# SOS: A Distributed Mobile Q&A System Based on Social Networks

*A Janardhan*
*Research Scholar,Department of Computer Science and Engineering,OPJSUniversity,India*

**Abstract—**

Recently, emerging research efforts have been focused on question and answer (Q&A) systems based on social networks. The social-based Q&A systems can answer nonfactual questions, which cannot be easily resolved by web search engines. These systems either rely on a centralized server for identifying friends based on social information or broadcast a user's questions to all of its friends. Mobile Q&A systems, where mobile nodes access the Q&A systems through Internet, are very promising considering a rapid increase of mobile users and the convenience of practical use. However, such systems cannot directly use the previous centralized methods or broadcasting methods, which generate high cost of mobile Internet access, node overload, and high server bandwidth cost with the tremendous number of mobile users. We propose a distributed Social-based mObile Q&A System (SOS) with low overhead and system cost as well as quick response to question askers. SOS enables mobile users to forward questions to potential answerers in their friend lists in a decentralized manner for a number of hops and then resort to the server. It leverages lightweight knowledge engineering techniques to accurately identify friends who are able to and willing to answer questions, thus reducing the search and computation costs of mobile nodes. The trace-driven simulation results show that SOS can achieve a high query precision and recall rate, a short response latency and low overhead. We have also deployed a pilot version of SOS for use in a small group in Clemson University. The feedback from the users shows that SOS can provide high-quality answers.

## I. INTRODUCTION

Traditional search engines such as Google [1] and Bing [2] have been significantly impacting our everyday lives in information retrieval. To improve the performance of search engines, social search engines [3–10] have been proposed to determine the results searched by keywords that are more relevant to the searchers.

These social search engines group people with similar interests and refer to the historical selected results of a person's group members to decide the relevant results for the person. Although the search engines perform well in answering factual queries for information already in a database, they are not suitable for non-factual queries that are more subjective, relative and multi-dimensional (e.g., can anyone recommend a professor in advising research on social-based Q&A systems?), especially when the information is not in the database (e.g., suggestions, recommendations, advices). One method to solve this problem is to forward the non-factual queries to humans, which are the most "intelligent machines" that are capable of parsing, interpreting and answering the queries, provided they are familiar with the queries. Accordingly, a number of expertise location systems [11–14] have been proposed to search experts in social networks or Internet aided by a centralized search engine. Also, web Q&A sites such as Yahoo!Answers [15] and Ask.com [16] provide high-quality answers [17] and have been increasingly popular. Recently, emerging research efforts have been focused on social network based question and answer (Q&A) systems [17–22], in which users post and answer questions through social network maintained in a centralized server.

As the answerers in the social network know the backgrounds and preference of the askers, they are willing and able to provide more tailored and personalized answers to the askers, enhancing the satisfaction on the Q&A sites. The social-based Q&A systems can be classified into two categories: broadcastingbased [17–19] and centralized [20–22]. The broadcastingbased systems broadcast the questions of a user to all of the user's friends. The centralized systems [20–22] rely on a centralized server to identify possible answerers to a question without broadcasting. The centralized server constructs and maintains a social network of users, and searches the answerers for a given question from the asker's friends, friends of friends and so on. In respect to the client side, the rapid prevalence of smartphones has boosted mobile Internet access, which makes the mobile Q&A system as a very promising application. The number of mobile users who access Twitter [23] increased 182% from 14.28 million in Jan 2010 to 26 million in Jan 2011. It was estimated that Internet browser-equipped phones will surpass 1.82 billion units by 2013, eclipsing the total of 1.78 billion PCs by then [24]. The mobile Q&A systems enable users to ask and answer questions anytime and anywhere at their fingertips. However, the previous broadcasting and centralized methods are not suitable to the mobile environment, where each mobile node has limited resources. Broadcasting questions to all friends of a user generates a high overhead to the friends, since many friends (including those unlikely to answer questions) receive questions. Also, broadcasting results in many costly interruptions to users by sending questions that they cannot answer and increase their workload of looking for questions that they can answer through a pool of received questions. The centralized methods,by serving a social network consisting of hundreds of millions of mobile users (which are also rapidly increasing), suffer from high cost of mobile Internet access, high query congestion, and high server bandwidth and maintenance costs. It was reported that Facebook spent more than 15 million per year for server bandwidth costs and data center rent in addition to 100 million for purchasing 50,000 servers to release the high burden of traffic

To tackle the problems in the previous social-based Q&A systems and realize a mobile Q&A system, a key hurdle to overcome is: How can a node identify friends most likely to answer questions in a distributed fashion? To solve this problem, in this paper, we propose a distributed Social-based mObile Q&A System (SOS) with low node overhead and system cost as well as quick response to question askers. SOS is novel in that it achieves lightweight distributed answerer search, while still enabling a node to accurately identify its friends that can answer a question. We have also deployed a pilot version of SOS for use in a small group in Clemson University1. The analytical results of the data from the real application show the highly satisfying Q&A service and high performance of SOS

SOS leverages the lightweight knowledge engineering techniques to transform users' social information and closeness, as well as questions to IDs, respectively, so that a node can locally and accurately identify its friends capable of answering a given question by mapping the question's ID with the social IDs. The node then forwards the question to the identified friends in a decentralized manner. After receiving a question, the users can decide to forward the question or answer the questions if they can. The question is forwarded along friend social links for a number of hops, and then resorts to the server. The cornerstone of SOS is that a person usually issues a question that is closely related to his/her social life. As people sharing similar interests are likely to be clustered in their social network [26], the social network can be regarded as social interest clusters intersecting with each other. By locally choosing the most potential answerers in a node's friend list, the queries can be finally forwarded to social clusters that have answers for the question. As the answerers are socially close to the askers, they are more willing to answer the questions compared to strangers in the Q&A websites. In addition, their answers are also more personalized, trustable and accurate

SOS is featured by three advantages:

(1) Decentralized. Rather than relying on a centralized server, each node identifies the potential answerers from its friends, thus avoiding the query congestion and high server bandwidth and maintenance cost problem
(2) Low cost. Rather than broadcasting a question to all of its friends, an asker identifies the potential answerers who are very likely to answer this question, thus reducing the node overhead, traffic and mobile Internet access
(3) Quick response. Due to the close social relationshipbetween the question receivers and an asker, the question receivers are likely to be willing to provide answers quickly.

The contributions are summarized as follows:

(1) As far as we know, it is the first work to design a distributed Q&A mobile system based on social networks, which can be extended to low-end mobile devices. The system can tackle the formidable challenge facing distributed systems: precise answerer identification.

(2) We propose a method that leverages lightweight knowledge engineering techniques for accurate answerer identification.

(3) We propose a method that considers social closeness in addition to interest similarity in question forwarder selection in order to increase the likelihood of the receiver to answer/forward the question.

(4) We have conducted extensive trace-driven simulations based on the crawled data from Yahoo!Answer and Twitter with regards to node interactions in online Q&A systems and online social networks. Experimental results show the high answerer identification accuracy, low cost and short response delay of SOS.

(5) We have deployed a pilot version of SOS for use in a small group in Clemson University and revealed interesting findings in the mobile social-based Q&A system. Though Google earns a little higher user satisfaction degree than SOS on factual questions, users have much higher satisfaction degree on SOS for non-factual questions than Google. Also, socially close users tend to respond questions quickly.

Note that we do not endorse a complete removal of the centralized server from the system. We believe that dedicated servers still play an important role in the system, particularly when a node cannot find answerers in the social network. The rest of the paper is organized as follows. Section II presents related work. Section III present the design of SOS. Section IV and Section V show the trace-driven simulation results and real testbed results. We conclude this paper with remarks on future work in Section VI.

## II. RELATED WORK

While there has been relatively little research on distributed Q&A systems based on social networks, we take a slightly larger view of the problem space and compare SOS with social search, expertise location, and online Q&A systemsSocial search: In order to improve the user experience in a web search engine [1, 2], a number of works have been proposed to enable users to find resources by using social annotations or bookmarks. Evan et al. [3] pointed out that social interactions play an important role throughout the search process, and suggested that sharing search information among people may be valuable to individual searchers. The Phoaks [4], Answer Garden [5] and Designer Assistant [6] social search systems attempted to enable social interactions when existing information spaces are inadequate in providing experts' contact information. Amitay et al. [7] assumed that the interests of a searcher's friends provide a good prediction for the searcher's preferences, David et al. [8] proposed to re-rank the searched results by considering the strength of therelationship between the results and the searchers. Kolay et al. [9] studied how social bookmarked URLs lead to new or high-quality content on the Web. Bao et al. [10] proposed a SocialSimRank algorithm to calculate the similarity between social annotations and web pages as well as a SocialPageRank algorithm to capture the popularity of web pages. However, the social search aims to improve the web search engine, which perform poorly in non-factual questions [18]

Expertise location: Chen et al. [11] proposed an open system to recommend potential research collaborators for scholars and scientists based on the structure of the coauthor network and a user's research interests. Lin et al. [12] introduced SmallBlue, which is a social network search engine used to help IBM employees find and access expertise and information through their own social networks. ReferralWeb [13] mined public Web documents for the knowledge about potential experts through webpage content analysis. Expertise Recommender [14] studied software source control systems and technical support databases in order to find expertise. However, these systems only try to identify experts, but do not have mechanisms to ensure that the identified experts are willing to help. Online Q&A systems: Numerous online Q&A systems exist in the

Internet [15, 16], in which anonymous users can post questions and respond to others' questions. However, the systems cannot guarantee quick response of posted questions. Morris and Teevan [18, 19] studied how people use status message in a social network to ask questions. By posting questions on his/her status wall, a user can broadcast the questions to all of his/her friends. Hsieh et al. [17] proposed a marketbased Q&A service called MiMir, in which all questions are broadcasted to all users in the system. However, broadcasting a user's question to all of his/her friends only enables direct friends to see the question, generates high cost and produces interruptions to friends who are unable to answer the question. White and Richardson [20, 21] developed a synchronous Q&A system called IM-an-Expert, which automatically identifies experts via information retrieval techniques and facilitates real-time dialog via instant messaging without broadcasting. However, it also focuses on the direct friends of a user, and the synchronous communication faces challenges such as interruption costs and the availability of friends during the questioning time. Aardvark [22] is a centralized Q&A system, in which the centralized server receives a user's question, identifies and forwards the question to the most appropriate person in the Aardvark community. However, the centralized system structure may suffer from high query congestion, high server bandwidth and maintenance costs. As far as we know, SOS is the first distributed Q&A system that enables nodes to forward queries to efficiently and accurately find answerers.

## III. SYSTEM DESIGN

### A. Question Routing

SOS incorporates an online social network, where nodes connect each other by their social links. As shown in Figure 1, a registration server is responsible for node registration. Each user has an interest ID, which represents his/her interest

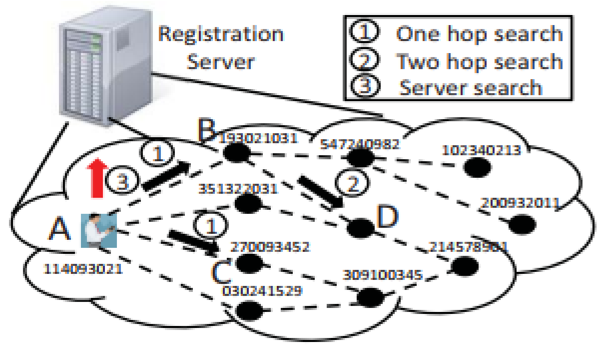


Fig. 1: Querying process in SOS.

The closeness of two user's interest IDs means the similarity between the two users' interests. Users sharing more common interests with an asker are more likely to be able to answer the asker's questions. Also, users having shorter social distances with an asker are more likely to be willing to answer the asker's questions

SOS has a metric similarity (S) that measures the likelihood of a node to be able and willing to answer another node's question. It is determined by the interest similarity between the question's interest and the receiver's interest as well as the social closeness between the question receiver and sender. SOS defines a constant K, which is the largest number of friends that a node can send/forward a question in its friend list. SOS allows each node to define TTL, which is the maximal number of hops that a question can be forwarded. A node determines TTL depending on how urgent the question is. Figure 1 shows the question routing process in SOS. After asker A initiates a question, it forwards the question to the top K friends (nodes B and C) who have the highest S in its friend list with the question. A question receiver replies to A if it has an answer for the question. Otherwise, the user forwards the question to its top K friends in its own friend list in the same manner (B to D) and reduces TTL by 1. The question is forwarded along node social links until TTL=0. If the question initiator has not received an answer after delay above its specified threshold corresponding to TTL (e.g., 1 hour), it sends the question to the server that holds a discussion board, which can be accessed by all users in the system. The discussion board serves as a store for unsolved questions in the distributed system. Then, the questions in the discussion board are handled as in online Q&A systems. From this process, we can see that three problems need to be resolved.

• How to derive the interests of a question or a user (Section III-B)?

Question/User Interest Representation When a user first uses the SOS system, s(he) is required to complete his/her social profile such as interests, professional background and so on. Based on the information, the registration server recommends friends to the user, and the user then adds friends into his/her friend list. The friend lists along with the profiles of the friends are stored in the local database of the user as shown in Figure 2. Users A, B and C connect with each other based on their social relationships, and each user has a social profile. Each node maintains the social identify representation (social ID in short) of each of its friends, which is used to measure the capability and willingness of a friend to answer the node's question. The social ID of a friend is retrieved by preprocessing the social information of the friend. As shown on the right part of Figure 3, to preprocess a friend based on his/her social information, the node first derives the first-order logic representation (FOL) [27], then conducts first-order logic inference to infer the friend's interests, from which it retrieves interest ID. The node then combines the interest ID with the social closeness between itself and the friend to calculate the friend's social ID (we will explain the combination in Section III-D) represented by a numerical string (e.g., 3202001001)

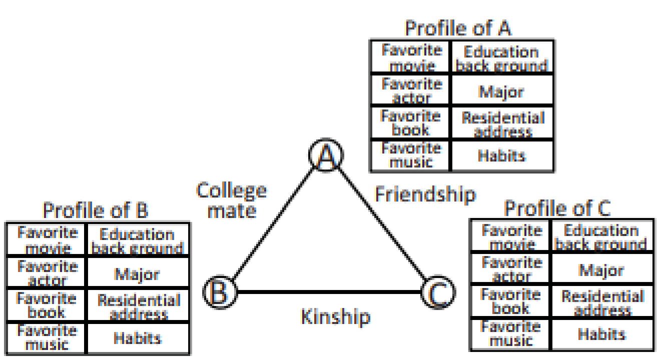


Fig. 2: An example of a node's social network.

Figure 3 shows the local answerer selection process for forwarding a question in one mobile node in the SOS system. To parse a question, the node first processes the questions in the nature language, it then represents the question in the FOL format and uses the FOL inference to infer the question's interests. Finally, it transforms the question to a numerical string (question ID). After the node parses its initiated question to an ID, it then finds the top K friends whose social ID are closest to the question's ID. Subsequently, it forwards the question to the identified friends. For instance, an asker may ask a question "Where is the best place to watch the movie Avatar in Clemson?". The corresponding keyword list of this question is resolved to the FOL format [where, place, movie, Avatar, Clemson] after natural language processing. After the FOL inference, the FOL format is changed to [movie(sci-fi), director (James Cameron), place(Clemson)], which is later encoded as a numerical string such as 3200001000. Similarly, a student in Clemson University who likes to watch sci-fi movie is represented as [movie(sci-fi), career(student), place(clemson)] after the FOL inference and be further encoded as 3202001001. Because the

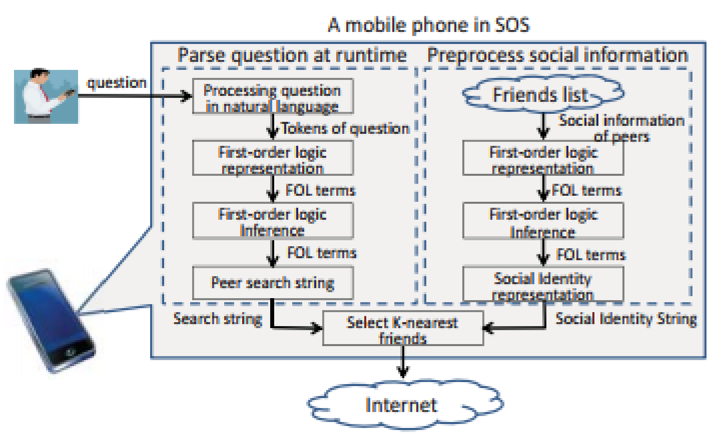


Fig. 3: Answerer selection process for forwarding a question in one node.

Fig. 3: Answerer selection process for forwarding a question in one node. student's social ID is close to the question's ID, he is identified as one of the K top friends to send the question to. By comparing the similarity between a question's ID and its friend's social ID, a node can identify its friends that are willing and able to answer/forward questions. More details of the parsing

process for a question or for a user is demonstrated in Figure 4 and Figure 5, respectively. The figures list the three steps in the process: FOL representation, FOL inference, and ID transformation. Below, we introduce the details of the three steps

1) Preliminary of the first-order logic (FOL): FOL is a powerful tool to describe objects and their relationships in real life. In FOL, the users need to define basic rules or axioms, which serve as the base of the inference. For example, the FOL for an axiom in nature language "All computer science (CS) male students who like reading like sci-fi movies" is
2) First-order logic representation: A question or user profile information is always expressed in the natural language. To convert a question or profile information into a format that a computer can understand, we can use part-of-speech tagging [28] or Modern natural language processing (NLP) techniques [29] to divide the question into a group of related words expressed by words, 2-word phrases, the wh-type (e.g., "what", "where" or "when"). Then we transform questions into the FOL representation. First, we parse the natural language into token keywords. These token keywords will be the constant symbols in the FOL representations. The step 1 in Figure 4 shows an example of FOL representation of the query. The keywords of the question "Where is the best cinema in location A?" is "cinema" and "location A".
3) mation includes his/her profile (e.g., job, hobby, favorites) and the social relationships with other users (e.g., kinship, colleague, classmate). A user's local database stores his/her own profiles, social relationship to and the profile of each of his/her friends. Specifically, a node first represents its profile in the form of name-values pairs such as "movies: Avatar, The Social Network", "music: Hey, Jude". That is, each interest is indexed by a unique name (e.g., movie, music), and the interest can have several values. The syntax name(value) is then transformed to the FOL representation expressed by predicate symbols. For example, the FOL representation of the previous example is "movie(Avatar)","movie(The Social Network)", "music(Hey, Jude)"

## CONCLUSION

In this paper, we have described the design and implementation of a distributed Social-based mObile Q&A System (SOS). SOS is novel in that it achieves lightweight distributed answerer search, while still enabling a node to accurately identify its friends that can answer a question. SOS uses the FOL representation and inference engine to derive the interests of questions, and interests of users based on user social information. A node considers both its friend's parsed interests and social closeness in determining the friend's similarity value, which measures both the capability and willingness of the friend to answer/forward a question. Compared to the centralized social network based Q&A systems that suffer from traffic congestions and high server bandwidth cost, SOS is a fully distributed system in which each node makes local decision on question forwarding. Compared to broadcasting, SOS generates much less overhead with its limited question forwardings. Since each user belongs to several social clusters, by locally selecting most potential answerers, the question is very likely to be forwarded to an answerer that can provide an answerer. The low computation cost makes the system suitable for low-end mobile devices. We conducted extensive trace-driven simulations and implemented the system on iPhone/iTouch mobile devices. The results show that SOS can accurately identifies answerers that are able to answer questions. Also, SOS earns high user satisfaction ratings on answering both factual and non-factual questions. In the future, we will release the application in the App Store and study the Q&A behaviors of users in a larger-scale social network.

## REFERENCES

[1] Google. http://www.google.com.
[2] Bing. http://www.bing.com.
[3] B. M. Evans and E. H. Chi. An elaborated model of social search. Information Processing & Management, 2009.
[4] L. Terveen, W. Hill, B. Amento, D. McDonald, and J. Creter. Phoaks: A system for sharing recommendations. Comm. of the ACM, 1997.
[5] M. S. Ackerman. Augmenting organizational memory: a field study of answer garden. ACM TOIS, 1998.
[6] L. G. Terveen, P. G. Selfridge, and M. D. Long. Living design memory: Framework, implementation, lessons learned. HumanComputer Interaction, 1995.
[7] E. Amitay, D. Carmel, N. Har'El, S. Ofek-Koifman, A. Soffer, S. Yogev, and N. Golbandi. Social search and discovery using a unified approach. In Proc. of HT, 2009.
[8] D. Carmel, N. Zwerdling, I. Guy, S. Ofek-Koifman, N. Har'el, I. Ronen, E. Uziel, S. Yogev, and S. Chernov. Personalized social search based on the user's social network. In Proc. of CIKM, 2009.
[9] S. Kolay and A. Dasdan. The value of socially tagged URLs for a search engine. In Proc. of WWW, 2009.
[10] S. Bao, G. Xue, X. Wu, Y. Yu, B. Fei, and Z. Su. Optimizing web search using social annotations. In Proc. of WWW, 2007.
[11] H. H. Chen, L. Gou, X. Zhang, and C. L. Giles. Collabseer: A search engine for collaboration discovery. In Proc. of JCDL, 2011.
[12] C. Y. Lin, N. Cao, S. X. Liu, S. Papadimitriou, J. Sun, and X. Yan. Smallblue: Social network analysis for expertise search and collective intelligence. In Proc. of ICDE, 2009.
[13] H. Kautz, B. Selman, and M. Shah. Referral web: combining social networks and collaborative filtering. Communications of the ACM, 1997.
[14] D. W. McDonald and M. S. Ackerman. Expertise recommender: a flexible recommendation system and architecture. In Proc. of CSCW, 2000.
[15] Yahoo answer. http://answers.yahoo.com.
[16] Ask.com. http://www.ask.com.
[17] F. Harper, D. Raban, S. Rafaeli, and J. Konstan. Predictors of answer quality in online Q&A sites. In Proc. of SIGCHI, 2008.
[18] M. R. Morris, J. Teevan, and K. Panovich. What do people ask their social networks, and why?: a survey study of status message q&a behavior. In Proc. of CHI, 2010.
[19] J. Teevan, M. R. Morris, and K. Panovich. Factors affecting response quantity, quality, and speed for questions asked via social network status messages. In Proc. of AAAI, 2011.
[20] R. W. White, M. Richardson, and Y. Liu. Effects of community size and contact rate in synchronous social q&a. 2011. [21] M. Richardson and R. W. White. Supporting synchronous social q&a throughout the question lifecycle. In Proc. of WWW, 2011.
[22] D. Horowitz and S. D. Kamvar. The anatomy of a large-scale social search engine. In Proc. of WWW, 2010.
[23] Twitter. http://www.twitter.com/.
[24] Mobile internet stats roundup. http://econsultancy.com/us/blog.
[25] Facebook may be growing too fast. http://techcrunch.com/.
[26] A. Mtibaa, M. May, C. Diot, and M. Ammar. Peoplerank: Social opportunistic forwarding. In Proc. of infocom, 2010. [27] R. M. Smullyan. First-order logic. Dover Publications, 1995.
[28] K. Toutanova and C. D. Manning. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In Proc. of SIGDAT, 2000.
[29] C. Manning and H. Schuetze. Foundations of Statistical Natural Language Processing. The MIT Press, June 18, 1999.
[30] M. Kirsten and S. Wrobel. Extending K-Means Clustering to First-Order Representations. In Proc. of ICILP, 2000.
[31] Z. Li, H. Shen, and K. Sapra. Leveraging Social Networks to Combat Collusion in Reputation Systems for Peer-to-Peer Networks. In Proc. of IPDPS, 2011