorithms, intelligent agents, and collaborative computing. He is also a principal investigator on the Illinois Digital Library project and on the NSF/CISE-funded Internet categorization and search project. Chen received a PhD in information systems from New York University. He has received several research grants in knowledge discovery, intelligent agents, and collaborative computing from the NSF, NIH, NCSA, and AT&T. His recent work on large-scale digital library analysis was featured in Science ("Computation Cracks: `Semantic Barriers' Between Databases," June 7, 1996).

Andrea Houston is a PhD student in management information systems at the University of Arizona. Her research interests are idea convergence in computer-supported cooperative work and content-based browsing and retrieval in digitized video collections. She worked as a programmer and a systems analyst for 15 years before pursuing her degree. Houston received an MBA from the University of New Hampshire. She is a member of the ACM.

Jay Nunamaker is regents professor of management information systems and computer science and director of the College of Business and Public Administration's Center for Management of Information. He established the university's MIS department, served as chairman and department head for 18 years, and developed the BS, MS, and PhD programs. His research interests are group decision-support systems, the automation of systems development, databases, expert systems, systems analysis and design, and strategic planning. He has published more than 200 papers and seven books that deal with these subjects. His work has appeared in Fortune, Business Week, The Wall Street Journal, The New York Times, and Los Angeles Times, as well as on BBC, CNN, and other television networks. Nunamaker received a BS from Carnegie Institute of Technology, an MS in industrial engineering from the University of Pittsburgh, and a PhD in operations research and systems engineering from Case Institute of Technology. He also received the GroupWare Achievement Award in 1993 and the editor's choice award from PC Magazine in 1994.

Jerome Yen is an assistant professor of computer science and coordinator of the Information Systems program at the University of Hong Kong. His research interests include information economics, information retrieval, digital libraries, electronic commerce, group decision and negotiation support systems, intelligent agents, and the Internet. Yen received a PhD in systems engineering from the University of Arizona. He is a member of the ACM and IEEE.

Contact Chen at MIS Dept., University of Arizona, Tucson, AZ 85721; hchen@bpa.arizona.edu.