Socio-semantic Conversational Information Access

Saurav Sahay[*]
Siemens Corporate Research
saurav.sahay@siemens.com

Ashwin Ram[†]
Palo Alto Research Center
ashwin.ram@parc.com

ABSTRACT

We develop an innovative approach to delivering relevant information using a combination of socio-semantic search and filtering approaches. The goal is to facilitate timely and relevant information access through the medium of conversations by mixing past community specific conversational knowledge and web information access to recommend and connect users and information together. Conversational Information Access is a socio-semantic search and recommendation activity with the goal to interactively engage people in conversations by receiving agent supported recommendations. It is useful because people engage in online social discussions unlike solitary search; the agent brings in relevant information as well as identifies relevant users; participants provide feedback during the conversation that the agent uses to improve its recommendations.

Categories and Subject Descriptors

H.4.0 [**Information Systems and Applications**]: General

Keywords

Conversations, Recommendations

1. INTRODUCTION

We introduce a natural language Socio-semantic Conversational Information Access method to facilitate agent assisted, socially filtered, semantically analyzed information access as a solution for interactive, collaborative and dynamic information access. Conversational Information Access is an interactive and collaborative information seeking interaction. The participants in this interaction engage in a conversation aided with an intelligent information agent (Cobot) that provides contextually relevant recommendations and connects relevant users together. This collaborative approach aims to engage users and raise awareness of relevant information, and improve the search and discoverability of relevant information. This work takes a knowledge centric and domain guided approach to information access. We have incorporated knowledge from domains such as health and Computer Science(CS) by creating semantic dictionaries extracted from large structured biomedical ontologies (ontology guided system) as well as CS social tags (tag assisted system) respectively. We also show that cheaply available, domain specific, socially generated tags are effective for domain specific conversational recommendations.

Conversational Information Access leverages the search and discovery process by integrating web information retrieval along with social interactions. A typical Google or Yahoo Answers experience is solitary and repetitive, while the conversational approach is collaborative, dynamic and interactive and aims to be engaging. Cobot monitors community interactions, uses domain specific knowledge for finding recommendations and brings relevant information to users by augmenting the conversations. Cobot's 'conversation engine' monitors user conversations with other users in the community and provides recommendations based on the conversation to the participants. Cobot's 'community engine' models conversations to capture user-user and user-information interactions. Cobot leverages collaborative and conversational information access by aiming to harness the collective knowledge of users and information from communities and web.

Intelligent question answering systems also make use of online social communities. For example, Aardvark [5] service allows a user to ask a question and get answered by another user in the user's extended network (including a user's friends' friends) by analyzing user profiles and past activities. This service is convenient for someone who is looking for an opinion from a person instead of a search engine. While this service benefits from information derived from the social network, it does not foster an organic growth of the community by sharing and leveraging from the information and connections created by users over time. IM–an–Expert [8] is another instant messaging based question answering service that identifies experts with potential knowledge about asked questions and routes the questions to these experts. User profiles are created in this service either explicitly by users by specification of keywords of interests or implicitly by extracting keywords from emails sent by users to mailing lists. Experts are matched by performing vector based searches using TF-IDF scoring multiplied by a temporal decay function to discount new messages from old ones.

[*]This work was done while this author was at College of Computing, Georgia Tech.

[†]This work was done while this author was at College of Computing, Georgia Tech.

Copyright is held by the International World Wide Web Conference Committee (IW3C2). Distribution of these papers is limited to classroom use, and personal use by others.
WWW 2012 Companion, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1230-1/12/04.

2. DESCRIPTION

The key dimensions of a recommendation system include factors such as relevance and timeliness of recommendations. To construct a successful conversational recommendation experience, it is critical to build a timely and actionable experience for the user. To build this effective experience, cobot leverages from different resources such as past interactions from similar conversations, external web based recommendations as well as people recommendations.

We depict high level cobot system architecture in Figure 1. Cobot processes conversations to extract semantic information for generation of queries, online and past conversation search, adding information to user models, real time indexing and filtering of results for recommendations.

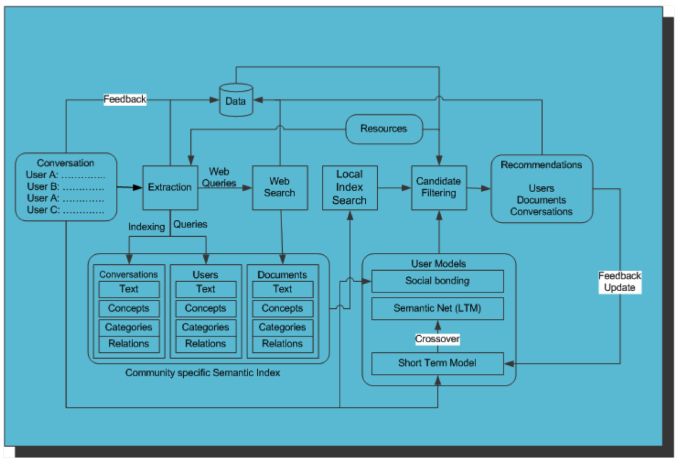

Figure 1: Semantic Components

The main functional components of cobot can be divided into the following components:

1. *Language Understanding and Search*
2. *User Modeling*
3. *Filtering and Recommendations*

We briefly describe each of the components for conversational recommendations in the following sections.

2.1 Language Understanding and Search

2.1.1 Intent Detection

Conversational interactions are classified into one of the following categories based on Verbal Response Modes linguistic theory[6].

- *Question*: Asking a question, e.g. somebody posts a problem. This is usually, but not always, the first post of a thread.
- *Disclosure*: Reveals thoughts, feelings, wishes, perceptions or intentions (declarative first person)
- *Edification*: States objective information
- *Advisement*: Attempts to guide behavior - suggestions, commands, permission, prohibition
- *Acknowledgement*: Being recognized or acknowledged
- *Reflection*: Repetitions, restatements and clarifications
- *Interpretation*: Judgement or evaluation of other's experience or behavior
- *Confirmation*: Compares speaker's experience with other's agreement, disagreement, shared experience or belief

Cobot uses this information during query generation and filtering stages to help make the decision if the agent should insert some type of recommendation into the conversation[9]. For example, if a question is asked in a conversation, cobot gives a higher weight to recommendations containing disclosures and advices. We have trained our Intent Detection classifier by annotating and training WebMD conversations threads. The accuracy of our classifier is close to 70% [11].

2.1.2 Semantic Tagging

Social conversational systems such as forums and Q&A sites have an intrinsic property of self-governance, regulation and evolution by its community. While the major challenge remains getting a critical mass, these social systems may require lesser internal knowledge organization and coordination for automated processing and maintenance. The problem with such social systems for automated processing is that the noise-signal ratio may become high due to informal nature of the language in conversations. Cobot normalizes these conversations to extract meaningful conceptual representations using the extensive UMLS ontology[1] with millions of concepts and a large CS/Math tags vocabulary. Cobot's internal knowledge representation system uses these extracted concepts for its knowledge representation.

2.1.3 Augmented Transition Network

We have developed a top-down backtracking search based parsing algorithm based on Augmented Transition Network to extract candidate query phrases from sentences in cobot. An augmented transition network is a directed graph in which parsing is described as the transition from a start state to a final state in a transition network corresponding to an English grammar [13]. The nodes represent states in the parse; each arc contains a test which must succeed for the arc to be traversed. If the arc is traversed, an action is performed. The parse proceeds by means of a depth-first search of the ATN; it succeeds if no more arcs are to be followed and end of input is reached. ATN was first used in LUNAR system, one of the first question answering systems [14]. An ATN is similar to a Finite State Machine in which labels or arcs between states can be calls to other machines. Arcs in an ATN may contain words, categories (e.g., noun phrase), they may push to other networks, or perform procedures.

We have built a fast shallow semantic parser (Figure 2) capable of extracting relationships, phrase focus and their properties using Augmented Transition Networks.

2.1.4 Query Generation

Cobot analyzes conversations to extract concepts, relationships between concepts and sentence focus of conversations to generate meaningful queries for bringing in relevant candidate results from various resources. We use OpenNLP chunker [4] to extract phrases and map them into concepts using UMLS ontology or CS tags depending on the domain of the conversation. We extract SVO triples [10] from sentences

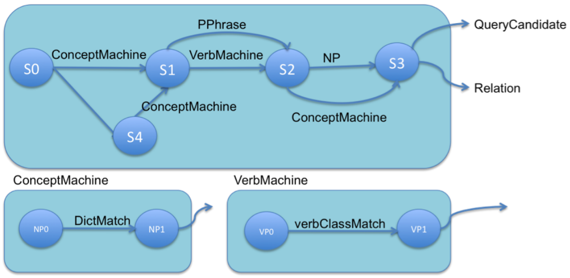

Figure 2: ATN Parser

as candidate queries to retrieve entities from the semantic index that closely match the context in conversations.

Cobot uses a mix of strategies in its knowledge goal and task goal for generating queries from conversations. The generated queries are sent to retrieval engines to generate a candidate pool of recommendations for downstream filtering processes. There are two query generation systems in Cobot, one for retrieving results from the web and another for retrieving results from Cobot's semantic search index that contains current (results retrieved for the current conversation) as well as past results from recommendations generated from community conversations. Cobot's knowledge based goal is to recommend specific content giving pointers to answers and related support and validation sources of information. Cobot's task goal is to recommend learning resources providing definitions, facts, methods, tutorials and other learning resources from informational pages, forums and Q&A sites.

2.2 User Modeling

Cobot takes a hybrid learning approach to user modeling by learning from user's past behavior as an indicator for her future behavior. This past behavior includes watching conversations that users are participating in, pages and conversations they are clicking and the explicit ratings they are providing on such entities. User Modeling in Cobot is designed to help users get notified about related conversations, documents and people at times they are being accessed in the community. User models are captured and learnt through content extraction from conversations, relevance feedback and click monitoring. Cobot maintains rich user models maintaining short term and long term profiles of the user. The process involves extraction and storage of domain concepts from user's conversations or other participating conversations and documents. Factors such as semantic similarity between concepts, recency of information, learning and unlearning of concepts, weights and their associations are modeled and used as filters for generating proactive right time user access user information. It is not just important to recommend right information to users but to recommend such information at 'now access' time or at times when there is some activity in the recommended information source . This helps in getting people engaged in conversations when it's happening in real time, thus providing access to such information at the right time. We feel that such a system would be very useful for e-learners and knowledge workers where the users can stay informed and connected on the latest community interactions on their topics of interests. Cobot does not aim to model user's understanding, but rather her potential knowledge and interests using concepts, ratings, semantic nets and some basic activity statistics. Such a system is inherently better suited for longer term learning tasks compared to other tasks such as single shot transactional web search and ad hoc retrieval. This modeling approach affords us to get direct user intervention as well, if needed, to update the users models. Many statistical machine learning based approaches do not provide this option to end users.

2.2.1 *Concept*

A concept forms the core of cobot's representation. Each concept represents a word along with associated information. The models store as much information about concepts as desirable coming from downstream processes. Where possible, concepts have associated semantic types, parts of speech, their overall and short term frequency counts, etc. represented in the system.

2.2.2 *Association*

Concepts are connected using links that are called associations. The associations capture co-occurrence of concepts. If any two concepts appear often together then they would develop a strong association. The strengths of association is a weight between concepts that is determined by factors such as co-occurrences in conversations and documents, learning and unlearning rate parameters in the semantic net. By default there is no association between concepts.

2.2.3 *Short Term Model*

The purpose of the Short-term model (STM) is to capture the user's short-term interests which are concepts collected from recent user interaction episodes from conversations, feedbacks and page clicks. The STM is marked by a sliding window

$$Score = Score \times (1 - 0.2)^x \quad (1)$$

In equation 1, x is the number of days since the last update.

2.2.4 *Crossover*

Crossover process takes instances from STM window and adds them to the LTM semantic net if the concepts to be added are above certain threshold frequencies. Currently all the qualifying concepts get added to the LTM, but a better approach would be to have some classifier decide if a particular instance should be discarded or added to the LTM. When a user engages in a conversation, rating or a document click episode, cobot determines if the time lapse between current interaction and last crossover operation stretches over the STM window size. If so, crossover operation is performed and last crossover operation times are updated for the user.

2.2.5 *Long Term Model*

The Long-term model (LTM) captures the user's long-term associated interests. This model tries to capture concepts and their co-occurrences that interest the user in general and for a prolonged period of time. We represent the LTM in the form of a Semantic Graph. The nodes of the graph are concepts the user is interested in. The concepts are connected with associations which develop when concepts co-occur frequently in user activities. Initially the LTM contains no concepts and starts building up after the first crossover operation. Over a period of time when the user engages in more conversational or interaction activity, new concepts are added to the LTM. All the concepts that appear in the LTM have a rating associated with them. The concept rating is computed using the following function(equation 2):

$$Rating = 1 - \frac{1}{Concept_{freq}} \quad (2)$$

For a new rating obtained for an already existing concept in the LTM, the rating is adapted as follows (equation 3):

$$Rating = Rating_{old} + LearningRate \times Rating_{new} \quad (3)$$

The Learning Rate parameter decides how much of the new rating to incorporate in the already existing rating. As mentioned earlier, the concepts are connected using associations in the LTM. Associations just capture co-occurrence of concepts in conversations and documents. When concepts belonging to a conversation, for example, are added to the LTM, they might strengthen old associations or create new ones. The strength of an existing association is updated using the following formula 4:

$$Association = Association_{old} \times (1 + LearningRate) \quad (4)$$

We have also implemented 'unlearning' or 'forgetting' of concepts from LTM that do not appear too often. The need for this was felt as many outlier concepts spring up in the LTM incidentally. So such concepts are slowly weakened and when they go below a certain threshold they are removed from the LTM permanently.

We have heuristically fixed the Short term model and long term model weights but these can be explicitly set by users. There are other parameters in the system as well that users can set for biasing the models to their preferences. These parameters define the user's inherent preference. If the user has some long-term information goals (learners, doctors, patients, programmers etc) then the LTM weights needs to be more and if the user is just looking for the latest current recommendations (a casual user) then STM weights should be more. Currently, these parameters are set manually, but we feel that they can be inferred as well by analyzing users' activity patterns.

3. RECOMMENDATIONS

In this section, we briefly describe some of the filters used in cobot social recommendation engine.

3.1 Social Capital Filter

Social capital in general refers to features of social organizations such as networks, norms, and social trust that facilitate coordination and cooperation for mutual benefit. Reciprocity is a key mechanism for explaining how social capital functions among individuals [2]. In the context of online social community, reciprocity means people benefit from the community and also give back to the community. In this sense, establishing social capital aligns with the goal of a social community, which is to serve its members while growing by members' supports. Moreover, social capital gained from online communities can also be transferred to offline contexts [7]. For example, Facebook is used by people to maintain weak social ties, such as staying in touch with acquaintances from high school, or to bond close ties, such as emotional support for family members. Although different types of social capital created from online communities are not equally convertible into economic, symbolic, or cultural capital offline, researchers suggest that positive social capital outcomes can include career advancement, better public health, and organizational success. We believe social capital gained from an online social learning community can be very valuable since this type of social capital represents one's education, professional skills, and expertise. In fact, It has been shown that directed communication between individuals have increased bonding social capital of a user [3].

In a community question answering(CQA) system, for example, there is generally no notion of explicit friendship network. However, there are some recent CQA systems such as Quora.com that allow for explicit asymmetric connections in the community to receive push notifications from the community. Besides the explicit network, there is a social network built implicitly based on user user interactions in the community. We leverage from the works of [12] to compute this implicit and explicit affinity networks in the community.

$$bonding(i, j) = s_{ij}^{IAN} \times s_{ij}^{ESN} \quad (5)$$

$$bridging(i, j) = (1 - s_{ij}^{IAN}) \times s_{ij}^{ESN} \quad (6)$$

In Equations 5 and 6, s_{ij}^{IAN} is the score of the Implicit Affinity Network between users i and j, and s_{ij}^{ESN} is the score of the Explicit Affinity Network between users i and j. The implicit affinity formulas work on a set of attributes (here topics, groups, description, school), where each attribute has a set of possible values (for example, topics=math, health sciences, cs, physics, biology). The attribute affinity scores are the number of values in common divided by the total number of possible values, so for each attribute, the affinity is higher if the users have more values in common. The overall implicit affinity score is the sum of the affinity scores for each attribute divided by the number of attributes the two users both have values for. The explicit affinities are 1 if users have ever had a directed communication in the system before.

In cobot, when a user asks a question or responds in a conversation, the bonding capitals between the participants in the conversation and the asker/responder is computed and used as a signal to boost the recommendations.

3.2 Feedback Filter

Besides filtering based on user's bonding capital, cobot also monitors user clicks and explicit feedbacks and registers them at individual preference level, conversational level and the entity level. For example, if a document receives a positive rating in a conversation, the document rating increases (entity level), the user-document rating increases (individual level) and the recommended document rating increases (conversation level). This helps in promoting community preferred results as recommendations in the system. If a recommendation has received better than average rating in the system in past, cobot adds a small factor (normalized average rating) to the final score of the candidate recommendation. The implicit feedback through user clicks is not currently being used in score modifications but this is an important feature that will play a role in a recommendation system in live deployments.

4. EVALUATION AND EXPERIMENTS

We partnered with an existing e-Learning website called Openstudy.com that allows users to ask questions and participate in conversations on different study related areas such as health and biomedical domain, as well as the sciences and arts domain. The site enables real time conversational interactions between students/members to post questions, receive responses and study together with other students. We developed a browser script that, once installed on the browser, could transfer conversations from this site to the cobot server, process them in realtime and send back recommendations. In the process, cobot indexed the conversations and recommendations along with capturing the user models for the users in different conversations.

We conducted experiments to evaluate the conversational recommender for web based recommendations. We evaluated the tag assisted system for Math/CS domain conversation dataset and the ontology guided system (based on UMLS ontology) using the Health domain conversation dataset. We obtained relevance judgement ratings for the conversational recommendations using Amazon Mechanical Turk(AMT) platform. Amazon's Mechanical Turk is a crowdsourcing marketplace in which anyone can post tasks to be completed by paying for it. These micro-tasks, also known as Human Intelligence Tasks, are chosen by 'workers' to be completed. Once the worker has completed the task, the requester accepts or rejects the task and can also initiate further dialogue with the worker if needed. The success of this method of experimentation lies in how clearly one has created these micro-tasks, how well they have explained what needs to be done and provided enough information for the worker to complete the task. Also, the requester can require certain types of qualifications on these tasks. For example, we only wanted workers from U.S. region to complete our tasks since we thought this group could easily understand the language in the task and the problem context.

4.1 Datasets

We used our web based widget to collect data from Openstudy by fetching conversations and populating cobot database with semantically processed information. We collected data from several study groups on the site. We divided the dataset into two classes corresponding to tag assisted recommendation system (tags from StackOverflow data) and ontology guided recommendation system (using UMLS based biomedical ontologies). We extracted conversations from the following study groups:

- Tags: *Mathematics*
- Ontology: *Biology, Chemistry and Health Sciences*

Table 1: Dataset

Type	#c	#words	#episodes
Ontology	77	18.2	2.12
Social Tags	119	16.8	3.37

where #c is the number of conversations, #words is the average length of words in conversation and #episodes is the number of interactive recommendations generated by cobot on average for each conversation.

Table 1 describes some properties of the dataset. We sequentially retrieved conversations from the site on a particular day without any bias to the kind of conversations being retrieved from the site.

4.2 Experimental Setup

We processed the data, generated recommendations and sent them to Amazon Mechanical Turk for annotation. We created 3 assignments for each conversation that contained recommendations at different rank positions and at different conversation depths. The question in the conversation generated a maximum of 3 recommendations and responses in conversation generated a maximum of 1 recommendation. We limited the recommendations to this small number as we didn't want to inundate and distract the user in her conversation and focus her attention to a few web recommendation results. We asked the workers to rate the recommendations on a scale of 1 to 5, with 5 being 'Very good' or 'Informative and Helpful' recommendation and 1 being 'Very bad' or 'Where did this come from?' relevance judgement.

Our goal in these AMT experiments was to assess the quality of the recommended documents in conversations. We plotted graphs for the average ratings provided by the AMT workers, the best recommendation rating out of the three recommendations we suggested to the users and interactive recommendations at different trigger points in the conversation. We also modeled the task from Information Retrieval perspective and calculated the Mean Average Precision (MAP) scores for the data we had processed. MAP scores for a set of queries is the mean of the average precision scores for each query. For computation of MAP values, we assumed all recommendations to be relevant if they got an average rating value equal to or above 3 in the conversation.

4.3 Ontology guided Web Recommendations

The ontology based recommendation modules are the most complex modules in cobot with millions of terms in its vocabulary along with semantic types and synonyms of the biomedical terms in its knowledgebase. For this system, we fetched conversations from three different biomedical domains including Biology, Chemistry and Health Sciences. These conversations were the most subjective and domain specialized conversations in the system being asked by students taking online courses for these subjects in graduate study programs.

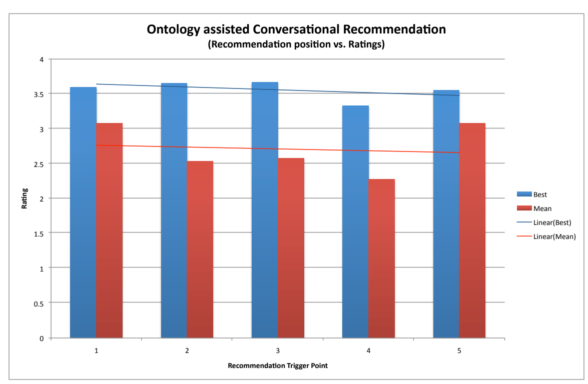

Figure 3: Ontology based recommendations - Overall

Figure 3 shows the best and the mean ratings at different

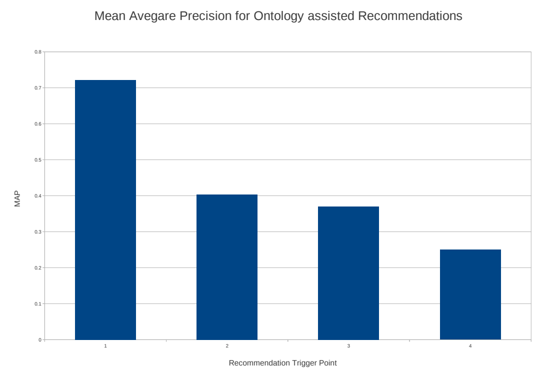

Figure 4: Ontology based recommendations - MAP

trigger points (question recommendations and response recommendations) in conversations. Figure 4 shows the MAP values for the ontology guided recommendation system. We observe that the quality of recommendations decreased as the interactivity in the conversation increased. This decrease in relevance is attributed to the increased contextual complexity in interactive conversations.

The goal of this experiment was to understand the individual contribution of the ontological support in cobot recommendation engine. We repeated the same experiments for the health/medical conversations, this time not using the ontology but the extracted keywords from the conversations for query formulations and search. For conduction this experiment, we generated recommendations on the same dataset that we used for ontology assisted recommendations and got ratings for AMT workers for this study. Note however that rest of the modules remained intact in cobot (like recommendation filters such as speech act filters etc). Figure 5 and 6 show the overall average ratings and MAP score based results of the Ablation Study on the tags dataset. We plotted the relevance of the conversational recommendations for short (<15 words), medium (between 15 and 30 words) and large conversations (>30 words) based on length of question. It is interesting to note that ontology support didn't enhance the performance in cobot for short conversations but contributed to increased scores for medium and long conversations.

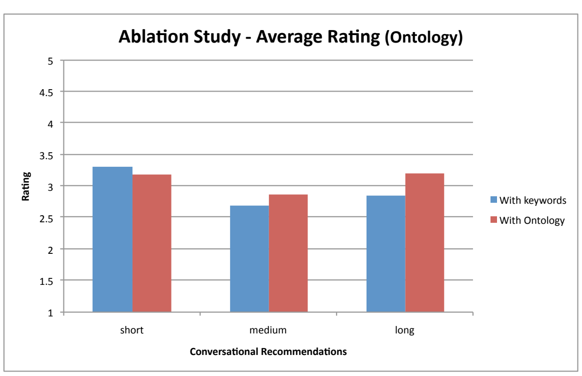

Figure 5: Ablation Study - Ratings

4.4 Tag assisted Web Recommendations

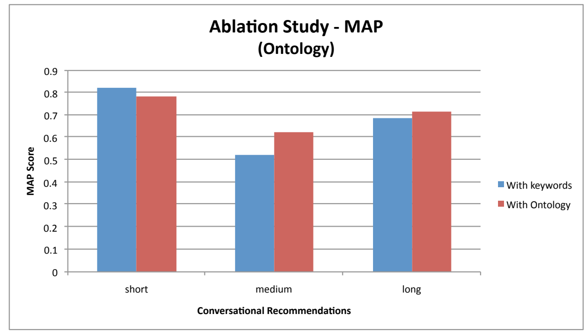

Figure 6: Ablation Study - MAP Scores

Figure 7 depict similar relevance judgement scores by AMT workers on conversations from Mathematics domain and different lengths and different interactive levels. We did not have any long conversations in our dataset for the tag assisted conversational recommendations.

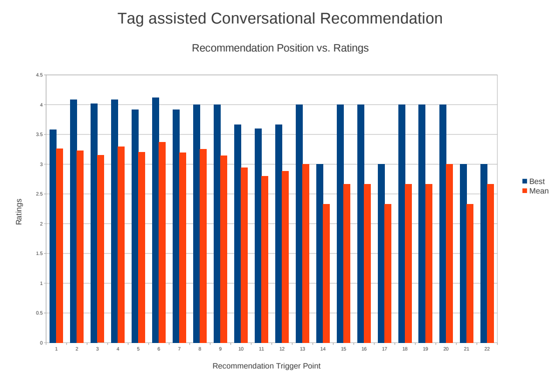

Figure 7: Tag based recommendations - Overall

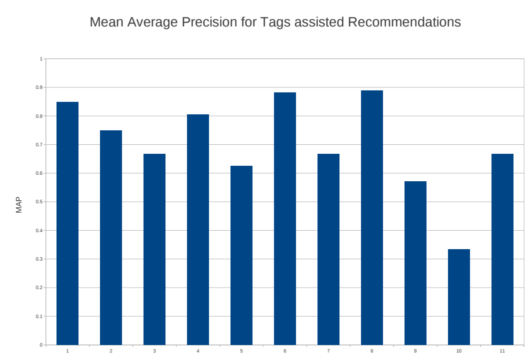

Figure 8: Tags based MAP

Figure 8 shows the MAP values for the tag assisted recommendation system. The quality of the tags assisted recommendations decreased very gradually at different trigger points in the conversation.

We repeated the same experiments for the Math/CS conversations, this time not using the tags but the extracted keywords from the conversations for query formulations and search. As with the case of Medical/Health conversations, the rest of the modules remained intact in cobot (like recommendation filters such as speech act filters etc). Figure 9 and 10 show the overall average ratings and MAP score based results of the Ablation Study on the tags dataset.

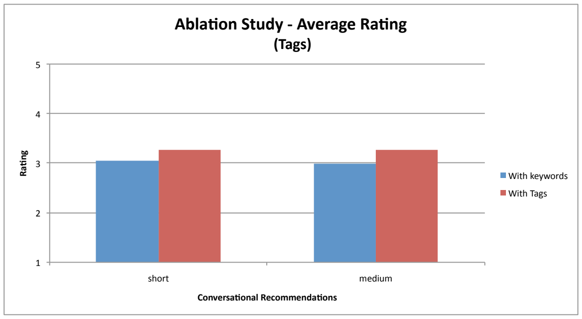

Figure 9: Ablation Study - Ratings (Tags)

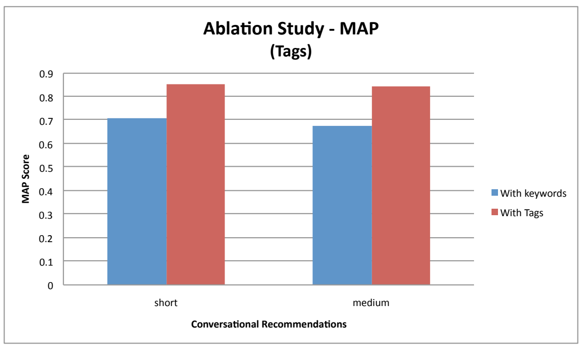

Figure 10: Ablation Study - MAP Scores (Tags)

It is interesting to note that tags support enhanced the performance in cobot for both short and medium sized conversations.

4.5 User Models

In User Model Adaptation, information about the learner (user) is evaluated and updated, if needed, with every system episodic interaction. This process requires a continuos addition and/or removal of user model knowledge, knowledge about the concepts, number of times they occurred, when they occurred, concept co-occurrences, associations developed, unlearning and decay with time, etc. Since learner characteristics are not constant properties, a change over time has to be considered by the learner model. Since our social recommendation infrastructure is closely tied to the user models, an effective social recommender would imply that the user models have the ability to capture and update the models well to make good social recommendations in the system.

4.5.1 Social Capital Contribution

We plotted the social bonding interaction network on our dataset as shown in Figure 11. This graph shows the user interaction network in the system. We observe that there were few users in the system that had strong bonding networks while others had had few interaction episodes with shared concepts together that created bonding capital between them. Therefore, this module contributed scores towards picking up users who had spoken about similar concepts with each other before. We think of this module as

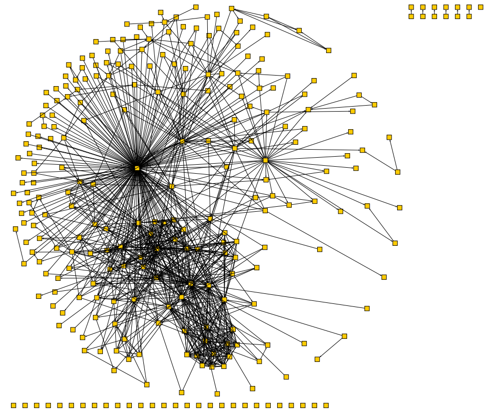

Figure 11: Community Implicit Capital

being very important in real community deployed systems since it is a common phenomenon that users develop social bonds with others through interactions and may continue to do so because of social bonding.

5. DISCUSSION

We present a summary of the main contributions in this work. We proposed a socio-semantic community based conversational recommendation system and ideated that such a system would help address users' information access problem. We conducted some experiments, evaluations and user studies with the system and got a mixed bag of different results.

We provide a summary of our findings as follows:

- The ontology supported bio-medical domain recommender got overall best maximum ratings by mechanical turk workers for long conversations (> 30 words).
- The social tags based recommender for the Mathematics domain got overall best average ratings and MAP scores for short and medium length conversations.
- Ratings for recommendations stayed about the same or improved with increase in both length (size of question/response) and height (number of interactions in conversation) of conversation.
- For short conversations, extracted keywords (slightly) outperform the ontology based setup (on average and based on MAP scores), given every other recommendation module stayed the same in cobot (Ablation Study).
- For short conversations, tag assisted recommendations outperform the extracted keywords based recommendations(on average and based on MAP scores), given every other recommendation module being the same in cobot.
- For medium and long conversations, both tags and ontology supported recommendation system did better than extracted keywords (on average and based on MAP scores).
- In a small user study, users reported that the conversational recommendation system was useful for long, detailed and specific questions.

We organize our contributions and findings along the following primary dimensions and sources of power for this work, i.e. blended recommendation environment, knowledge based information architecture and evidence based recommendations.

- *Blended recommendation environment.* Cobot provides a unique blended recommendation environment with multi-modal recommendations for conversational content. This setup helps in reduction of additional search effort due to the system's ability to extract multiple queries automatically and bring in different recommendation results. Recent advances and successes in blended learning models also suggest that 'blended learning' is more effective than traditional learning alone, especially when the blended learning experiences are well designed. There are two main components of a successful blended learning experience. One is access to consistent and reliable online content, and the other is contextual and timely human interaction. Cobot provides both components and therefore has the potential to be a practical recommendation system providing content access and an engaging experience through expert social interaction together.

- *Knowledge based information architecture.* Cobot is a knowledge based domain adaptable information processing system. It bootstraps on the recognized evidential knowledge to trigger downstream extraction components for generating queries for semantic search, indexing of extracted data, user model update with extracted knowledge and semantic filtering or candidate results for final recommendation generation. The main advantage of this approach over purely statistical approaches center around having a handle on precise knowledge in recommendation candidates for applying different usecases for integrated reasoning, decision support and problem solving. With proper engineering effort, a knowledge based system with lexico-syntactic rules and patterns, effecting parsing and applied reasoning can result in very effective decision support recommendation system. Our goal in this work was not to constrain the agent to a particular sub-domain like diseases, treatments or drugs related conversations, etc. but to stay as generic and automated as possible using large controlled vocabularies such as the UMLS ontology for generic domain recommendations using social conversations.

- *Evidence based Recommendations.* Cobot's ability to bring in new knowledge from external web data, process and extract that knowledge and use it as past cases while scoring and evaluating new candidates against the conversation makes it an experience gathering recommendation system. This also gives it the ability to promote community preferred results by using the community preferences as an evidence filter for final recommendation generation. Cobot retrieves candidates in real time from trusted sources thus making sure that fresh evidences are evaluated along with past episodes for generating the final recommendations. Cobot also uses social metrics based on social capital theory to promote such social recommendations that have had interactions in the past along with people in the conversation.

6. REFERENCES

[1] A. R. Aronson. Effective mapping of biomedical text to the umls metathesaurus: The metamap program, 2001.

[2] L. Berkman and I. Kawachi. *Social epidemiology.* Oxford University Press, 2000.

[3] M. Burke, C. Marlow, and T. Lento. Social network activity and social well-being. In *Proceedings of the 28th international conference on Human factors in computing systems*, CHI '10, pages 1909–1912, New York, NY, USA, 2010. ACM.

[4] M. P. . U. H. E. Buyko, J. Wermter. Automatically adapting an nlp core engine to the biology domain. In *In Proceedings of the Joint BioLINK-Bio-Ontologies Meeting, Fortaleza, Brasil*, pages 65–68, 2006.

[5] D. Horowitz and S. D. Kamvar. The anatomy of a large-scale social search engine. In *WWW*, 2010.

[6] A. Lampert, R. Dale, and C. Paris. Classifying speech acts using verbal response modes, 2006.

[7] S. Nah. Connecting social capital offline and online: The effects of internet uses on civic community engagement. In *The American Association for (AAPOR) 59th Annual Conference, 2004 and WAPOR 57th Annual Conference, 2004*, 2004.

[8] M. Richardson and R. W. White". Supporting synchronous social qna throughout the question lifecycle. In *WWW'11*, 2011.

[9] S. Sahay, H. Ai, and A. Ram. Intentional analysis of medical conversations for community engagement. In *FLAIRS Conference'11*, 2011.

[10] S. Sahay, S. Mukherjea, E. Agichtein, E. V. Garcia, S. B. Navathe, and A. Ram. Discovering semantic biomedical relations utilizing the web. *TKDD*, 2(1), 2008.

[11] S. Sahay and A. Ram. Conversational framework for web search and recommendations. In *In Proceedings of the ICCBR Workshop on Reasoning from Experiences on the Web*, 2010.

[12] M. Smith, C. Giraud-Carrier, and N. Purser. Implicit affinity networks and social capital. *Information Technology and Management*, 10(2):123–134, Sept. 2009.

[13] W. A. Woods. Transition network grammars for natural language analysis. *Commun. ACM*, 13:591–606, October 1970.

[14] W. A. Woods. *Semantics and quantification in natural language question answering*, pages 205–248. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1986.