MentorMatch: Using student mentors to scaffold participation and learning within an online discussion board

Erin SHAW, and Jihie KIM and Pachara SUPANAKOON
Information Sciences Institute, University of Southern California, CA USA

Abstract. In this paper, we present a novel approach to scaffolding student participation and learning within discussion forums using student mentors, i.e., course peers with relatively good understanding of a particular domain topic. First, we identify mentors using domain topic models, student discussion profiles, and a similarity comparison of the topics being discussed. Second, we provide an interface that encourages classmates to invite mentors to participate. The feature, named MentorMatch, was integrated into an undergraduate course discussion board. Some results are reported.

Keywords. Content Analysis & Indexing, Communications Applications, Information interfaces & presentation: Miscellaneous, Natural Language Processing, Social Learning Techniques, Discussion Forums.

Introduction

In this paper, we present a novel approach to scaffolding student participation and learning within discussion forums using student mentors, that is, course peers with relatively good understanding of a particular domain topic who respond frequently to posted queries. To connect student help-seekers to their question-answering peers based on topic knowledge, we first determine which topics within a discussion are relevant, and how to represent them computationally. We then create student profiles that summarize the types of messages and the topics about which each student contributes. The student profiles are used to formulate a model of potential student mentors on a given course topic.

The feature, named *MentorMatch*, was deployed in an undergraduate course discussion board in October, 2008. The board is an integral component of a project-based Operating Systems course in the Computer Science Department at the University of Southern California. Though the number of messages posted per project can reach two hundred, we often see very short threads and sometimes unanswered questions. Our goal is to promote deeper and more collaborative participation by recognizing class experts, identifying them to help-seekers, and encouraging them to participate in related conversations.

1. Computational approach

The process begins when a help-seeker initiates a new thread and posts a question. At this point, the system identifies the relevant topic in the question based on its knowledge about the topic representation of the domain, for example, a particular project assignment. The system searches for potential mentors by matching identified topics and individual student discussion profiles. The mentors found are listed as contacts at the top of the thread and the help-seeker is given the option to contact the mentors personally or automatically. In either case, email is sent to the mentors inviting them to participate in the discussion. The following sections describe our approach to modeling topics using domain terms and building student discussion profiles.

1.1. Topic modeling with domain terms

Supervised machine learning approaches to topic classification typically require a set of manually labeled data. It is difficult to generate consistent labels due to the high variance and noise of the discussions. Messages also often contain domain terms relevant to multiple course topics. We solved the data-labeling problem by creating topics models using a corpus of text about the relevant topics. The corpus consists of textbook term ontology and course lecture and assignment materials from both our own course and from similar courses whose materials were available online.

To generate features for modeling individual messages we used domain terms that were extracted from the index of the course textbook. Existing unigram-based models did not work well due to the multi-word domain terms and acronyms that are common in student discussions (e.g. virtual memory, RPC). To build a more effective model, we improved on our original approach to semi-automatically generate a domain term dictionary with term mappings from scanned images of a textbook glossary [1]. The new approach supports the use of multi-word terms and acronym mappings, which strengthens the creation of the feature vector of a message. The term features are extracted after the mappings are established. In our particular course there are four project assignments, each of which covers multiple course topics. Using the topic classifier on the course assignment materials, we identified the most relevant topics per assignment and weighted them appropriately. For example, Project 1 was comprised of three topics, *Nachos* (*w*=0.088), *Threads* (*w*=0.061) and *Deadlocks* (*w*=0.053). The weights are used in the analysis described in the next section.

We used TF*IDF (term frequency * inverse document frequency) transformations [6] to distinguish closely related topics in discussions. Each topic is modeled with a representative vector using the term ontology and course documents. The classification of each message is accomplished by calculating the lexical similarity score between topic vectors and message vectors for a given threshold. We also explored a latent semantic analysis (LSA) approach for modeling discussions. LSA has been used for topic analysis [3], and was appealing because the discussions contain a lot of noise. But unlike in other applications where topic categories are fairly differentiated [4], LSA was less effective than TF*IDF in modeling related course topics [2].

1.2. Student profiling with course topics

Student profiles were created to accumulate information about student activities. We track the topics in which the student participates and the types of posts made.

Each student's message is classified using topic vectors, and the topic similarity scores are stored in the profile. Messages are also classified based on positions in the thread, i.e. whether it is the first post or a response. More than 80% of student discussion threads start with a message containing a question [1]. This information is used to differentiate help-seekers from answer-providers. Yes/no responses and short acknowledgements will not have high similarity scores for relevant topic vectors. To incorporate these contributions into the profile, the similarity score for response message Mi and topic category Tc is computed as in Eq.(1).

$$Score\ (M_i, Tc) = w * Sim\ (M_i, Tc) + \sum_{k=1}^{i-1} w_k * Sim\ (M_k, Tc) \tag{1}$$

That is, the score of response message M_i includes the weighted similarity scores of previous messages, including the first message, i.e., $M_1,\ldots,M_{i-1}$. Our current implementation uses a uniform distribution over previous and current messages. The student contribution scores for all topics are accumulated over time and used to identify mentors who provide the highest number of responses on the topic. For each topic, the top three students within a given threshold are identified as mentors for the topic.

2. Future work and conclusion

Topic classification, student profiling, dynamic similarity matching, and mentor awareness and notification were integrated into a discussion board being used by undergraduate computer science students. The data show that help-seeking students took advantage of being able to contact mentors but that mentors did not necessarily help when it was suggested they do so. We think that removing the automatic contact option and asking help-seekers to personalize their request for help may further motivate the mentors. The data is uneven given the point in the semester the feature was introduced but there is high student-reported interest in the feature for both mentors and help-seekers, and the feature is currently available for the spring 2009 course. In conclusion, we hope the MentorMatch feature will serve two key purposes: The first is to get students the help they need, especially from peers, whose messages are more likely to be further discussed than an instructor's message. The second is to motivate students to participate, especially advanced students, again, to promote collaboration. In both cases the goal is to increase student comprehension and achievement.

Acknowledgements

The authors are indebted to Senior Lecturer Michael Crowley for his continuous support. The work was funded under a NSF CCLI Phase 2 program grant (award #0618859).

References

[1] 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 (AAAI-2006).

[2] Kim, J., Shaw, E., Chern, G. & Herbert, R. 2007. Novel tools for assessing student discussions: Modeling threads & participant roles using speech act and course topic analysis, Proc. of AIED, 2007.

[3] Landauer, T., Dumais, S. 1997. A Solution to Plato's Problem: The Latent Semantic Analysis Theory of Acquisition, Induction, and Representation of Knowledge, Psychological Review, 104 (2).

[4] Newman, Chemudugunta, Smyth, and Steyvers. 2006. Analyzing Entities and Topics in News Articles Using Statistical Topic Models. In LNCS -- IEEE ISI.

[5] Ravi, S., Kim, J. 2007. Profiling Student Interactions in Threaded Discussions with Speech Act Classifiers, Proceedings of AI in Education.

[6] Salton, G. 1989. Automatic Text Processing, The Transformation, Analysis, and Retrieval of Information by Computer. Addison-Wesley, Reading, MA.