Towards identifying unresolved discussions in student online forums

Jihie Kim · Jeon-Hyung Kang

Published online: 21 December 2013
© Springer Science+Business Media New York 2013

Abstract Online discussion is a popular form of web-based computer-mediated communication and is a dominant medium for cyber communities in areas of information sharing, customer support and distributed education. Automatic tools for analyzing online discussions are highly desirable for better information management and assistance. For example, a summary of student Q&A discussions or unresolved questions can help the instructor assess student dialogue efficiently, which can lead to better instructor guidance for student learning by discussion.

This paper presents an approach for classifying student discussions according to a set of discourse structures, and identifying discussions with confusion or unanswered questions. Inspired by the existing spoken dialogue analysis approaches, we first define a set of forum “speech acts” (F-SAs) that represent roles that individual messages play in threaded Q&A discussions, such as questions, raising issues, and answers. We then model discourse structures in discussion threads using the F-SAs, such as whether a question was replied to with an answer. Finally, we use such discourse structures in classifying and identifying discussions with unanswered questions or unresolved issues.

We performed an analysis of the discussion thread classifiers and the system showed accuracies from 0.79 to 0.87 on several discussion classification problems. This analysis of human conversation via online discussions provides a basis for development of future information extraction and intelligent assistance techniques for online discussions.

Keywords On-line discussion board · Speech act · Discussion assessment · Discourse · Student Q&A forum

1 Introduction

Discussion boards have become an essential tool for communication in higher education, in part due to their integration into course management systems that are now centrally supported by many colleges and universities. For some courses, discussion boards are mandatory for supporting distance learners who participate in courses ‘virtually’, alongside on-campus peers. Recent studies have pointed to online discussion boards as a promising strategy for promoting collaborative problem solving and discovery-oriented activities [1, 14, 20, 24, 29]. Students often use discussion forums to exchange information, and to seek answers to problems from their instructors and classmates. However, as such courses become more successful and their enrollment increases, the heavier on-line interaction places considerable burdens on instructors and teaching assistants. Thus, the ultimate success of web-based education is constrained by limited instructor time and availability. Our goal is to develop instructional tools that will help instructors assess student discussion dialogue more efficiently. Automatic or semi-automatic tools for analyzing student discussions are highly desirable, to lead to better instructor guidance for student learning by discussion.

Our work focuses on an assessment of Q&A style student discussions. In this paper, we present a novel discussion analysis framework that automatically identifies discussions that have unresolved issues or unanswered questions. Instructors are typically interested in knowing whether the question poster was satisfied by the answer, whether all the
questions in the discussion were answered, etc. The resulting discussion analysis results, such as unresolved discussions, can be reported to instructors for further assistance.

Our technical work is inspired by existing spoken dialogue analysis research. We view individual online discussions as dialogues. In particular, we model individual messages in discussions with respect to Q&A dialogue roles that they play, such as question, issue raising, answer, and acknowledgement. We define a set of Forum Speech Acts (F-SA) by adopting the theory of Speech Acts proposed for spoken dialogue [2, 25] and extending it to capture forum dialogue dynamics. When a message replies to another one, F-SAs define the roles that the former plays against the latter, such as an answer to a question. A discussion or a discussion thread includes all the messages connected by reply-to relations.

While many existing models for spoken dialogue typically capture a linear sequence of turn exchanges between two people, asynchronous online discussions can include contributions (messages) from more than two people and the reply relations among the messages can be nonlinear. As there can be more than one reply to a message, reply chains can form a tree structure where each link represents which message replies to which. Moreover, while a short spoken utterance tends to play a single role, a ‘written’ forum message can be longer and often play multiple roles with respect to a prior message. One message can also respond to multiple messages from more than one person. Our F-SAs capture multiple roles that a message plays in a discussion.

Our work is also inspired by spoken dialogue research that emphasizes use of high-order structural information in analyzing dialogue patterns [6, 10]. In the manner that speech acts are used in some of the spoken dialogue analysis research, we use F-SAs in modeling forum discourse. Furthermore, with F-SAs, we capture various discourse structures that are relevant to identifying unanswered questions or assessing whether the question poster was satisfied by the answer. Such discourse structures include existence of certain F-SAs between messages (e.g. whether there is an answer to a question), locations of F-SAs within the thread (e.g. whether the last message in the thread has a question) and F-SAs of certain authors (e.g. did the first author post another question later). Such discourse structure information can be used as features for classifying the status of the forum dialogue, such as whether there are hanging (unanswered) questions.

In developing discussion thread classifiers, we take a two-step classification process. We first create F-SA classifiers that identify roles of a message toward the one it replies to. The F-SA classifiers use content (n-gram) features, previous message content and relational information between the two messages, such as whether the poster has changed. In addition, thread-level location information, such as whether it is an initial message, is used as features. That is, in identifying roles of a message, F-SA classifiers make use of relational and locational discourse information since the roles can depend on information about the neighboring messages, author relations and the message position in the thread. Given a discussion, when F-SAs are identified for each message, discourse structures of the discussion can be further analyzed using the F-SAs. For example, the system can check whether a reply to the question message contains certain F-SAs (an Answer) or the last message contains certain F-SAs, as described above. Such discourse structures capture relational information among F-SAs and their locations within the discussion thread. Finally, the discussion can be classified using the F-SA-based discourse structure information. For example, when there is an answer to the question and the discussion ends without a further question, the system may predict that the question was answered.

In evaluating the resulting system, we use a set of discussion assessment questions that are relevant to identifying unanswered questions or checking whether a particular message answers the student question. We compared the performance of the discussion classifiers that use the abovementioned discourse structure information against a discussion classifier that uses existence of F-SAs only. Since the F-SA only classifiers check existence of different F-SAs in discussions, they still capture useful discourse information for assessing the status of the dialogue. Additional discourse structure features represent relational or locational information about the F-SAs that appear in the discussion. We used discussion data from undergraduate operating system courses, and compared assessment results from the two sets of classifiers for each discussion assessment question. The performance was measured with human annotations. The preliminary results indicate that the only F-SA classifier can identify threads with unanswered questions reasonably. The discussion classifiers that use additional information from discourse structure provide slightly improved results in identifying threads for assessment questions that check specific structural information, such as whether the first answer satisfies the questioner.

In the next section, we introduce a set of F-SA categories that we developed based on an analysis of student discussions. We then provide details of F-SA classifiers including the steps for extracting features that are appropriate for classifying student online discussions. The following section presents discourse structure features that we developed using F-SAs and how they are used in creating discussion classifiers. We show how the current classifiers perform in terms of precision, recall and accuracy. Finally we present a summary and future work.

The contributions of this paper are twofold. First, we present a novel forum speech act classification approach that defines roles that individual messages play with respect to
Towards identifying unresolved discussions in student online forums

the message it replies to. Second, we provide a set of features for forum discussions that capture rich information about discourse structures in Q&A forums, and show how such information can help student discussion classification.

2 Modeling student online discussions

Our study takes place in the context of an undergraduate course discussion board that is an integral component of an Operating Systems course in the Computer Science Department at the University of Southern California. We obtain our data from an existing online discussion board that hosts student technical discussions. Total 240 discussion threads with 904 messages from two semesters’ discussions were used for this study.

Unlike prototypical collaborative argumentation where a limited number of members take part in the conversation with a strong focus on solving specific problems, student online discussions have much looser conversational structure, possibly involving multiple discussants. Figure 1 shows an example discussion thread that is relatively technical and formal. The first message contains questions on a project assignment. The following message provides a suggestion. The next one expresses confusion, as the answer did not fully resolve the issue. The fourth message contains elaborated answer to the question. There can be more than several participants in discussions and a message can be replied to by more than one message from more than one person. A message can play multiple roles. For example, the same message can include both an answer and a related question. That is, a message can play multiple roles toward the message it replies to. A discussion can include multiple reply chains to form a tree structure as it diverges into several different aspects of the given problem. For example, in Fig. 1, there are two reply chains as there are two different replies to message M3.

Student discussions are very informal and noisy with respect to grammar, syntax and punctuation. There is a lot of variance in the way that students present similar information. Messages about programming assignments include various forms of references to programming code. The raw data include humorous messages and personal announcements as well as technical questions and answers. The average number of messages per discussion thread in our undergraduate course is 3.9, and many discussion threads contain only two or three messages. Discussions often start with a question from a student on a project or an assignment. In some cases, the discussion ends with an answer that follows the question. In some other cases, the original question poster may raise additional issues or ask questions about the answer. The discussion can continue with a following answer from another student as in Fig. 1. However, sometimes the discussion ends with hanging issues or questions without an answer.

2.1 Forum speech acts: identifying roles that a message plays

For conversation analysis, we adopted the theory of Speech Acts (SAs) to capture relations between messages [2, 25]. We define Forum Speech Act (F-SA) for defining roles that a message plays against another message it replies to, such as question, answer, acknowledgement and objection. A message may play multiple different roles. A message can include a question for a particular problem, or it could contain an answer or suggestion with respect to a previous question in the thread. Since F-SAs are useful in understanding contributions made by students in discussions, and are natural indicators for unanswered questions or unresolved issues, we use F-SAs and discourse structures modeled with F-SAs as features for classifying discussion threads. Table 1 lists the F-SA categories that are relevant to the problem of identifying discussion threads with unanswered question, unresolved issues or checking whether the answer satisfy the questioner.

The message might contain a question about a particular problem (QUES) or report a misunderstanding, unclear concepts or issues in solving a problem (ISSUE). It might propose an answer or suggestion with respect to a previous question in the thread (ANS-SUGG). Finally, a message might acknowledge the previous message with support (Pos-Ack) or show disagreement or objection (Neg-Ack). A message can be labeled with multiple F-SAs with respect to the
Table 1  Forum speech act categories and kappa values

| F-SA category | Description | Example cue words from annotations | %   | kappa |
|---------------|-------------|------------------------------------|-----|-------|
| QUES          | A question about a problem, including question about a previous message | “how” “what” “can we” “are”/“is” “why” “just/were/was wondering” “I/we have a question” “my question” | 40.9 | 0.94 |
| ANS-SUGG      | A simple or complex answer to a previous question. Suggestion or advice | “perhaps” “how about” “you might” “you probably” “maybe” “try” “I think” “I am/was thinking” “I’m guessing” “my guess” “it should” “it seems” “look at” “check” | 54.6 | 0.72 |
| ISSUE         | Report misunderstanding, unclear concepts or issues in solving problems | “I am still confused” “I was confused” “doesn’t make sense” “I’m not sure” “I’ve no idea” “Not sure” “We ran into” “I am getting fault” “I discovered an error” “problem I am facing” “I am not able to find anything” “I couldn’t find any specific” “I still have no clue” | 13.3 | 0.88 |
| Pos-Ack       | An acknowledgement, compliment or support in response to a previous message | “good job” “you got it” “good plan” “good/nice/correct answer” “correct” “thank you/thanks” “I got it:)” “ok/okay” “I agree” “its fine with me” “I’m okay with...” “good job” “good/nice/correct answer” “correct” | 7.7  | 0.87 |
| Neg-Ack       | A correction or objection (or complaint) to/on a previous message | “WRONG” “but not correct” “the above is incorrect.” “Not going to work” “not true” “is actually wrong” “That is true, but” “sure but” “yes, but” “right but” “certainly, but” “I understand but” | 2.3  | 0.85 |

Fig. 2  Example patterns in Q&A discussion threads

message it replies to. This allows us to capture various relations among messages. Figure 1 shows such F-SA relations between message pairs. For example, M2 provides an answer toward M1. Table 1 describes the F-SA categories and their distribution in our corpus. Question and Ans-Sugg categories cover many F-SAs annotated. Both Pos-Ack and Neg-Ack categories are rarer compared to other categories. Due to lack of negative acknowledgements (<3 %), we exclude that category in our analysis.

2.2 Analyzing discussions with F-SAs

Figure 2 shows typical patterns of interactions in our corpus. Many threads follow a pattern (a) where the first message includes a question and the subsequent message provides an answer. In (b), after an answer, the student presents an additional question or misunderstanding (ISSUE), which is followed by another answer. Often students provide positive acknowledgement when an answer is satisfying. Pattern (c) covers cases for when some of the questions or issues are not answered. Pattern (d) shows when the question is replied to by another question to elaborate the details of the problem and then followed by the actual answer for the first question. When someone provides an answer, it is often acknowledged by the initial questioner or other answer providers as a confirmation, (pattern (e)) or followed by other answers as other possible solutions or additional description (pattern (f)).
Towards identifying unresolved discussions in student online forums

Inspired by the assessment needs of the instructors, we are interested in answering the following assessment questions by presenting useful discourse structure models.

(Q1) Were all the questions answered? (Y/N)
(Q2) Were there any issues or confusion? (Y/N)
(Q3) Were those issues or the confusion resolved? (Y/N)
(Q4) Did the first answer satisfy the first questioner? (Y/N)

Note that “all the questions are answered” is not the same as “some questions are answered”. As there can be additional questions to the initial answer, we want to understand whether the later questions were answered as well. On the other hand, Q4 analyzes more specifics on the discourse such as whether the first answer was satisfactory.

F-SAs are useful for distinguishing different interaction patterns or discourse structures, including threads with unanswered questions or threads with unresolved issues. The following heuristics can be used in answering these assessment questions.

(Q1) Were all the questions answered?
– Whether there was a question or an issue in the discussion.
– For each question, whether there was an answer that replies to the question and the reply chain does not end with a question or an issue without an answer.

(Q2) Were there any issues or confusion? (Y/N)
– Whether there was an issue in the discussion.

(Q3) Were those issues or the confusion resolved? (Y/N)
– Whether there was an issue in the discussion.
– For each issue, whether there was an answer that replies to each issue, and the reply chain does not end with a question or an issue without an answer.

(Q4) Did the first answer satisfy the first questioner? (Y/N)
– Whether there was an answer for the first question and the first question author did not raise further issues or questions after the answer.

Answering the above questions require information on

(a) Existence of certain F-SAs in a discussion thread:
   Example: Whether an issue exists in a thread.
(b) Relative locations of F-SAs with respect to reply-to relations:
   Example: Whether the message that replies to the first message contains an answer.
(c) Existence of certain F-SAs in the message that ends the reply chain:
   Example: Whether the last message has an issue or a question.
(d) Relative position of the author in the discussion thread and his/her F-SAs:
   Example: F-SAs of the first question author’s messages

We create feature spaces that cover these discourse structures and author information.

3 Developing F-SA classifiers for student Q&A discussions

In this section, we first explain how raw discussion data is processed and describe the features used for F-SA classification. We then present the current F-SA classifier accuracy. 240 discussion threads (180 for training and 60 for test) with 904 messages (634 for training and 270 for test) from two semesters’ discussions were used for this study.

3.1 F-SA annotation process

During annotation of the corpus, the annotators analyzed whether abovementioned F-SA categories are contained in a message. An F-SA relates a pair of messages where one replies to the other. More than one F-SAs can be noted within a message, as described above. Annotators marked the cues that are relevant to a particular F-SA as well as the F-SA itself. Such information provides hints on the kinds of features that are useful. We also interviewed the annotators to capture additional cues or indicators that they used during the annotation. We iterated with several different annotation approaches until we reached enough agreement among the annotators on a new dataset that was not seen by the annotators before. Table 1 shows the kappa values from two annotators. The annotators had enough (>0.7) agreement on the dataset that they reviewed.
3.2 Discussion data pre-processing: handling incoherent and noisy discussion data

Discussion data from undergraduate students are highly incoherent and noisy. The raw data includes humor and personal announcements as well as technical text. Student messages are very informal and there are high variances in the way they present similar information. A lot of messages on programming assignments include programming code. Besides typical data preprocessing steps, such as stemming and filtering which are taken by most NLP systems, our system performs additional steps to reduce noise and variance.

We first remove the text from previous messages that is automatically inserted by the discussion board system when the user clicks a “Reply to” button. Each line of such auto-inserted text is marked with a right angle bracket (>). We also apply a simple stemming algorithm that removes “s” and “es” for plurals. Short-handed forms with apostrophes are also converted to their original forms. E.g. “I’m” is converted to “I am”. For discussions on programming assignments, messages can include programming code fragments. Each section of programming code or code fragment is replaced with a single token called “code”. Similar substitution patterns were used for a number of categories like file-type extensions (“.html”, “.c”, “.c++”, “.doc”), URL links and others. Students also tend to use informal words (e.g. “ya”, “yeah”, “yup”). We substitute some of such words with one form (“yes”). We generated 20 popular word form categories from a randomly selected training set to combine raw terms and reduce variances; “I” and “we” were replaced by CAT_SUB_I_WE; “is”, “was”, “are”, and “were” were replaced by CAT_BE; interrogative words such as “what” and “where” were replaced by CAT_WH and “fault”, “problem”, and “error” were categorized by CAT_ISSUE. However, we do not merge them arbitrarily; for example, “you” and “I” belong to different categories since “You can” may be a cue for ANS but “I can” may not. Appendix (Table 9) lists some of the representative categories that we use in the text processing. After these text-preprocessing steps, we could decrease the size of the unigram vector size from 5257 to 457, the bigram vector size from 3025 to 1218 and the trigram vector size from 1041 to 1031.

We also apply a simple sentence divider with simple cues (punctuation and white spaces such as a new line) in order to capture the locations of the features in the message, such as cue words in the beginning sentences vs. cues in the other sentences, or cue words in the first half of the message vs. cues in the second half of the message.

3.3 Feature selection

We used six different types of features based on the input from the annotators. In addition to typical message-level features including n-grams and information about previous message that are often used in dialogue act classifications [4, 23], we used additional thread-level features that capture the message location and poster orderings.

F1: Cue phrases and their positions: In addition to F-SAs (e.g. QUES), the human annotators marked the parts within the message (cue phrases or sentences), which helped them identify the F-SAs in the messages. In order to overcome data sparseness, we generate features from the marked phrases. From each phrase, we extract all the unigrams, bigrams, trigrams (sequences of 1/2/3 words) and add them to the feature set. Positions of the cues are included since in longer messages the cues in the beginning sentences and the ones in the end sentences can indicate different F-SAs. For example, THANK in the beginning indicates a positive answer but THANK in the latter part of the message usually means politeness (thank in advance). For each cue phrase, we include three possible position combinations: the first sentence vs. other sentences, the first half part of message vs. latter part of message, and any position in the message.

F2: Cue phrases in the previous message: The F-SAs in the previous message are important indicators for the current message’s F-SAs. Intuitively, if the previous message contains QUESs and ISSUEs, the current message will more likely include ANSs rather than QUESs.

F3: Position of the current author (e.g. whether this message is written by the first author, the second author, or the last author): In Q&A discussions, the poster position information, according to the poster participation ordering, can be useful. The first poster, the second poster, and the last poster tend to play certain roles for the discussion, and their roles may stay the same within the discussion in some threads. For example, the initial posters’ messages tend to be a question message and their other messages in the same thread are likely question messages as well. Likewise, the second author tends to provide an answer to the initial question and his/her other messages in the same thread may provide information (answers) to the help seeker rather than posing new or related questions.

F4: Position of the previous author (e.g. whether the previous message is written by the first author, the second author, or the last author): The poster position of the previous message can be a good indicator of F-SAs. For example, if the current message replies to the message written by the first author, it is more likely to include ANSs rather than QUESs.

F5: Position of the current message (e.g. Is this the first message, the second message, or the last message?): Similarly, message position information can be important in identifying F-SAs. Most of the first messages include
Table 2  Example top N-grams selected with information gain scores

| QUES      | ANS-SUGG         | ISSUE         | Pos-Ack        |
|-----------|------------------|---------------|----------------|
| [?]_ALL   | 1stAuthor 0.20   | [ISSUE]_ALL  | [thank]_ALL    |
| 1stMessage| 1stMessage 0.18  | [IWE]_ALL    | [thank+YOU]    |
| 1stAuthor | PRE_[?]_ALL 0.31 | [IWE+BE]_ALL | [same]_ALL     |
| 2ndAuthor | 2ndAuthor 0.15   | 1stAuthor    | 1stMessage     |
| replyTo1stAuthor | 1stAuthor 0.14 | [get]_ALL    | [same+ISSUE]_ALL |
| PRE_1stAuthor | 1stMessage 0.06 | 1stMessage   | [BE+correct]_ALL |
| replyTo1stMessage | 1stAuthor 0.14 | [get]_ALL    | [BE+correct]_ ALL |
| PRE_1stMessage | 1stMessage 0.10 | [not+sure]_ ALL | [YOU+for]_NOTFIRST |
| PRE_[IWE] _ALL 0.13 | PRE1stMessage 0.10 | PRE_[?]_ALL | [BE+get+same]_ALL |
| 2ndMessage | 2ndMessage 0.08  | PRE1stMessage| PRE_ [BE]_NOTFIRST |
| LastMessage | 2ndMessage 0.08  | PRE1stAuthor 0.04 | PRE_ [BE]_ NOTFIRST |
| [WH]_ALL 0.10 | PRE_[WH]_ALL 0.05 | [BE+not+sure]_ ALL | PRE_ [wrong] _BOTTOM 0.02 |
| [IWE]_ALL 0.09 | PRE_[BE]_ALL 0.05 | [not]_ALL 0.03 | [YOU+ BE+correct]_ ALL 0.02 |
| PRE_[BE]_ALL 0.09 | PRE_[doc]_ALL 0.05 | [sure]_ALL 0.03 | PRE_[correct+MEUS]_NOTFIRST 0.02 |
| [do+IWE]_ALL 0.07 | [IWE]_ALL 0.05 | [give]_NOTFIRST 0.03 | PRE_[MEUS +if]_NOTFIRST 0.02 |
| PRE_[WH]_ALL 0.07 | LastMessage 0.04 | 2ndMessage 0.03 | [IWE+did]_BOTTOM 0.02 |
| [YOU]_ALL 0.06 | LastAuthor 0.04 | replyTo1stMessage 0.03 | PRE_[MEUS +if+ IWE]_ NOTFIRST 0.02 |
| [can+IWE]_ALL 0.06 | [YOU+can]_ALL 0.04 | PRE1stMessage 0.03 | PRE_[IWE+ BE+wrong]_BOTTOM 0.02 |
| LastAuthor 0.05 | [BE+not]_ALL 0.03 | PRE_[correct IWE+if]_ NOTFIRST 0.02 |

QUESs that initialize the discussion and the second messages tend to provide ANSs to resolve the given problem.

F6: Position of the previous message: Likewise, the position of the previous message can be a good indicator for the F-SAs of the current message. For example, if the previous message was the first message, the current message will likely include ANSs.

Note that some of these features capture location and relational information of the messages and message authors within the discourse as well as content features. For example, we capture either orderings or positions of the message or the poster. We also incorporate information about previous messages.

We use the Information Gain theory [31] for pruning the feature space and selecting features. The information gain value for a particular feature gives a measure of the information gained in classification prediction, i.e., how much information about the target value we can get based on the absence or the presence of the feature. First, we compute the information gain scores for the abovementioned features extracted from the training data. Subsequently, all the features are sorted using their information gain scores. We use the top 200 features for each classifier. Some of the top N gram features for QUES, ANS-SUGG, ISSUE, and Pos-ACK are shown in Table 2.

Table 3  Statistics for each forum speech act category

| F-SA category | Training set | Test set |
|---------------|-------------|---------|
|               | # of msgs   | Percentage | # of msgs | Percentage |
| QUES          | 276/674     | 40.94%    | 92/225   | 40.89% |
| ANS           | 368/674     | 54.59%    | 123/225  | 54.67% |
| ISSUE         | 66/674      | 9.79%     | 22/225   | 9.78%  |
| Pos-Ack       | 52/674      | 7.71%     | 17/225   | 7.56%  |

3.4 F-SA classifiers

We applied LibSVM [5] and Weka algorithms [12] in creating binary classifiers for each F-SA category. Each F-SA classifier checks whether certain F-SA exists for the given message pair.

Table 3 shows the distribution statistics of each F-SA category among the whole training and test corpus. Since a message may have more than one F-SA, the percentage sum of all the F-SAs does not equal 1. Question and Ans-Sugg categories cover many cases.

For LibSVM, we did 10-fold cross validation in the training. RBF (Radial Basis Function) is used as the kernel function as it is one of the most popular ones, and the value of the RBF kernel can be interpreted as a similarity measure since it ranges from zero to one. We performed a grid search
to get the best parameter (C and gamma) in training and applied them to the test corpus.

Table 4 shows the current classification accuracies with SVM, J48 and Naïve Bayes. The main reason that ISSUE and Pos-Ack show lower scores is that they have a relatively small number of examples or true cases (see statistics in Table 3). Even though the current F scores for those categories are lower than the scores for QUES or ANS, their accuracies are relatively high (90.6 and 94.6 for ISSUE and Pos-Ack respectively). We plan to add more examples as we collect more discussion annotations.

4 Using forum speech act (F-SA) classification for identifying unanswered or unresolved questions

This section describes our models for representing discourse in Q&A discussions and how we use them in answering the above assessment questions (Q1–Q4).

We identify messages using their positions. We also identify the poster (or author) of each message. For example, in Fig. 3, the first author u1 posted messages M1 and M4. The discussion has two reply-to chains, and M3 and M6 represent the messages that end the chains. We call such messages as thread-ending messages. Table 5 lists notations that we use for representing discourse features.

Table 4 F-SA classification results

| F-SA category | SVM Precision | SVM Recall | SVM F score | J48 F score | Naïve Bayes F score |
|---------------|--------------|------------|-------------|-------------|-------------------|
| QUES          | 88.2         | 89.1       | 88.6 (90.6) | 88.8        | 82.6              |
| ANS           | 89.2         | 81.1       | 85.0 (84.4) | 83.9        | 84.7              |
| ISSUE         | 71.4         | 50.0       | 58.8 (90.6) | 64.3        | 0.0               |
| Pos-Ack       | 85.7         | 35.3       | 50.0 (94.6) | 59.3        | 0.0               |

Table 5 Representation for messages, users and message relations

| F-SA^u_k | Boolean function that returns true when there exists a F-SA posted by u in the message set k |
|----------|---------------------------------------------------------------------------------------------|
| F-SA ∈ {Q, A, I, P} (QUES, ANS-SUGG, ISSUE, and Pos-Ack). |
| k: A set of message sequence Ids |
| k ⊂ {1, . . . , total number of messages in the thread} |
| (e.g. Q^k: whether there was any QUES in k) |
| u: A message author sequence Id |
| u ∈ {1, . . . , total number of message authors in the thread} |
| (e.g. Q^a: whether there was a QUES in any messages from a) |

reply-to(k) A set of message sequence Ids that reply to message k.

last-to(k) A set of thread ending messages in the reply-to-chains that contain message k.

Idx(i,Q) Function that returns message Id of i-th QUES message. |
| (e.g. Idx(1,Q): message Id of the initial question message, A^_reply-to(Idx(1,Q)): whether there was any answer that replies to the initial question message) |
Table 6 Discourse feature categories

| DF1  | F-SA*: Whether there exists a given F-SA (Question, Answer, Issue, or Pos-Ack) in any messages in the discussion. |
| DF2  | F-SA reply-to(k): Whether there was a given F-SA in the messages that reply to k. |
| DF3  | F-SA last-to(k): Whether there was F-SA in any thread-ending messages that reply to k. |
| DF4  | F-SA reply-to(1): Whether there was a F-SA in any messages replying to the first message. |
| DF5  | F-SA reply-to(ldx(F-SA2)): For any F-SA2 in the thread, whether there was a F-SA in any messages replying to the F-SA2. |
| DF6  | F-SA last-to(ldx(1,F-SA2)): Whether there was a F-SA in any thread-ending messages that reply to the first F-SA2. |
| DF7  | F-SA i:j: Whether there were F-SA in any of j-th author’s messages (excluding i-th message) |

Table 7 Discourse feature examples for answering Q1–Q4

Q1: DF1. e.g. Whether there were [Question] in any messages in the thread
   DF5. e.g. Whether there were [Answer] in any messages replying to [Question]
   DF6. e.g. Whether there were [Question, Issue] in any thread ending messages that reply to [Question]

Q2: DF1. e.g. Whether there were [Issue] in any messages in the thread
   DF5. e.g. Whether there were [F-SA: Question, Issue] in any messages replying to [Issue]
   DF6. e.g. Whether there were [F-SA: Question, Issue] in any thread ending messages that reply to [F-SA: Issue]

Q3: DF1. e.g. Whether there were [Issue] in the first message in the thread
   DF4. e.g. Whether there were [Answer] replying to the first [Question]
   DF5. e.g. Whether there were [Question, Issue] in any messages replying to the first [Answer] message
   DF6. e.g. Whether there were [Question, Issue] in any thread ending messages that reply to the first [Answer] message
   DF7. e.g. Whether there were [Question, Issue] in the initial questioner’s other messages (except for the initial message)

Q4: No, the first answer raised an issue.
   • DF1: There was a question in the thread.
   • DF4: There was an answer replying to the question.
   • DF5: There was an issue that replies to the answer.

Before we start evaluating the system performance in answering Q1–Q4, we compute the kappa values for the questions. Two people compared their annotations on randomly selected 20 threads. The kappa value for each question is the following:

Q1: Yes, there is a question and the question has an answer and the reply chain does not end with a question or an issue without an answer.
   • DF1: There was a question in the thread.
   • DF5: There was an answer replying to the question.
   • DF6: There was no question/issue that ends the thread.

Q2: Yes, there is an issue in the discussion.
   • DF1: There was an issue in the thread.

Q3: Yes, there is an issue that is resolved by an answer and the reply chain does not end with a question or an issue without an answer.
   • DF1: There was an issue in the thread.
   • DF5: There was an answer replying to the issue.
   • DF6: There was no question/issue that ends the thread.

Q4: No, the first answer raised an issue.
   • DF1: There was a question in the thread.
   • DF4: There was an answer replying to the question.
   • DF5: There was an issue that replies to the answer.

We experiment Q1–Q4 using DF1 only and with additional discourse features. DF1’s check the existence of different F-SAs in the discussions to capture useful discourse information. Additional discourse structure features (DF2–DF7) can represent relational or locational information about F-SAs that appear in the discussion. Table 8(a) and (b) show the classification results for each respectively.

When we compare the two (DF1 only and with DF1–DF7) with human annotated F-SA labels (shaded parts), we can see how discourse structure features affect the results. In case of Q1, including additional discourse features does not improve F scores, and we only have small improvement in accuracy. However, additional discourse features improve both the F score and the accuracy for Q3 and Q4’s, as answering such questions requires more discourse structure information. The accuracy is computed as the sum of diagonal in confusion matrix/total number of data point.

When an end-to-end classification with both the automated F-SA classifiers and discussion thread classifiers (with DF1 only or with DF1–DF7) in a pipeline is used, the system shows precisions (recalls) of 85.7% (89.3%), 83.3% (38.4%) and 89.3% (95.5%) for Q1, Q3, and Q4

For the example discussion shown in Fig. 1, the answer for each question is the following:

**Q1**: Yes, there is a question and the question has an answer and the reply chain does not end with a question or an issue without an answer.
- **DF1**: There was a question in the thread.
- **DF5**: There was an answer replying to the question.
- **DF6**: There was no question/issue that ends the thread.

**Q2**: Yes, there is an issue in the discussion.
- **DF1**: There was an issue in the thread.

**Q3**: Yes, there is an issue that is resolved by an answer and the reply chain does not end with a question or an issue without an answer.
- **DF1**: There was an issue in the thread.
- **DF5**: There was an answer replying to the issue.
- **DF6**: There was no question/issue that ends the thread.

**Q4**: No, the first answer raised an issue.
- **DF1**: There was a question in the thread.
- **DF4**: There was an answer replying to the question.
- **DF5**: There was an issue that replies to the answer.
Table 8  Thread classification results

| Q type | With human annotated F-SAs | With automatically classified F-SAs |
|--------|---------------------------|-----------------------------------|
|        | Precision  | Recall | F Score | Accuracy | Precision  | Recall | F Score | Accuracy |
| (a) Classification results with simple features DF1 only |             |        |         |          |             |        |         |          |
| Q1     | 83.9       | 100.0  | 91.3    | 84.5     | 84.9       | 95.7   | 90.0    | 82.7     |
| Q2     | 94.8       | 90.0   | 92.3    | 94.8     | 85.7       | 60.0   | 70.5    | 92.7     |
| Q3     | 70.6       | 85.7   | 77.4    | 87.9     | 58.3       | 50.0   | 53.8    | 79.3     |
| Q4     | 75.8       | 100.0  | 86.2    | 75.9     | 75.9       | 100.0  | 86.3    | 75.9     |
| (b) Classification results with additional discourse structure features |             |        |         |          |             |        |         |          |
| Q1     | 93.3       | 89.3   | 91.3    | 86.2     | 85.7       | 89.3   | 87.5    | 79.3     |
| Q2     | N/A        | N/A    | N/A     | N/A      | N/A        | N/A    | N/A     | N/A      |
| Q3     | 83.3       | 76.9   | 80.0    | 91.3     | 83.3       | 38.4   | 52.6    | 84.4     |
| Q4     | 89.1       | 93.2   | 91.1    | 86.2     | 89.3       | 95.5   | 92.3    | 87.9     |

respectively. Q2 is not included in Table 8(b) since it requires existence of an F-SA (Issue) only. There are lower recall values in Q3 mainly because of the low accuracy of the ISSUE classifier. To answer Q3, we first need to know whether current thread has ISSUE or not. As we have wrong labels for some ISSUE messages, it is hard to get good accuracy for answering Q3. We expect that the accuracy can improve if there are more examples of ISSUE messages.

5 Related work

Rhetorical Structure Theory [16] based discourse processing has attracted much attention with successful applications in sentence compression and summarization. Most of the current work on discourse processing focuses on sentence-level text organization [27] or the intermediate step [28]. Analyzing and utilizing discourse information at a higher level, e.g., at the paragraph level, still remains a challenge to the natural language community. In our work, we utilize the discourse information at both message and discussion-thread level.

There has been prior work on dialogue act analysis and associated surface cue words [11, 13, 22]. There have also been Dialogue Acts modeling approaches for automatic tagging and recognition of conversational speech [30] and related work in corpus linguistics where machine learning techniques have been used to find conversational patterns in spoken transcripts of dialogue corpus [26]. Although spoken dialogue is different from message-based conversation in online discussion boards, they are closely related to our thread analysis work, and we plan to investigate potential use of conversation patterns in spoken dialogue in threaded discussions.

Carvalho and Cohen present a dependency-network based collective classification method to classify email speech acts [4]. However, estimated speech act labeling between messages is not sufficient for assessing contributor roles or identifying discussions that need help. We included other features like message and participant positional information. Also our corpus consists of less informal student discussions rather than messages among project participants, which tend to be more technically coherent.

Requests and commitments of email exchange are analyzed in [15]. As in their analysis, we have a higher kappa value for questions than answers, and some sources of ambiguity in human annotations such as different forms of answers also appear in our data. However, student discussions tend to focus on problem solving rather than task request and commitment as in project management applications, and their data show different types of ambiguity due to different nature of participant interests.

There has also been work on non-traditional, qualitative assessment of instructional discourse [3, 9, 19]. The assessment results can be used in finding features for student thinking skills or level of understanding. Although the existing work has not been fully used for discussion thread analysis, we are investigating opportunities for using such models of users to develop additional discourse analysis capabilities.

High-order structural information has been emphasized in spoken dialogue analysis [10]. Some of the dialogue analysis work exploits high-level features [6]. We plan to explore more opportunities to exploit dialogue structure information or high-level features.

6 Summary and future work

We have presented an approach for automatically classifying student discussions to identify discussions that have unanswered questions and need instructor attention. We first define a set of forum speech acts (F-SAs) that represent the roles that individual messages play, such as question, issue raising and answers, in student Q&A discussions. Using the F-SAs, we capture relational and locational information that
are relevant to identifying unanswered or hanging questions. We use such discourse structure features in classifying discussion threads and answering assessment questions that can help instructors. The current results indicate that discourse structure features may help classification tasks that require specific relational or locational information about F-SAs and their authors. We plan to perform more analysis on different types of thread classification problems and explore use of similar discourse information in solving them. We also plan to explore approaches for optimizing machine classifiers [7] to improve accuracies.

We found that automatic classification of undergraduate student discussions is very challenging due to incoherence and noise in the data. Messages that contain long sentences, informal statements with uncommon words, and answers in form of question, are especially difficult to classify. We plan to analyze more annotated data to improve performance of the F-SA classifiers.

A deeper assessment of online discussions requires incorporating other information such as discussion topics [8]. For example, classification of discussion topics can be used in identifying topics that participants have more confusion about. Furthermore, such information can also be used in profiling participants such as identifying mentors or help seekers on a particular topic. We are investigating extensions in order to generate more useful instructional tools.

Past approaches to mining information from discussion board text focused mainly on finding answers to questions [8, 17]. Most of these techniques simply consider discussion data as text corpus. However, there are increasing needs for modeling discussion activities more explicitly. A discussion thread consists of a set of messages arranged in chronological order where people may express their ideas, elaborate arguments and answer others’ questions; many of these features in threaded discussions are unexplored by traditional IR techniques. Instructors want to review student discussions in order to understand the kinds of contributions that are made by students and whether they need any assistance or guidance [21]. We may need to identify undesirable distractions, including contributions that are unrelated to the main focus. To support such an assessment, we must be able to track the topics of discussion and determine if the contributions are focused and productive.

We plan to investigate other Q&A forums and investigate opportunities for applying similar approaches. We also would like to relate our work to other educational data mining results such as clustering students based on their performance [18].

Acknowledgements This work is supported by the National Science Foundation, CCLI-Phase 2 grant (Award No. 0618859). We thank Taehwan Kim for his help in data preparation.

Appendix

Table 9 Categories for forum text processing

| Category name | Example                      |
|---------------|------------------------------|
| NUMBER        | 1, 2, 3, 4, 0.1, 0.045       |
| WH            | why, when, where, who, what, which, whose, how |
| IWE           | I, we                        |
| YOU           | you                          |
| HESHESTHEY    | He, she, they                |
| MEUS          | Me, us                       |
| YOU           | You                          |
| HIMHERTHEM    | Him, her, them               |
| MYOUR         | My, our                      |
| YOUR          | Your                         |
| HISHERTHEIR   | His, her, theirs             |
| MINEOURS      | mine, ours                   |
| YOURS         | Yours                        |
| HISHERTHEIRS  | His, hers, theirs            |
| YES           | Ya, yeah, yea, yep, yup, yes, yaah |
| BE            | Is, was, are, were, am, be, being, been |
| ISSUE         | error, fault, problem, errors |

References

1. Ambady N et al (2001) Stereotype susceptibility in children: effects of identity activation on quantitative performance. Psychol Sci 12:385–390
2. Austin J (1962) How to do things with words. Harvard Univ. Press, Cambridge
3. Boyer K, Phillips R, Wallis M, Vouk M, Lester J (2008) Learner characteristics and feedback in tutorial dialogue. In: ACL workshop on innovative use of NLP for building educational applications
4. Carvalho VR, Cohen WW (2005) On the collective classification of email speech acts. In: Proceedings of SIGIR, pp 345–352
5. Chang C-C, Lin C-J (2001) LIBSVM: a library for support vector machines
6. Di Eugenio BD, Xie Z, Serafin R (2010) Dialogue act classification, instance-based learning, and higher order dialogue structure. In: Dialogue & discourse, vol 1
7. Diosoan D, Rogozan A, Pecuchet J (2012) Improving classification performance of support vector machine by genetically optimising kernel shape and hyper-parameters. Appl Intell 36(2):280–294
8. Feng D, Kim J, Shaw E, Hovy E (2006) Towards modeling threaded discussions through ontology-based analysis. In: Proceedings of national conference on artificial intelligence
9. Graesser AC, Olney A, Ventura M, Jackson GT (2005) AutoTutor’s coverage of expectations during tutorial dialogue. In: Proceedings of the FLAIRS conference, pp 518–523
10. Grosz B, Sidner C (1986) Attention, intentions, and the structure of discourse. Comput Linguist 12(3):175–204
11. Hirschberg J, Litman D (1993) Empirical studies on the disambiguation of Cue phrases. Comput Linguist 19(3). http://dl.acm.org/citation.cfm?id=972490
12. Holmes G, Donkin A, Witten IH (1994) WEKA: a machine learning workbench. Tech Report 94/9, Dept. Computer Science, Waikato University, New Zealand
13. Kim J, Chen G, Feng D, Shaw E, Hovy E (2006) Mining and assessing discussions on the web through speech act analysis. In: Proceedings of the ISWC’06 workshop on web content mining with human language technologies
14. Krupnick CG (1985) Women and men in the classroom: inequity and its remedies. In: On Teaching and Learning, vol I. http://ddi.cs.uni-potsdam.de/HyFISCH/Informieren/WomenMenClassroomKrupnick.htm
15. Lampert A, Dale R, Paris C (2008) The nature of requests and commitments in email messages. In: AAAI workshop on enhanced messaging
16. Mann WC, Thompson SA (1988) Rhetorical structure theory: towards a functional theory of text organization text. Interdiscip J Study Text 8(3):243–281
17. Marom Y, Zukerman I (2005) Towards a framework for collating help-desk responses from multiple documents. In: KRAQ’05: IJCAI workshop on knowledge and reasoning for answering questions
18. Marquez-Vera C, Cano A, Romero C, Ventura S (2013) Predicting student failure at school using genetic programming and different data mining approaches with high dimensional and imbalanced data. Appl Intell 38(3):315–330
19. McLaren B, Scheuer O, De Laat M, Hever R, de Groot R, Rose CP (2007) Using machine learning techniques to analyze and support mediation of student e—discussions. In: Proceedings of AI in education
20. Osborne JW (2006) Gender, stereotype threat and anxiety: psychophysiological and cognitive evidence. J Res Educ Psychol 8:109–138
21. Ravi S, Kim J (2007) Profiling student interactions in threaded discussions with speech act classifiers. In: Proceedings of AI in education
22. Samuel K (2000) An investigation of dialogue act tagging using transformation-based learning. PhD Thesis, University of Delaware
23. Scardamalia M, Bereiter C (1996) Computer support for knowledge building communities. In: Koschmann T (ed) CSCL: theory and practice of an emerging paradigm. Mahwah, Erlbaum
24. Searle J (1969) Speech acts. Cambridge Univ. Press, Cambridge
25. Shawar BA, Atwell E (2005) Using corpora in machine-learning chatbot systems. Int J Corpus Linguist 10:489–516
26. Soricut R, Marcu D (2003) Sentence level discourse parsing using syntactic and lexical information. In: Proceedings of the annual conference of the North American chapter of the association for computational linguistics
27. Sporleder C, Lapata M (2005) Discourse chunking and its application to sentence compression. In: Proceedings of human language technology conference
28. Steele C (1997) Threat in the air: how stereotypes shape intellectual identity and performance. Am Psychol 52:613–629
29. Stolcke A, Coccaro N, Bates R et al (2000) Dialogue act modeling for automatic tagging and recognition of conversational speech. Comput Linguist 26(3):339–373
30. Witten IH, Frank E (2005) Data mining: practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco
31. Yang Y, Pedersen JO (1997) A comparative study on feature selection in text categorization. In: Proc of the 14th international conference on machine learning, pp 412–420

Jiehie Kim is the director of the Future Technologies Lab at KT. She received her B.S. and M.S. degrees in Computer Science and Statistics from Seoul National University, and a Ph.D. degree in Computer Science from the University of Southern California. She has been working at USC Information Sciences Institute as the Principal Investigator of many NSF (National Science Foundation) projects on social dialogue, pedagogical technologies, and intelligent interfaces, leading the Pedagogical Technologies (PedTek) group.

Jeon-Hyung Kang is a Ph.D. student at the Department of Computer Science, Viterbi School of Engineering, University of Southern California. He also works in the Information Integration Research Group, at Information Sciences Institute (ISI) as a Graduate Research Assistant. His research focuses on understanding and predicting human behaviors in online social network by building or using intelligent model.