TURNING UNSTRUCTURED AND INCOHERENT GROUP DISCUSSION INTO DATREE: A TBL COHERENCE ANALYSIS APPROACH

Completed Research Paper

Shasha Deng[1,2]
1. Department of Management
Information Systems, Shanghai
Jiaotong University
2. Department of Computer Science,
Shanghai University of Electric Power
1. No.535 Rd.FaHuaZhen, China
2. No.2103 Rd. Pingliang, China
dengshasha@sjtu.edu.cn

Pengzhu Zhang
Department of Management
Information Systems, Shanghai
Jiaotong University
No.535 Rd.FaHuaZhen, China
pzzhang@sjtu.edu.cn

Yilu Zhou
Information Systems and Technology Management, George Washington
University
Funger, Suite 515, USA
yzhou@gwu.edu

Abstract

Despite the rapid growth of user-generated unstructured text from online group discussions, business decision-makers are facing the challenge of understanding its highly incoherent content. Coherence analysis attempts to reconstruct the order of discussion messages. However, existing methods only focus on system and cohesion features. While they work with asynchronous discussions, they fail with synchronous discussions because these features rarely appear. We believe that discussion logic features play an important role in coherence analysis. Therefore, we propose a TCA method for coherence analysis, which is composed of a novel message similarity measure algorithm, a subtopic segmentation algorithm and a TBL-based classification algorithm. System, cohesion and discussion logic features are all incorporated into our TCA method. Results from experiments showed that the TCA method achieved significantly better performance than existing methods. Furthermore, we illustrate that the DATree generated by the TCA method can enhance decision-makers' content analysis capability.

Keywords: Discussion logic feature, coherence analysis, asynchronous group discussion, subtopic segmentation

Introduction

The Internet has dramatically changed our manner of communication. With users spread around the world, the need to organize group discussion via synchronous or asynchronous methods to support business decisions has increased (Nash 2005). Many customized systems such as group support systems (GSS) allow virtual work groups to participate different discussion regardless of physical boundaries. E-mail, forums and other online communities enable us to collaboratively work to seek solutions and share expertise with colleagues and even strangers at any time. Many institutional websites like blogs, chat rooms and instant message systems allow us to converse with old friends and new people without face-to-face interaction.

In spite of the numerous benefits and convenience of group discussion through the Internet, the fact that such user-generated texts create 'information overload' cannot be ignored (Sheridan and Ferrell 1974). Although computers are good at processing structured data, the ability to process unstructured textual data is still a challenge. Traditional research on text summarization, text categorization and information extraction aids analysis of unstructured data to some extent. However, these discussion texts face a different challenge: they are not only unstructured but also *incoherent* (Zechner 2002). Scholars call this "information entropy," meaning incoming messages are not sufficiently organized by topic and the content cannot be easily comprehended (Hiltz and Turoff 1985). Because the sequence of turns during multi-user group discussion is disturbed in online group discussions, it is more difficult/harder to understand the conversation content. Besides, intensive interactions among a large number of participants complicate the problem.

Current text-based forms of group discussions, including e-mail, newsgroups, forums, chat rooms and GSS, closely resemble spoken interaction. In contrast to spoken interaction, these systems often do not contain the coherent message relationships. Quantitative research experiments suggested that a structured group discussion would improve decision-making outcomes and efficiency (Farnham et al. 2000). Conversation trees are a manifestation of structured information that structure conversations in a more coherent way (Herring 1999). To construct such a tree structure, it is important to understand the sequence of messages and their reply-to relationships (Smith et al. 2000). Some group discussion analysis systems have been developed to address these needs (Viegas and Donath 1999; Sack 2000; Yee 2002; Eklundh and Rodriguez et al. 2004; Fu et al. 2008). However, they generally identified correct turn adjacency by using explicit system features such as header information and quotations. Very little research taps into message body text to gain social cues to improve text analysis capabilities (Fu et al. 2008). Moreover, while system features and social cues work for e-mails, newsgroups and forums, they do not apply to chat rooms and group discussions. However, little research has explored automatic analysis of online group discussion due to its unstructured and incoherent nature.

Some group discussion texts may contain very few system features and social cues that can help structure texts. They are however rich in logic cues such as argument process and utterance emotions (Raghu et al. 2001). In this study, we propose a TBL Coherence Analysis (TCA) system to turn unstructured and incoherent group discussion texts into Discussion Analysis Trees, or DATrees. TBL refers to Transfer-Based-Learning, a machine learning algorithm. DATree reorganizes unstructured discussion texts by automatically discovering subtopics and identifying reply-to relationships between messages. TCA attempts to address limitations in previous studies by utilizing a holistic feature set which is composed of not only explicit linguistic social cues and system features but also implicit discussion logic cues. Combined with various feature types and customized machine learning algorithms, the TCA system is able to capture important and implicit discussion logic cues for better performance. The TCA system is composed of feature extraction, subtopic segmentation and TBL-based classification. Our experimental results on various themes of group discussions showed that the TCA system can transfer multiple implicit features into decision rules and identify the correct reply-to relationship.

The remainder of the paper is organized as follows. We first compare different characteristics of text-based group discussion systems and provide a review of methods developed to support discussion text analysis. We then present the many facets of application in discussion text analysis and describe challenges associated with group discussion text. Subsequent sections provide an overview of our system design and elaborate its components. We then present experimental evaluations of the proposed system. We conclude with our research contributions and future directions.

Related Work

Although the Internet provides us with an unprecedented opportunity for remote group discussions, many studies have expounded upon the significance of analyzing group discussion text to understand online discourse patterns (Chia 2000; Wilson and Peterson 2002; Abbasi and Chen 2008). They emphasized that accurate use of analytic and flexible tools can greatly help people cheaply and handily grasp valuable information from gigantic and wild digital data (Paccagnella 1997). Unfortunately, the unstructured and incoherent nature of discussion texts is a big obstacle to such research. In this section, we describe the core obstacles to analyzing unstructured discussion text, review previous coherence analysis research and analyze the benefits of tree structure in text presentation.

Importance of Coherence in Unstructured Group Discussion Text

In spoken and face-to-face conversations, the sequential structure of turn-taking facilitates coordination among multiple speakers (Clark 1991). However, in online group discussion, the sequence of individual messages is determined by the server's receiving time. The order of turn-taking is disrupted because there are considerable time lags between when a message is sent and when it is responded to (Cherny 1995). With the greater openness of group discussion, the gap becomes bigger and the unstructured degree is higher (McGrath 1990; Kuechler 2007). Herring (1999) pointed out two properties of this medium: "lack of simultaneous feedback" and "disrupted turn adjacency."

Resolving the disrupted turn adjacency problem remains an arduous yet vital endeavor (Fu et al. 2008). In a multi-participant conversation, the "adjacency pairs" structure will contribute to discussion text coherence (Schegloff 1968). Hence, once turn adjacency is disrupted, users may have difficulty tracking message sequence in a computer-mediated environment. Plenty of previous research has observed this phenomenon. Nash (2005) manually analyzed 1,099 turns from Yahoo! Chat and found the gap between a message and its response can be as many as 100 turns. Herring and Nix (1997) safely concluded nearly half of all turns were "off-topic." McDaniel et al. (1996) were surprised at how often thread confusion occurred.

Additionally, group discussions often encompass two or more parallel topics in a conversation. Besides topic fragmentation over time, multiple competing new directions (subtopics) are also characteristic of group conversation (Herring 1999). Group discussion systems define the order of the messages as they appear in the discussion window. As a result, some subtopics are twisted and mixed together confusing participants about which previous message is being referenced (Holmer et al. 2009). Lambiase's (2010) study showed that topic delay exists even in group discussions with a high control of being 'on-topic'. This problem is more severe in synchronous group discussions than asynchronous ones.

Coherence Analysis for Group Discussion Text

Text comprehension involves constructing a coherent mental representation of situations described by the text. Coherence is often used to refer to representation of mental relationships while cohesion represents the textual indications of coherent representations (Louwerse 2002). While cohesion seeks the answer in overt textual signals, a coherence approach considers connectedness to be of a cognitive nature (Sanders and Maat 2006). For web discourse, "coherence is represented in terms of coherence relations between text segments, such as elaboration, cause and explanation" (Barzilay and Elhadad 1997). Fu et al. (2008) further pointed out that coherence of online discourse is represented in terms of *reply-to relationships* between messages. Thus, coherence analysis in group discussion text can be considered as accurately identifying reply-to relationships to construct discussion structure. Various features in communication systems can contribute to coherence analysis.

Various Features in Coherence Analysis

Although almost all forms of group discussion text suffer from disrupted turn adjacency, different forms of discussion text have different degrees of incoherence which require different analysis techniques (Herring 1999). According to participant patterns, the discussion text can be divided into asynchronous and synchronous. Asynchronous discussion text includes e-mail, Usenet newsgroups, forums and blogs. Synchronous discussion text includes chat and customized group discussion systems such as GSS. Our focus is on synchronous group discussions because they are under-researched. We will review features for both asynchronous and synchronous group discussions.

System features. System features are often extracted from asynchronous discussion text. During transmission, e-mails and Usenet newsgroups are required to add header information and specify quoted sections. Using this information, Netscan extract the "contents of Subject, Date, Organization, Lines, MessageID and Reference lines" to generate the relationships in selected newsgroups (Xiong 1998). Smith (2001) found that "when an author posts a reply to a previous message, the new message indicates its parent message by citing the parent's message ID in its own References header." Using these references, they can construct an n-ary tree. By detecting quoted text, Zest divides each message into contiguous blocks of quoted or unquoted text (Yee 2002).

Although header information and quotations are effective in identifying reply-to relationships, it is difficult to generalize. On one hand, not all forms of discussion text contain system features, especially synchronous text. On the other hand, users do not always use the system features to refer to the previous message. Consequently, using system features alone does not solve our problem.

Cohesion features. Cohesion features include social cues and cohesion-based linguistic features. Previous research has suggested some interesting linguistic features to solve the incoherence problem (Halliday and Hasan 1976).

Direct address appears when a reply message includes the screen name of the author of the previous message. Using the direct addressing method, Coterie constructed an interaction network for conversation (Spiegel 2001). *Lexical relation* is defined as a "cohesive relation where one lexical item refers back to another, to which it is related by having common referents" (Nash 2005). It is divided into three subcategories: repetition, synonymy and super-ordinate. Lexical relation has been widely used in previous studies. For example, Choi (2000) used repetition of keywords to identify the message relationship. *Co-reference* describes the situation when a lexical item refers to another lexical item by the context instead of repeating the same item name. Nash (2005) divided co-reference into three subcategories: personal, demonstratives and comparatives. Soon (2001) adopted a machine learning approach to identify co-reference of noun phrases. Bagga (1998) built co-reference chains by CAMP and used the VSM to resolve ambiguities between people with the same names. Other cohesive features such as *conjunctions*, *substitution* and *ellipsis* have rarely been incorporated in previous research due to the difficulty in identification. All of these features rely on linguistic analysis. Asynchronous text is particularly rich in cohesion features, since asynchronous discussion allows users to think and edit their messages with no time pressure (Fu et al. 2008).

Cohesion features provide more evidence than system features. However, it is not without pitfalls. First, some of these features are hard to extract automatically from the message body. Identifying the relation between the antecedent and the anaphor is a complex task in co-reference (Soon et al. 2001). Secondly, using these cohesion features alone cannot identify all reply-to relationships. Lastly, not all group discussion text contains completed cohesion features. Direct address features rarely appear in a synchronous group discussion system, when participants pay more attention to solving the problem under great time pressure (Paul and Nazareth 2010). Moreover, lexical term repetition is directly used in previous studies (Bagga and Baldwin 1998). It is not clear if two messages with reply-to relationship should be the most similar ones.

Discussion logic features. Discussion logic features are decision-making cues and context-based features. They have not been used in previous coherence analysis research but we believe they are especially important in synchronous group discussions. Toulmin's model of discussion revealed the nature of the argument process, especially in tracing the solver's line of discussion (Toulmin 1958). Based on Toulmin's model, Raghu et al. (2001) modeled collaborative decision-making as a dynamic process in which individuals assert their positions through both primitive and derivative statements. A primitive statement is a stand-alone assertion and a derivative is obtained as a strictly logical or defeasible consequence of others. Although an individual could make a primitive assertion, a cogent argument requires the assertions to be linked and organized in some logical sense. So the subsequent logical argument structure proceeds in the form of challenge/response exchanges between its proponents and opponents.

The argumentation process contains the following steps: 1) bring forth a candidate subtopic or solution; 2) demonstrate or refute the subtopic or solution; 3) raise question; and 4) provide some information or fact. This logic can be traced by multi-agent automatic systems (Carbogim et al. 2000). Furthermore, based on/using a simulation experiment for four types of decision problems, Carbogim (2000) found a group discussion can be represented in a two-step process: "first, arguments are generated; then, arguments are evaluated in terms of their acceptability." Obviously, the inner procedure of one group decision discussion is a repetitive process of subtopic or solution-generating and its evaluation. We refer to the inner procedure as discussion logic features. How to automatically identify the proposed subtopic and its evaluation is a big challenge for discussion logic features extraction (Braak et al. 2008). We still believe that discussion logic features can be extracted to help identify the reply-to relationships.

Coherence Analysis Techniques

Using the above-mentioned features except for discussion logic features, many techniques have been used in coherence analysis, especially to construct reply-to relationships. They can be classified into four major categories: manual, linkage, heuristic-based and classification-based methods.

Nash (2005) manually identified some linguistic features, including lexical relations, direct address and co-reference, for online discussions. Barcellini (2005) manually analyzed quotation practice in the online discussions of Open Source Software design which took place in mailing list exchanges. Chesnevar (2006) described specifications for an argument interchange format to represent data exchange between various argumentation tools. Turoff (1999) and Rienks (2005) diagrammed argument process in discussions based on three models. Chesnevar (2006) described specification of an argument interchange format (AIF) to present discussion content based on linguistic words and the Toulmin model. In general this method can achieve high accuracy, but is only possible with small datasets.

Linkage methods build interaction trees with the system features and some predesigned rules (Newman 2002; Yee 2002; Smith and Fiore 2001; Sack 2000). The form of discourse that works well with this method is e-mail-based discussion lists. Some typical systems are Netscan (Usenet group analysis tool, 2001), Conversation Map (2000) and Zest (2002). Obviously, the linkage method is relatively easy to implement, but it requires that users follow system features to post messages and clearly quote messages being responded to.

Heuristic-based methods use both explicit system features and implicit cohesion features to construct a set of rules and matching algorithms to identify the interactional pattern. For example, HIC utilized both system features and linguistic features. Furthermore, several similarity-based methods have been applied to perform lexical match and even residual match (Fu et al. 2008). Murray (2007) proposed to use term-weighting in open group discussion to exploit interactional patterns. By employing pattern recognition using finite state automata, one can automatically identify patterns of interactions with both social and semantic terms in multi-person and multi-topic chat rooms (Khan et al. 2002). This method is effective in text summarization. However it is prone to being affected by unrelated sentences. Also, it only considers sentence semantics and ignores sentence structures.

In classification-based methods, coherence analysis is formulated as a binary classification problem. For example, in order to handle highly incoherent text from student online forums, Kim et al. (2010) used "speech act" features to classify discussion threads. Soon (2001) adopted a machine learning approach to identify the co-reference of noun phrases. Using statistical features and sentence structure information, they showed that machine learning is effective in binary classification. Machine learning methods need a set of handcrafted training documents. Their performances also highly depend on the features used in classification.

Previous research has provided a good foundation of coherence analysis for group discussions. We categorize these important researches in terms of domains, features and techniques in Table 1. Group discussion text is divided into asynchronous and synchronous (D1 and D2). System features, cohesion features and discussion logic features are indicated by F1, F2 and F3. Techniques include manual (T1), linkage (T2), heuristic-based (T3) and classification-based (T4).methods

Table 1. Previous Studies in Coherence Analysis for Group Discussion Text

Previous Studies	Domain		Features			Techniques			
	D1	D2	F1	F2	F3	T1	T2	T3	T4
Turoff et al. 1999		√			√	√			
Sack 2000	√		√	√			√		
Smith and Fiore 2001	√		√				√		
Soon et al. 2001		√		√					√
Yee 2002	√		√				√		
Khan et al. 2002		√		√				√	
Newman 2002	√		√				√		
Barcellini et al. 2005	√		√			√			
Nash 2005		√		√		√			
Rienks et al. 2005		√			√	√			
Chesnevar et al. 2006		√		√	√	√			
Murray and Renals 2007		√		√				√	
Fu et al. 2008	√			√				√	
Kim et al. 2010	√		√	√					√

Tree-like Structure to Support Understanding of Group Discussions

Disrupted adjacency turns leads to chaos in group discussion. Finding reply-to relationships can help transform unstructured group discussion texts into some kind of more structured representation. Using/Based on people's cognition and memory, Schweickert (1993) believes the tree model helps trace the detailed argument process in discussion. Qualitative analysis illustrated that visualization of text structure can show the dynamics of collaboration, disentangle intertwined discussion subtopics and grasp the situation of group discussion at a glance. MuViChat and Mediated Chat are tree-based visualization tools that offer different possibilities to follow and understand group discussion and thereby diminish the chat confusion which often occurs in standard chat systems (Holmer et al. 2009; Fuks et al. 2006). Through analysis of existing visualization tools, such as ArguMed, Convince Me and Reason!Able, research showed that these visualization tools contributed to higher quality discussion and more coherent argument (Braak et al. 2008).

Besides tools that support construction and visualization of a tree-like structure during online group discussions, researchers have also proposed methods to analyze unstructured group discussion afterwards. By capturing connections between turns and replies, a "conversation tree" can present the basic turn-taking structure of human conversations (Smith et al. 2000). Conversation Space and typical Usenet newsgroup browsers use a schematic tree view where a message is linked with replies (Popolov et al. 2000; Smith and Fiore 2001). Conversation Map use tree-like visualization tools to facilitate understanding of social and semantic structure of VLSCs (Sack 2000). Using quotation information in mailing list exchanges, a tree model can reveal links in social structure, critical elements in discussion space and how they shape the OSS design process (Barcellini et al. 2005). Fu et al. (2008) created interaction diagrams for web forums to facilitate enhanced social network and role analysis capabilities.

Research Gaps and Questions

Based on our review, we have identified several important research gaps. Firstly, little coherence analysis research has been conducted for synchronous discussion text. Most existing studies focused on asynchronous discussion text. E-mail and newsgroups contain rich system features to capture user interactions, and forums contain rich cohesion features in the message body to identify reply-to relations. Secondly, previous research failed to take internal discussion logic into consideration during coherence analysis. They focused on system features and cohesion features only. It is questionable to directly apply these features in synchronous discussions such as chat rooms and customized group discussion systems.

Finally, little work has been done to automatically turn synchronous discussion text into a tree structure. Researchers have manually analyzed such discussion text and presented them in a tree-like structure. However, they have not automatically performed coherence analysis. We raise the following research questions:

1. How can we design a system that utilizes not only system and cohesion features, but also discussion logic features in synchronized group discussion coherence analysis?
2. Will discussion logic features improve coherence analysis performance over using system features and cohesion features?
3. What is a good structured representation of synchronous group discussion texts that can potentially enhance one's analysis capability?

System Design

In order to address these research questions, we designed the TBL Coherence Analysis (TCA) system to perform automatic discussion text coherence analysis. The proposed system has two major components: *Subtopic Segmentation* and *Reply-To Relationship Identification* (as shown in Figure 1). Reply-to relationship identification contains two sub-components: Message Feature Extraction and Classifier Construction. The feature extraction stage acquires various system features, linguistic features and discussion logic features from the group discussion messages. Discussion logic features relies on input from subtopic segmentation. All extracted features are passed forward to the classifier construction stage, which involves creation of a TBL classifier for candidate message pairs and then utilizes the predefined rules to determine residual coherence relationships. Finally, the unstructured group discussion messages are formatted to a discussion analysis Tree (DATree).

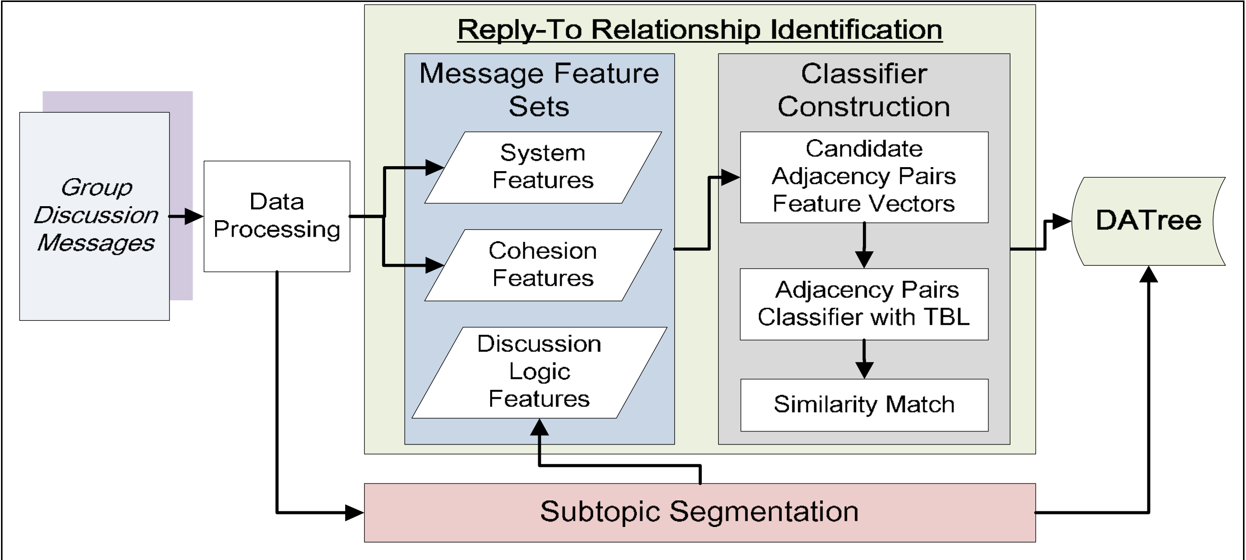

Reply-To Relationship Identification
Message Feature Sets
System Features
Cohesion Features
Discussion Logic Features
Classifier Construction
Candidate Adjacency Pairs Feature Vectors
Adjacency Pairs Classifier with TBL
Similarity Match
Group Discussion Messages
Data Processing
DATree
Subtopic Segmentation

Figure 1. TBL coherence analysis system design

Definition

Since there are no unified names for certain phrases in GSS and the social media field, it is necessary to provide some definition to avoid confusion in following our system design description.

Definition 1: *message* is one utterance or post. One message can include many sentences posted by an individual author at one time.

Definition 2: *discussion* is one discussion based on a specific topic. In group discussions, there can be different discussions at the same time based on the same or different topics. A "*discussion"* refers to one of these group discussions.

Definition 3: *subtopic* defines a proposal or claim or argument in one discussion. In other words, one discussion has one or more subtopics.

Data Processing

Before we perform any analysis, the data processing component is designed to extract message bodies and their related meta-information from Group Support Systems. All relevant property information is extracted and tagged with corresponding symbols. In order to perform deep text analysis, the message body's syntactic structure is parsed by a tool called Language Technology Platform (LTP) developed by the Harbin Institute of Technology (http://ir.hit.edu.cn/demo/ltp/Sharing_Plan.htm). The parser is good at lexical analysis including word segmentation and part-of-speech (POS) tagging based on dependency grammar. The parser has also showed good performance in distinguishing noun compounds which could greatly influence information extraction accuracy. Finally, all group discussion messages are labeled by property information and lexical structures, such as author screen name, time stamps and other lexical relationships. This step prepares data for system and cohesion feature extraction as well as subtopic segmentation.

Message-Level Feature Extraction

The extraction stage involves derivation of structural, linguistic and discussion logic features. Table 2 provides a detailed description of such categories. System features capture static attributes such as current message ID (curMsgID), message's author (viewer) and message post time (viewTime). Cohesion and discussion logic features are more complicated and are derived from the message body. We describe them in more detail.

Table 2. Message Feature Sets

Group	Category	Type	Description
System feature	curMsgID	Number	The current message's ID number
	viewer	Number	The message's author
	viewTime	Number	The message post time
Cohesion feature	MsgSimList	Vector	List the similarity between the current message and the other messages
	msgSentiment	True/False	True if the message includes user's subjective sentiment. False otherwise.
Discussion logic feature	subTopicID	Vector	List all the subtopics of one discussion

Cohesion Feature Extraction

Two cohesion features are extracted here: similarity feature (MsgSimList) measures the similarity between the current message and all other messages and sentiment feature (msgSentiment) identifies if the message is subjective or objective.

Message Semantic Similarity Measure. Being able to identify semantic similarity between every two messages serves two purposes. It is used as a cohesion feature to be included later in the classification algorithm. Meanwhile, it is also used in the subtopic segmentation component. In order to measure the semantic similarity between messages, we propose a Revised cosine Similarity Algorithm (RSA). RSA improves cosine similarity measurement in three ways. First, it incorporates POS tagging information into a Vector Space Model. Research has shown that noun phrases and verb phrases carry most of the important meaning in a sentence, while conjunction, adverbs and adjectives are less important. Thus, we define meaningful keywords to be noun, noun compound, named entity, verb and verb phrase. Instead of taking into consideration every single keyword, we only focus on these meaningful keywords. We represent all keywords in a message using the Vector Space Model (VSM), one of the most popular methods to identify lexical-level similarity (Salton and McGill 1986). Secondly, a traditional VSM uses term frequency count to represent a document. We improve it by using term frequency and inversed document frequency: *tfidf*. *tfidf* is a measure derived from Information Retrieval research. It measures the importance of a term to a message in an entire collection. This will downgrade some of the common words that are less important. Finally, in group discussion text, users tend to use different words to express the same meaning (Nash 2005). A traditional VSM will treat these synonyms as two different entries and this will result in a zero similarity value. We further modify VSM by calculating a term-level similarity value $coef_k$ and incorporate the value into our *tfidf* matrix. When all the terms in two messages are different, but there are synonyms or terms that are semantically related, these two messages will still have some level of similarity. The similarity score between a pair of messages X, Y is computed using RSA similarity, shown in Equation 1.

$$RSim(X,Y) = \frac{\sum_{k=1}^{n} Sw_{xk} * Sw_{yk}}{\sqrt{\sum_{k=1}^{n} Sw_{xk}^2 * \sum_{k=1}^{n} Sw_{yk}^2}} \quad (1)$$

Where $$Sw_{yk} = tf_{yk} * idf_k * coef_k \quad (2)$$

$$Sw_{xk} = tf_{xk} * idf_k \quad (3)$$

RSim(X, Y) is the similarity score between message X and message Y. Sw_{yk} is the modified *tfidf* of K^{th} dimension in message Y, as shown in Equation 2. $coef_k$ is the similarity value between two terms and is calculated by an "Xsimilarity" tool based on HOWNET and CILIN (http://code.google.com/p/xsimilarity/) thesauri. Because we select the most similar term from Y as a dimension based on terms in message X, $coef_k$ does not need to be considered in Equation 3. We calculate the RSim score for all message pairs in the entire discussion text and the resulting vector is called MsgSimList. It is used as one of our cohesion features.

Sentiment Detection and Computation. During group discussion, it is very popular for a user to express his opinion about the previous message with simple sentences such as "Yes, I agree with you" and "No, your point is wrong." Such a message often does not include enough useful words for deep semantic analysis. However, identifying the sentiment of these sentences will help to find the reply-to relationship. Therefore, in the sentiment detection and computation step, each term in a message is compared to a set of sentimental keywords from a sentiment dictionary based on HOWNET. Using a simple sentiment detection method, messages with obvious subjective sentiment can be identified and stored in the system's msgSentiment feature.

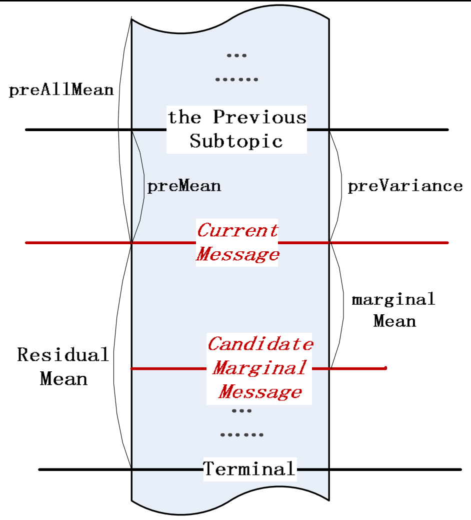
Discussion Logic Feature Extraction and Subtopic Segmentation

It is obvious that disrupted turn adjacency is caused by the high degree of subtopics intertwined in discussions. Identifying subtopics in a discussion not only helps to structure the discussion text, but also serves as input for discussion logic features. Subtopic segmentation assumes that a subtopic is always proposed in advance of discussions of that subtopic. Our rationale is that *a subtopic is a message that has low relevance to all previous messages in a previous subtopic and has high relevance to all following messages till the end of that subtopic*. Relevance of two messages is measured by the MsgSimList feature described in the previous section. Figure 2 illustrates a subtopic segmentation scenario and definition of variables. The steps for our subtopic segmentation algorithm are provided below. Our rationale is to use some statistical measures such as mean and variance between messages to predict a true subtopic:

1. Initialize the subtopic list
   If preAllMean = 0, Then set current message = new subtopic
2. Identify candidate subtopic
   If (distance from the previous subtopic to this current message <2) (preAllMean < msgMean), Then current message = candidate subtopic

If (distance from the previous subtopic to this current message >=2) and (preMean < msgMean),
Then current message = candidate subtopic

3. Find marginal message based on candidate subtopic
   We define marginalMeanSet as collection of marginalMean.
   Select the maximum similarity score from marginalMeanSet. If its marginalMean is greater than residualMean, then mark the corresponding message as marginal message.
4. Find subtopic
   If (preVariance > msgVariance) and (marginalMean > msgMean)
   Then candidate subtopic = new subtopic;
   ElseIf marginalMean > msgMean and the maximum in candidate subtopic's MsgSimList is not generated with previous message, then candidate subtopic = new subtopic



Definition of Variables

msgMean : average similarity score between current message and all other messages
preAllMean : average similarity score between current message and all previous messages
preMean : average similarity score between current message and all previous messages after the previous subtopic
residualMean : average similarity score between current message and all remaining messages
marginalMean : average similarity score between current message and all remaining messages before the next candidate marginal messages
msgVariance : variance of similarity score between current message and all other messages
preVariance : variance of similarity score between current message and all previous messages after the previous subtopic

Figure 2. The relationship among variables in subtopic segmentation algorithm

Classifier Construction

Once all message features are extracted, we adopt a corpus-based machine learning approach to identify reply-to relationships. We model reply-to relationship identification as a binary classification problem. For each candidate adjacency message pair, they either construct a true reply-to relationship or not. The classifier construction includes three stages. We first transform message features into candidate adjacency pair features. We then present the TBL classifier. The residual matching mechanism is developed for remaining unpaired messages. It improves performance by matching messages based on similarity and discussion logic features.

Transforming Message Features into Candidate Adjacency Pair Features

To build a learning-based classifier, we need to derive features between candidate adjacency pairs from the message feature set. We use five features described in Table 3. For any message pair (*i*, *j*), suppose *i* precedes *j*.

Table 3. Features of candidate message pair

Feature	Categorical Value	Description
DistRange	0-5	The distance of the message pair
SimDegree	0-5	The similarity degree of the message pair
SentiOrNot	0-3	Whether or not two messages include sentiment polarity
SubtopicOrNot	0-3	Whether or not two messages are subtopic messages
LogicPosition	0-7	The logic position of the two messages in the discussion

The distance of each message pair is derived from a system feature, the message's ID number. These continuous numbers are dispersed from 0 to 5 degrees, denoted as DistRange. SimDegree and SentiOrNot are derived from linguistic features. The SimDegree feature is represented by six degrees from 0 to 5, with 5 being the most similar. In message-level features, msgSentiment captures the sentiment polarity of a message. To derive the SentiOrNot feature, we assign the value of a pair according to the 4 possible combinations: both contain subjective sentiment, both do not, or one contains subjective sentiment (two possibilities with either i or j).

SubtopicOrNot and LogicPosition are derived from discussion logic features. SubtopicOrNot determines whether the two messages are subtopic messages or not. With the LogicPosition feature, we assign the value based on whether the two messages exist in the same subtopic region and the number of subtopics between them. The discussion logic feature not only obtains the subtopic message but also subtopics' corresponding marginal messages. The subtopic region refers to the range from subtopic message to marginal message.

Adjacency Pair Classifier with TBL

The above features are fed into the/our transformation-based learning (TBL) classifier. TBL has been successfully applied in spoken dialogue act classification (Samuel 2000; Brill 1995). It starts with the unlabeled corpus and learns the best sequence of suitable "transformation rules" that can be applied to the training corpus to get better performance for the task. In our approach, each candidate adjacency pair is represented as a feature vector. One advantage of TBL is that the generated rules are easy to understand.

Each rule derived from TBL is composed of two parts: combination of features as the condition and reply-to relationship tag as the classification result. For example, "1 || 0 || || 0 || 2 || => 1" is a rule. The five feature value is separated by "||". This rule means: if DistRange=1 and Position=0 and SentiOrNot=0 and SubtopicOrNot=2, the resulting relationship is a true reply-to relationship.

Assigning Reply-to Relationships for Remaining Pairs

Not all of the messages are taken care of in rules generated by TBL. For those remaining messages that are not subtopics, we use a recursive similarity matching method to identify reply-to relationships. Prior match method is intuitive method which assigns each remaining post to the prior one. This method tends to have lower precision in GSS. We proposed a new recursive similarity matching method that considers the message's similarity and discussion logic. The recursive process for our similarity match is provided below:

Define X as the residual message sequence in chronological order. Define *Xi* as the *i*-th element of X. Define Y-Set as the set of entire identified messages with root Y.

1) Initialize R is topic of this discussion and its first-level messages is subtopics.
2) Treat the message R as root and then split the tree structure into subTree by R message. Compute similarity value between Xi and the first-level messages. And then mark the message with the highest value as Y.
3) If the Y is a leaf or the similarity value between Xi and Y is the largest value in Xi's MsgSimList, then Y is Xi's antecedent message.
4) Else, mark Y as R and goto 2).

Evaluation

Testbed and Dataset

Our proposed TCA system is a language-independent method. It has several language dependent components such as POS tagging and noun phrase identification. With the development in computational linguistics, these parsing tools are available not only in English but also in other popular languages. We evaluate TCA using a Chinese group discussion testbed. Chinese poses some special challenges in Natural Language Processing tasks. From a linguistic perspective, Chinese is a language without explicit word boundaries. It uses fewer function words and applies a looser grammar structure compared with English. These characteristics of Chinese create more ambiguities during parsing of syntax structure than English (Levy et al. 2003). Therefore, the accuracy of Chinese text parsing is often lower and Chinese parsing tools are also less developed. However, as the second most popular language on the Internet and with almost 5 times the growth rate of English, it is believed that Chinese will become the most popular language on the Internet in the future. Thus, we evaluate the TCA system on a Chinese group discussion testbed described in a prototype GSS system (Li et al. 2009). To record the true reply-to relationships, the GSS system is defined six categories according to Toulmin's argument process: Question, Answer, Agreement, Objection, Explanation and New Topic. A new topic does not respond to any of the antecedent messages. The other five categories indicate different sentiments or argument processes in a certain subtopic. During the discussion process, users are required to select a previous message as the antecedent message and specify the purpose of the message from the five categories. If he is posting a new topic, there will be no previous message selected. We use these user-generated reply-to relationship tags as the "gold standard."

We recruited 80 subjects who are doctoral and master's students of Management Information Systems at a university in Shanghai, China. They all had experience with GSS-supported electronic meetings and knew how to properly use the GSS software. They were divided into 20 groups to discuss the topic "how to address the overproduction problem for a tea bag manufacturer." Subjects were told that they need to discuss the solutions to this overproduction problem and reach a business decision in thirty minutes. We obtained a total of 20 group discussion texts and each consists of input from four participants. The maximum number of messages in all 20 discussions is 71, the minimum is 22 and the average is 42. We observed that linguistic features such as direct address and co-reference are not popular in these messages.

Experiment 1: Subtopic Segmentation

Experimental Design

In the first experiment, we evaluated the effectiveness of our similarity-based subtopic segmentation method. Subtopic segmentation is an important component in our TCA system. It contributes to the tree structure and serves as the discussion logic feature. Our subtopic segmentation is based on our innovative RSA similarity measure, revised VSM and statistical calculation. We compare our approach with a traditional text clustering approach. The benchmark clustering approach uses the similarity scores derived by traditional VSM between these messages and the Euclidean Distance measure. Precision, recall and F-measure were used as our performance measures. They are defined as follows:

$$precision = \frac{tp}{tp + fp},\ recall = \frac{tp}{tp + fn} \text{ and } F - measure = \frac{2 \times precision \times recall}{precision + recall}$$

where *tp* is the number of correctly identified subtopics, *fp* is the number of identified subtopics that are not true subtopics and *fn* is the number of true subtopics that are missed.

Hypothesis

We propose the following hypothesis:

H1: Using an innovative RSA similarity measure, our topic segmentation algorithm will improve the performance compared to ordinary clustering methods.

Experimental Results and Discussion

Table 4 shows the experimental results. Our algorithm has better performance on the 20 discussions. Obviously, TCA subtopic segmentation achieves much higher precision, recall and F-measure than that of traditional text clustering. Our average precision is 0.653, average recall is 0.738 and F-measure is 0.677. Figure 3 shows the F-measure performance of the two methods on all 20 discussion texts. The subtopic segmentation method outperformed the clustering algorithm on every dataset. We believe that the subtopics selected by the text clustering method are highly similar to their classes in terms of lexical similarity. We believe two factors contributed to our superior performance. The traditional categorization method failed to differentiate between different types of terms that might hurt the similarity measure. Also, the benchmark categorization method performs an overall similarity measure and failed to capture the discussion logic sequences. We also observed two discussion texts with 100% precision and recall using similarity-based subtopic segmentation. The experimental results demonstrated that considering the discussion logic process is necessary in subtopic segmentation. Our H1 is supported.

Table 4. Experimental results for Experiment 1

Technique	Precision	Recall	F-measure
TCA subtopic segmentation	0.653	0.738	0.677
Clustering algorithm	0.332	0.269	0.283

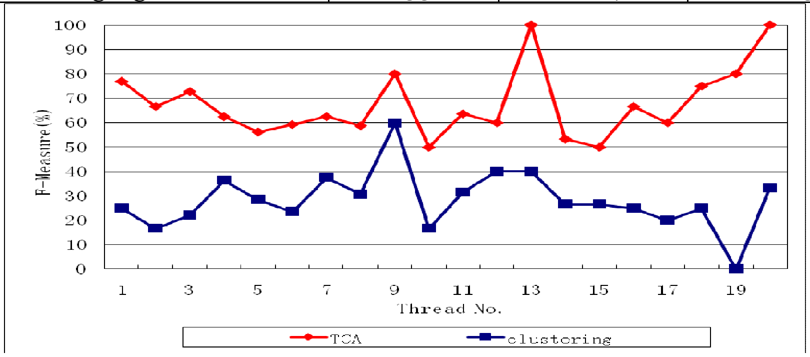


Figure 3. Experiment 1 F-measure performances for each discussion

Experiment 2: Comparison of Techniques in Reply-to Relationship Identification

Experiments Setup

In the second experiment, we compared our TCA classification method with several benchmark methods:

1) Linkage method based on system features obtained from message heading. Since most synchronized GSS do not provide quotation function, we only used the timestamp in each message header as system features. This method assumes that each message replies to the previous message. In a tree construction, this method will construct a tree with chronologically linked leaves.

2) Heuristic-based method that relies on cohesion features in the message body. The heuristic-based method uses three features: direct address match, lexical similarity match and residual match. Direct address match identifies reply-to relationships based on matching author screen names. The second part uses the "Xsimilarity" tool to compute lexical similarity and then obtains the similarity score between messages using traditional VSM. In the last part, we applied rules defined in (Fu et al. 2008) for messages without obvious above cues.

3) Classification-based method using syntactic features and system features. These benchmark techniques have been adopted in previous studies with web forums (Kim et al. 2010). However, no discussion logic features were used. The classification-based method represents the coherence analysis as a binary classification problem. Each message makes a pair with all previous messages. We extract four types of features from the message pairs. These features include "time_gap" and "dist" which are the intervals of time and distance between message pairs, respectively, "repeatNoun" which is the number of repeated nouns between message pairs and "viewer_timeGap" which examines the messages pairs from the same author and with time interval less than five seconds. Again, precision, recall and F-measure were used as our evaluation criteria.

Hypotheses

We believe that discussion logic features can play an important role when discussion text lacks system and cohesion features. Thus, our TCA method that incorporates all three types of features is likely to obtain better performance. We propose the following hypotheses:

H2a: TCA algorithm will outperform the linkage method for GSS-based coherence analysis.

H2b: TCA algorithm will outperform the heuristic-based method for GSS-based coherence analysis.

H2c: TCA algorithm will outperform the classification-based method for GSS-based coherence analysis.

Experimental Results and Discussion

Table 5 shows the experimental results for all four methods. The TCA method achieved the best performance in terms of precision, recall and F-measure. The F-measure was 20-50% higher than the other three methods. Such improved performance was consistent across all 20 discussions, as depicted in Figure 4. The heuristic-based method performed better than the other two methods. Surprisingly, the traditional classification-based method performed worst and F-measure is less than 0.1.

Table 5. Experimental results for Experiment 2

Technique	Precision	Recall	F-measure
TCA method	0.509	0.509	0.509
Linkage method	0.166	0.166	0.166
Heuristic-based method	0.296	0.296	0.296
Classification-based method	0.271	0.05	0.082

The TCA method performed best among the four methods for our GSS-based testbed. And the superior performance was consistent across all 20 discussions, as depicted in Figure 4. This is due to the introduction of discussion logic features and the composition of advanced linguistic features. We observed that the enhanced accuracy of subtopic segmentation helped the performance of reply-to relationship identification, as illustrated in the performance comparison chart in Figure 3. Although both the TCA method and traditional classification-based method use the same machine learning algorithm, TBL, the results from TCA were significantly better. This is due to the additional discussion logic features used in the TCA method. In the traditional classification-based method, system features and simple linguistic features do not represent complete characteristics in identifying reply-to relationships. The linkage method only linked messages in chronological order. Our results showed that in synchronous group discussions, relying on system features alone does not help structure discussion texts. Meanwhile, because discussion texts lack interactional cues, only lexical similarity contributed to the heuristic-based method. The performance is better than all three other benchmark methods, but still significantly lower than TCA. The experimental results demonstrate the impact of decision logic features and complicated linguistic features on group discussion coherence analysis as well as effective application of the two features in the TCA method.

Synchronous group discussion can support problem solution by a set of decision-makers working together as a group. The process of decision discussion focuses on analyzing the problem itself instead of interaction among participants. And the object of utterance represents the issue about discussion topic not care about who is spokesman. Consequently, personal behavior with characteristic of communication fails to identify reply-to relationship. The experimental results support this point of view.

Despite achieving superior performance by incorporating many complex and advanced computational linguistic theories into TCA method, the performance of coherence analysis in synchronous group discussion is still lower than in asynchronous group discussion. In the study by Fu et al. (2008), they achieved an F-measure of 0.76 while we only achieved 0.509. Language difference could also contribute to the lower performance. But we believe that this is mainly due to the unique challenges in synchronous group discussion. The corpus of this research includes more than70% interaction features, which means coherence analysis in synchronous group discussion is a more difficult problem.

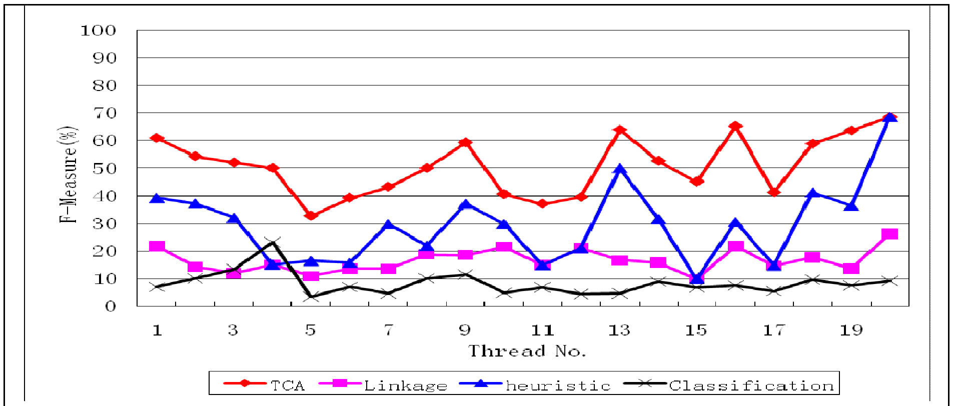


Figure 4. Experiment 2 F-measure performances for each discussion

Experiment 3: DATree Representation

After identifying the subtopics and reply-to relationships, we can represent the original unstructured group discussion text in a discussion analysis tree structure, which we call a DATree. DATree can enhance content analysis capability. Important subtopics are floated as top branches, sequence of discussion is corrected and the phenomenon of disrupted adjacency turns can be avoided. In Figure 5 we present an example of a DATree constructed by the TCA method on the right in comparison with the original unstructured text on the left. DATree provides more clear visualization in contrast to the unstructured view. From the general tree structure, it is easy to tell the number of discussion points (subtopics), how thoroughly each topic is discussed, what topics attracted intensive discussions and what topics are not well elaborated. Discussion details are also presented as tree leaves. Our example DATree was constructed with Chinese texts, but the techniques can be generalized to other languages. Although DATree was proposed for the analysis of synchronous discussion text, it can be applied in asynchronous text analysis as well.

Conclusions

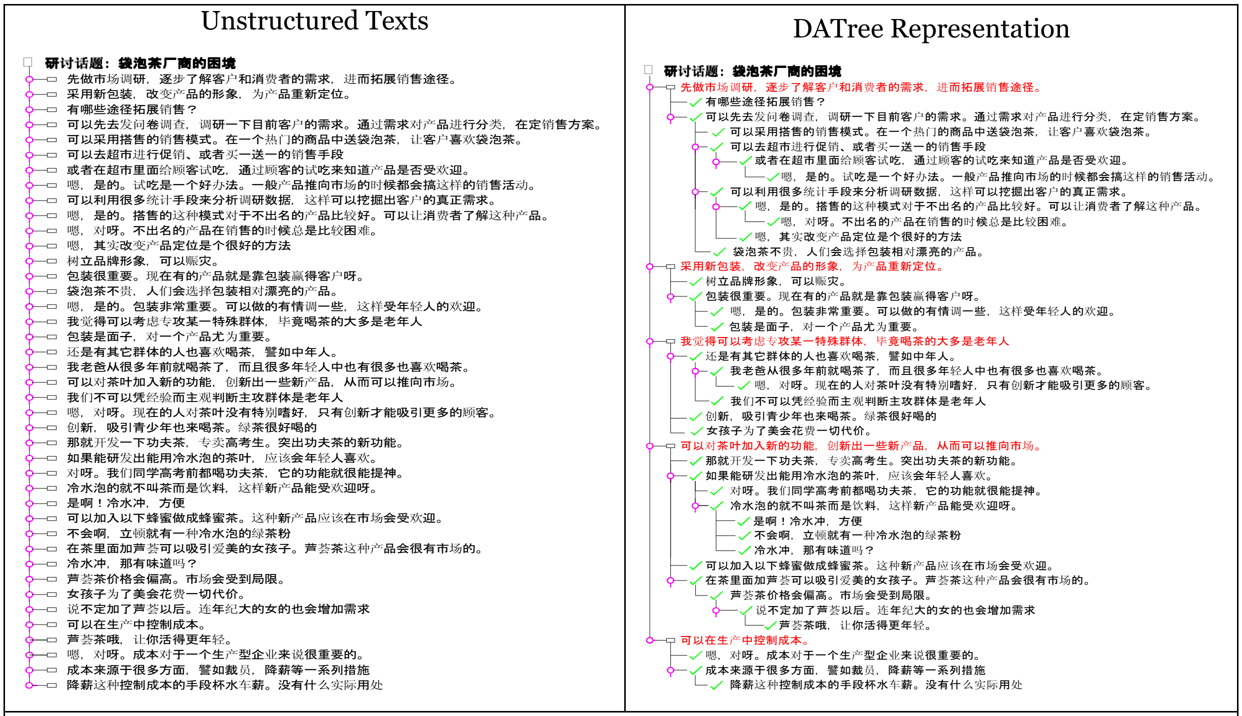
In this study we applied coherence analysis to GSS discussion. Our research has several contributions. First, we propose the TBL Coherence Analysis (TCA) method to identify reply-to relationships in synchronous GSS discussions. GSS discussions generate the most unstructured texts among all CMC environments TCA is unique in two ways. Conceptually, this method is based on Toulmin's model (Toulmin 1958) where internal discussion logic is considered. Discussion logic features are divided into two categories: "hard" features such as a subtopic sequence and its related messages sets, and the "soft" feature which is the soul of the TCA method. In order to automatically detect the "hard" features from unstructured discussion text, a novel subtopic segmentation algorithm is introduced. The "soft" feature places great emphasis on discussion logic. Thus, similarity match assigned the reply-to relationship not only based on the highest similarity between pairs but also based on subtopic and child node relationships.

On the technical side, to measure the similarity between two messages, we revised traditional VSM by taking into consideration *tfidf*, POS information and word-level similarity. We incorporated system features, cohesion features and discussion logic features in our TCA classification method. The last step in the TCA method is a recursive process to take care of any remaining message pairs.

The results showed that the subtopic segmentation algorithm outperformed the traditional clustering algorithm in terms of precision, recall and F-measure. Our TCA classification method outperformed traditional linkage, heuristic-based and traditional classification-based methods in reply-to relationship identification. The results revealed that discussion logic features can greatly improve the performance of coherence analysis. Furthermore, the use of discussion logic features is promising for extremely unstructured user-generated discussion text where system features and cohesion features are less effective.

Finally the DATree generated from the TCA method demonstrated the advantage of turning unstructured discussion text into a structured tree format. We believe that such a visualization tool will greatly help decision-makers obtain valuable information from massive user-generated discussion texts in an effective fashion.

In the future we will test our TCA method with different topic discussion text. We are also interested in analyzing the participant behavior in group discussion. We plan to apply the method to analyze other forms of user-generated data.



Unstructured Texts

□ **研讨话题：袋泡茶厂商的困境**

- 先做市场调研，逐步了解客户和消费者的需求，进而拓展销售途径。
- 采用新包装，改变产品的形象，为产品重新定位。
- 有哪些途径拓展销售？
- 可以先去发问卷调查，调研一下目前客户的需求。通过需求对产品进行分类，在定销售方案。
- 可以采用搭售的销售模式。在一个热门的商品中送袋泡茶，让客户喜欢袋泡茶。
- 可以去超市进行促销、或者买一送一的销售手段
- 或者在超市里面给顾客试吃，通过顾客的试吃来知道产品是否受欢迎。
- 嗯，是的。试吃是一个好办法。一般产品推向市场的时候都会搞这样的销售活动。
- 可以利用很多统计手段来分析调研数据，这样可以挖掘出客户的真正需求。
- 嗯，是的。搭售的这种模式对于不出名的产品比较好。可以让消费者了解这种产品。
- 嗯，对呀。不出名的产品在销售的时候总是比较困难。
- 嗯，其实改变产品定位是个很好的方法
- 树立品牌形象，可以赈灾。
- 包装很重要。现在有的产品就是靠包装赢得客户呀。
- 袋泡茶不贵，人们会选择包装相对漂亮的产品。
- 嗯，是的。包装非常重要。可以做的有情调一些，这样受年轻人的欢迎。
- 我觉得可以考虑专攻某一特殊群体，毕竟喝茶的大多是老年人
- 包装是面子，对一个产品尤为重要。
- 还是有其它群体的人也喜欢喝茶，譬如中年人。
- 我老爸从很多年前就喝茶了，而且很多年轻人中也有很多也喜欢喝茶。
- 可以对茶叶加入新的功能，创新出一些新产品，从而可以推向市场。
- 我们不可以凭经验而主观判断主攻群体是老年人
- 嗯，对呀。现在的人对茶叶没有特别嗜好，只有创新才能吸引更多的顾客。
- 创新，吸引青少年也来喝茶。绿茶很好喝的
- 那就开发一下功夫茶，专卖高考生。突出功夫茶的新功能。
- 如果能研发出能用冷水泡的茶叶，应该会年轻人喜欢。
- 对呀。我们同学高考前都喝功夫茶，它的功能就很能提神。
- 冷水泡的就不叫茶而是饮料，这样新产品能受欢迎呀。
- 是啊！冷水冲，方便
- 可以加入以下蜂蜜做成蜂蜜茶。这种新产品应该在市场会受欢迎。
- 不会啊，立顿就有一种冷水泡的绿茶粉
- 在茶里面加芦荟可以吸引爱美的女孩子。芦荟茶这种产品会很有市场的。
- 冷水冲，那有味道吗？
- 芦荟茶价格会偏高。市场会受到局限。
- 女孩子为了美会花费一切代价。
- 说不定加了芦荟以后。连年纪大的女的也会增加需求
- 可以在生产中控制成本。
- 芦荟茶哦，让你活得更年轻。
- 嗯，对呀。成本对于一个生产型企业来说很重要的。
- 成本来源于很多方面，譬如裁员，降薪等一系列措施
- 降薪这种控制成本的手段杯水车薪。没有什么实际用处

DATree Representation

□ **研讨话题：袋泡茶厂商的困境**

- 先做市场调研，逐步了解客户和消费者的需求，进而拓展销售途径。
  - ✓ 有哪些途径拓展销售？
  - ✓ 可以先去发问卷调查，调研一下目前客户的需求。通过需求对产品进行分类，在定销售方案。
    - ✓ 可以采用搭售的销售模式。在一个热门的商品中送袋泡茶，让客户喜欢袋泡茶。
    - ✓ 可以去超市进行促销、或者买一送一的销售手段
      - ✓ 或者在超市里面给顾客试吃，通过顾客的试吃来知道产品是否受欢迎。
        - ✓ 嗯，是的。试吃是一个好办法。一般产品推向市场的时候都会搞这样的销售活动。
    - ✓ 可以利用很多统计手段来分析调研数据，这样可以挖掘出客户的真正需求。
      - ✓ 嗯，是的。搭售的这种模式对于不出名的产品比较好。可以让消费者了解这种产品。
        - ✓ 嗯，对呀。不出名的产品在销售的时候总是比较困难。
      - ✓ 嗯，其实改变产品定位是个很好的方法
    - ✓ 袋泡茶不贵，人们会选择包装相对漂亮的产品。
- 采用新包装，改变产品的形象，为产品重新定位。
  - ✓ 树立品牌形象，可以赈灾。
  - ✓ 包装很重要。现在有的产品就是靠包装赢得客户呀。
    - ✓ 嗯，是的。包装非常重要。可以做的有情调一些，这样受年轻人的欢迎。
    - ✓ 包装是面子，对一个产品尤为重要。
- 我觉得可以考虑专攻某一特殊群体，毕竟喝茶的大多是老年人
  - ✓ 还是有其它群体的人也喜欢喝茶，譬如中年人。
    - ✓ 我老爸从很多年前就喝茶了，而且很多年轻人中也有很多也喜欢喝茶。
      - ✓ 嗯，对呀。现在的人对茶叶没有特别嗜好，只有创新才能吸引更多的顾客。
    - ✓ 我们不可以凭经验而主观判断主攻群体是老年人
  - ✓ 创新，吸引青少年也来喝茶。绿茶很好喝的
  - ✓ 女孩子为了美会花费一切代价。
- 可以对茶叶加入新的功能，创新出一些新产品，从而可以推向市场。
  - ✓ 那就开发一下功夫茶，专卖高考生。突出功夫茶的新功能。
  - ✓ 如果能研发出能用冷水泡的茶叶，应该会年轻人喜欢。
    - ✓ 对呀。我们同学高考前都喝功夫茶，它的功能就很能提神。
    - ✓ 冷水泡的就不叫茶而是饮料，这样新产品能受欢迎呀。
      - ✓ 是啊！冷水冲，方便
      - ✓ 不会啊，立顿就有一种冷水泡的绿茶粉
      - ✓ 冷水冲，那有味道吗？
  - ✓ 可以加入以下蜂蜜做成蜂蜜茶。这种新产品应该在市场会受欢迎。
  - ✓ 在茶里面加芦荟可以吸引爱美的女孩子。芦荟茶这种产品会很有市场的。
    - ✓ 芦荟茶价格会偏高。市场会受到局限。
      - ✓ 说不定加了芦荟以后。连年纪大的女的也会增加需求
        - ✓ 芦荟茶哦，让你活得更年轻。
- 可以在生产中控制成本。
  - ✓ 嗯，对呀。成本对于一个生产型企业来说很重要的。
  - ✓ 成本来源于很多方面，譬如裁员，降薪等一系列措施
    - ✓ 降薪这种控制成本的手段杯水车薪。没有什么实际用处

Figure 5. DATree of dataset #13.

Acknowledgements

This research was supported by the National Natural Science Foundation of China under Grant 71171131 and 70533030

References

Abbasi, A., and Chen, H. 2008. "CyberGate: a Design Framewrok and System for Text Analysis of Computer-Mediated Communication," *MIS Quarterly* (32:4), pp. 811-837.

Bagga, A., and Baldwin, B. 1998. "Entity-based Cross-document Coreferencing Using the Vector Space Model," In *Proceedings of the 36th Annual Meeting on Association for Computational Linguistics,* Morristown, NJ: ACL, pp. 79–85.

Barcellini, F., Detienne, F., Burkhardt, J., and Sack, W. 2005. "A Study of Online Discussions in an Open-Source Software Community: Reconstructing Thematic Coherence and Argumentation from Quotation Practices," In *Proceedings of the Communities and Technologies Conference* (C&T'05), New York, NY: Springer, pp. 301–320.

Barzilay, R., and Elhadad, M. 1997. "Using Lexical Chains for Text Summarization," In *Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, Morristown, NJ: ACL. pp. 10–17.

Braak, SW., Oostendrop, H., Prakken, H., and Vreeswijk, G.A.W. 2008. "A Critical Review of Argument Visualization Tools: Do Users Become Better Reasoners?" *Workshop Notes of the ECAI-06 Workshop on Computational Models of Natural Argument*, pp. 67-75.

Brill, E. 1995. "Transformation-based Error-drivenLearning and Natural Language Processing: a Case Study in Part-of-speech Tagging," *Computational Linguistics* (21:4), pp. 543-565.

Carbogim, D.V., Robertson, D., and Lee, J. 2000. "Argument-based Applications to Knowledge Engineering," *The Knowledge Engineering Review* (15), pp. 119-149.

Cherny, L. 1995. "The Modal Complexity of Speech Events in a Social MUD," *Electronic Journal of Communication* (5:4).

Chesnevar, C., Mcginnis, J., Modgil, S., Rahwan, I., Reed, C., Simari, G., South, M., Vreeswijk, G., and Willmott, S. 2006. "Towards an Argument Interchange Format," *The Knowledge Engineering Review* (21:4), pp. 293-316.

Chia,R. 2000. "Discourse Analysis as Organizational Analysis," *Organization* (7:3), pp. 513-518

Choi, F.Y.Y. 2000. "Advances in Domain Independent Linear Text Segmentation," In *Proceedings of the Meeting of the North American Chapter of the Association for Computational Linguistics* (ANLP-NAACL-00), San Francisco, CA: Morgan Kaufmann, pp. 26–33.

Clark, H.H., and Brennan, S.E. 1991. "Grounding in Communication," *Perspectives on Socially Shared Cognition*, American Psychological Association, Washington, DC, pp. 127-149

Eklundh, K.S., and Rodriguez, H. 2004. "Coherence and Interactivity in Text-based Group Discussions Around Web Documents," In *Proceedings of the 37th Annual Hawaii International Conference on System Sciences* (HICSS'04). Washington, DC: IEEE Computer Society, pp. 40108.3.

Farnham, S., Chesley H.R., McGhee D.E., Kawal R., and Landau J. 2000. "Structured Online Interactions: Improving the Decision-Making of Small Discussion Groups," in *Proceedings of the 2000 ACM conference on Computer supported cooperative work* (CSCW'00), Philadelphia, PA, pp. 299-308.

Fu, T., Abbasi, A., and Chen, H. 2008. "A Hybrid Approach to Web Forum Interactional Coherence Analysis," *Journal of the American Society for Information Science and Technology* (59:8), pp. 1195-1209

Fuks, H., Pimentel, M., and Lucena, CJP. 2006. "R-U-Typing-2-Me? Evolving a Chat Tool to Increase Understanding in Learning Activities," *Computer-Supported Collaborative Learning* (1:1), pp. 117-142.

Halliday M.A.K. and Hasan R. 1976. *Cohesion in English*, London: Longman.

Herring, S.C. 1999. "Interactional Coherence in CMC," *Journal of Computer-Mediated Communication* (4:4).

Herring, S.C., and Nix, C. 1997. "Is 'Serious Chat' an Oxymoron? Academic vs. Social Uses of Internet Relay Chat," *American Association of Applied Linguistics,* Orlando, FL.

Hiltz, S.R., and Turoff, M. 1985. "Structuring Computer-mediated Communication Systems to Avoid Information Overload," *Communication of the ACM* (28:7), pp. 680-689.

Holmer, T., Lukosch, S., and Kunz, V. 2009. "Diminishing Chat Confusion by Multiple Visualizations," *Journal of Universal Computer Science* (15:16), pp. 3139-3157.

Khan, F.M., Fisher, T.A., Shuler, L., Wu, T., and Pottenger, W.M. 2002. "Mining Chat-room Conversations for Social and Semantic Interactions," (http://www3.lehigh.edu/images/userImages/jgs2/Page_3471/LU-CSE-02-011.pdf).

Kim, J., Li, J., and Kim, T. 2010. "Towards Identifying Unresolved Discussions in Student Online Forums," *The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2010)*, Los Angeles, CA, pp. 84-91.

Kuechler, W.L. 2007. "Business Applications of Unstructured," Communications of the ACM (50:10), pp. 86-93.

Lambiase, J. J. 2010. "Hanging by a Thread: Topic Development and Death in an Electronic Discussion of Breaking News," *Language@Internet* (7), article 9.

Li, J., Zhang, P.Z., and Cao, J.W. 2009. "External Concept Support for Group Support Systems through Web Mining," *Journal of the American Society for Information Science and Technology* (60:5), pp. 1057-1070.

Louwerse, M. M. 2002. "An Analytic and Cognitive Parameterization of Coherence Relations," *Cognitive Linguistics* (12), pp. 291-315.

McDaniel, S.E., Olson, G.M., and Magee, J.C. 1996. "Identifying and Analyzing Multiple Threads in Computer-Mediated and Face-to-Face Conversations," in *Proceedings of the 1996 ACM conference on Computer supported cooperative work* (CSCW'96), Cambridge MA, pp. 39-47.

McGrath, J.E. 1990. "Time Matters in Groups," Intellectual teamwork: Social and technological foundations of cooperative work, pp. 23-61.

Murry, G., and Renals, S. 2007. "Term-Weighting for Summarization of Multi-party Spoken Dialogues," in *Proceeding of the 4th international conference on Machine learning for multimodal interaction*, Popescu-Belis, S.Renals, and H. Bourlard (Eds.), Springer-Verlag Berlin, Heidelberg, pp. 156-167.

Nash, C.M. 2005. "Cohesion and Reference in English Chatroom Discourse," in *Proceedings of the 38th Hawaii International Conference on System Science* (HICSS'05) (pp. 108.3), Washington, DC: IEEE Computer Society.

Newman, P.S. 2002. "Exploring Discussion lists: Steps and directions," In *Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital libraries*, New York, NY: ACM, pp. 126–134.

Paccagnella, L. 1997. "Getting the Seats of Your Pants Dirty: Strategies for Ethnographic Research on Virtual Communities," *Journal of Computer-Mediated Communication* (3:1) (http://jcmc.indiana.edu/vol3/issue1/paccagnella.html).

Paul, S., and Nazareth, D.L. 2010. "Input Information Complexity, Perceived Time Pressure, and Information Processing in Gss-based Work Groups: an Experimental Investigation Using a Decision Schema to Alleviate Information Overload Conditions," *Decision Support Systems* (49:1), pp. 31-40.

Popolov, D. 2000. "Conversation Space: Visualizing Multi-threaded Conversation," in *Proceedings of the Working Conference on Advanced Visual Interfaces* (AVI'00), pp. 246-249.

Raghu T.S., Ramesh R., Chang AM, and Whinston AB. 2001. "Collaborative Decision Making: a Connectionist Paradigm for Dialectical Support," *Information Systems Research* (12:4), pp. 363-383.

Rienks, R., Heylen, D., and Weijden, E. 2005. "Argument Diagramming of Meeting Conversations," (http://hmi.ewi.utwente.nl/~rienks/documenten/ICMI2005.pdf)

Sack, W. 2000. "Conversation Map: an Interface for Very Large-scale Conversations," *Journal of Management Information Systems* (17:3), pp. 73–92.

Salton, G., and McGill, M.J. 1986. "Introduction to Modern Information Retrieval," New York, NY: McGraw-Hill.

Salton, G., and Buckley, C. 1988. "Term-weighting Approaches in Automatic Text Retrieval," *Information Processing & Management* (24:5), pp. 513-523

Samuel, K., 2000. "An Investigation of Dialogue Act Tagging Using Transformation-Based Learning," PhD Thesis, University of Delaware.

Sanders, T., and Maat, H.P. 2006. "Cohesion and Coherence: Linguistic Approaches," *Cohesion in English*. London: Longman, pp. 591-595.

Schegloff, E.A., 1968. "Sequencing in Conversational Openings," *American Anthropologist* (70), pp. 1075-1095.

Schweickert, R. 1993. "A Multinomial Processing Tree Model for Degradation and Redintegration in Immediate Recall", *Memory & Cognition* (21:2), pp. 168-175

Sheridan, T.B., and Ferrell, W.R. 1974. *Man-Machine Systems: Information, Control, and Decision Models of Human Performance*, MIT Press, Cambridge, Mass.

Smith M., Cadiz J.J., and Burkhalter B. 2000. "Conversation Trees and Threaded Chats," in *Proceedings of the 2000 ACM conference on Computer supported cooperative work* (CSCW'00), Philadelphia, PA, pp. 97-105.

Smith, M.A., and Fiore, A.T. 2001. "Visualization Components for Persistent Conversations," in *Proceedings of the SIGCHI conference of Human factors in computing systems*, New York, NY, pp. 136-143.

Soon, W.M., Ng, H.T., and Lim D.C.Y. 2001. "A Machine Learning Approach to Coreference Resolution of Noun Phrases," *Computational Linguistics* (27:4), pp. 521–544.

Spiegel, D. 2001. "Coterie: a Visualization of the Conversational Dynamics Within IRC," MIT Master's Thesis, (http://alumni.media.mit.edu/~spiegel/thesis/Thesis.pdf).

Toulmin, S. 1958. *The Uses of Arguments*, Cambridge University Press, Cambridge, England

Turoff, M., Hiltz, S.R., Bieber, M., Fjermestad, J., and Rana, A. 1999. "Collaborative Discourse Structures in Computer Mediated Group Communications," *Journal of Computer-Mediated Communication* (4:4)

Viegas, F.B., and Donath, J.S. 1999. "Chat Circles," *in Proceeding of the SIGCHI Conference on Human Factors in Computing Systems (CHI'99)*, pp. 9-16.

Wilson, S.M., and Peterson, L.C. 2002. "The Anthropology of Online Communities," *Annual Review of Anthropology* (31), pp. 449-467.

Xiong, R., Smith, M.A., and Drucker, S.M. 1998. "Visualizations of Collaborative Information for End-users," *Technical Report MSRTR-98-52, Microsoft Research.*

Yee, K.P. 2002. “Zest: Discussion Mapping for Mailing Lists,” *In CSCW2002 Conference Supplement,* New York, NY: ACM, pp. 123–126.

Zechner, K. 2002.”Automatic Summarization of Open-Domain Multiparty Dialogues in Diverse Genres,” *Computational Linguistics* (28:4), pp. 447-485.