Measurement and Modeling of A Web-based Question Answering System

Chunyi Peng
Microsoft Research Asia
49 Zhichun Road, Haidian District
Beijing, P.R. China, 100080
Email: chunyip@microsoft.com

Zaoyang Gong
Department of Computer Science
University of Bei Hang
Beijing, P.R. China, 100083
Email: gongzy@nlsde.buaa.edu.cn

Guobin Shen
Microsoft Research Asia
49 Zhichun Road, Haidian District
Beijing, P.R. China, 100080
Email: jackysh@microsoft.com

***Abstract*— Question-Answering (QA) is one of the most natural and effective channels for specific information acquisition. QA systems emerged several decades back. In this paper, we perform a measurement study of a popular, large scale web-QA system, i.e. iAsk, and systematically investigate various behavior patterns and the system performance. To evaluate such a web-QA system, we propose three performance metrics, namely Reply-Rate, Reply-Number, and Reply-Latency, which are most closely related to the QoS of a web-QA system. Based on extensive measurement results, we propose a mathematical framework for the three performance metrics that capture our observation precisely. The framework reveals that the QoS of a web-QA system actually heavily depends on three key factors: the user scale, user reply probability and a system design artifact (related to webpage layout). We study their respective impacts on the system performance. Finally, we propose several ways through which current web-QA system can be improved. To the best of our knowledge, this is the first piece of work that studies the performances of web-QA systems.**

I. Introduction

People meet various kinds of questions in their daily lives, ranging from simple fact-based questions such as "what is the final result of the opening game of World Cup 2006?" to more sophisticated opinion questions such as "What do you think the movie Da Vinci's Code?". There are essentially two means for people to obtain the answer: one way is to find out the answer by themselves using various facilities such as libraries, search engines etc., and the other way is to ask others such as friends, colleagues, acquaintances, field experts etc.

Along the first way, it is now a common practice for people to resort to the web search engines to obtain an answer, thanks to the prevalence of the Internet and the great success of search engines, such as Google, Microsoft MSN etc., in the past few years. However, there are still a few hindering factors that make search not the optimal and ultimate solution to answer people's questions.

- Search engine typically returns pages of links, not direct answers. People still need to take significant extra efforts to find out the desired answer, mostly depending on how precise the search term(s) they input to the search engine can represent their questions.
- Some time it is very difficult for people to describe their questions in a precise way. This is especially the case when people have multimedia related questions. As a quick example, Lisa has a picture with a bird on it and she wants to learn more about this bird. How can she do it easily with current search engine?
- Although the content on the Web is overwhelming, not all information is readily available in the web. It is reported that most people actually do not like, if not impossible, to put down everything about their knowledge, skills, experiences, feelings, and thoughts.

On the other hand, it is very natural for people to ask around when they have questions. This has been actually well recognized long time back. For nearly half a century, researchers and engineers have explored and developed many question and answering (QA) systems. Early QA systems, e.g. 1960s' Intelligent Question-Answering Systems by Coles et.al [1], focused on how to kill the semantic ambiguity of questions using artificial intelligence (AI), and evolved to expertise systems [2].

Many groups working on question answering have followed the AI road, enhancing their works by progresses in natural language processing, information retrieval, information extraction and machine learning, and built QA systems to retrieve "answers" to questions rather than full documents or even best-matching passages as most information retrieval systems currently do [4][6]. As an example, MSN messenger has good Q&A support, using a server-side robot called Encarta®Instant Answers.[1] The answer provided is obtained through automatic background searches over the big commercial encyclopedia, Encarta®, that Microsoft possesses.

Recently, more researchers have resorted to the web as a resource for question answering, e.g. Mulder [5], AnswerBus [8], and built their QA systems on the top of the web search engine, packed with a smart and natural language input and output. Some of them do sophisticated parsing of the query and the full-text of retrieved pages, which is far more complex and computation-intensive. However, the study of question characteristics statistics shows that what users care most or expect to know are largely beyond those fact- or knowledge-based questions. On the contrary, most

[1]Everyone can add it as a buddy by the email address *encarta@conversagent.com* and ask it any question afterwards just as if asking a buddy.

questions are communicative-specific, location-specific, time-specific, and are not handled well by today's search engines and AI technologies. The difficulties seem to remain even with future relevancy, spelling or usability fixes. So the kind of AI-based QA systems is still far from being practical and reliable solutions.

Yet, another trend of QA systems, other than those mentioned above that try to build *automatic* QA systems, intends to explicitly utilize human intelligence, i.e., to involve human beings to directly provide answers. Early representative systems are various NewsGroups, Bulletin Board System(BBS).

Given the unsatisfactory performance of search engines, traditional QA systems and newsgroups for general question and answering, and motivated by other successful examples of utilizing the power of grassroots of Internet users, several new web-based, integrated[2] QA systems have emerged in past few years. Examples include Google Answers [9], Wondir [10], Naver [11], Sina's iAsk [12], etc, which provide free, publicly available live question and answering engines. Thanks to their user-friendly web-based interfaces and their quality of service, they have attracted millions of users and provided effective answers to millions of questions shortly after the debuts of these services. Clearly, as compared with the former AI-based QA system, today's web QA systems avoid the machine understanding problem through grassroots' intelligence and collaboration. However, it induces a new problem: how to find the proper persons for specific questions and stimulate them to help?

In this paper, we present one of the first measurement study of a large-scale web QA system, i.e., Sina's iAsk. We propose and perform a systematic study of several performance metrics, and reveal the driving force behind its success, using the real data set we collected from iAsk for a period of two months in 2005-2006, covering a total of 220 thousand of users. More specifically, we conduct measurement on the question/reply patterns over time, topic categories, users, and also the incentive mechanisms. Several interesting findings are observed include the Zipf-distribution topic popularity, narrow user interest scope, topic-relative user behavior, asymmetric question/reply patterns among users (altruists and free-riders), the negative impact of non-active web access on the reply performance, and the hardly effective incentive mechanism. We further propose a mathematical framework for the three performance metrics that capture our observation precisely. The framework reveals that the QoS of a web-QA system actually heavily depends on three key factors: the user scale, user reply probability and a system design artifact (related to webpage layout). Finally, we suggest various means to improve the current web-QA system's performance, including the active question notification and delivery, better incentive mechanism, better webpage layout design, and utilizing power of social networks, etc.

The rest of the paper is organized as follows: we first give a brief introduction to iAsk, the web-QA system under investigation, in Section II, and propose several performance metrics in Section III. We then present detailed measurement results to understand various behavior patterns across time, topics, users and rewarding scores in Section IV. We propose a mathematical framework for modeling the QA system in Section V, and evaluates the iAsk performance and analyzes some possible improvement to the current system in Section VI. Section VII gives more discussions on iAsk and other QA system, followed by our conclusions and future work in Section VIII.

[2]By integrated, we mean that these web-QA systems can accommodate almost any kinds of questions, which is in sharp comparison against newsgroups which is generally very specialized in certain topics.

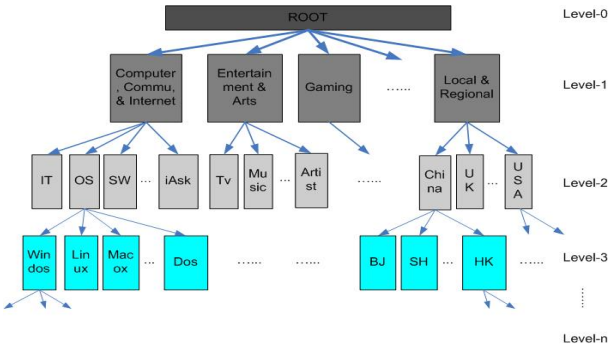


Fig. 1. Topic category tree at iAsk

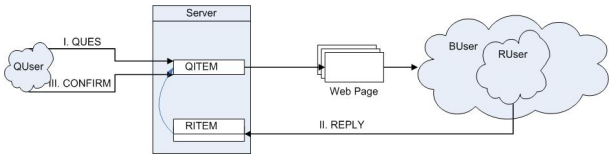


Fig. 2. Question-answering procedure in a QA system.

II. The iAsk Web-QA System

Sina's iAsk is one of the most popular web-QA systems in China, with more than about 6,000 new questions and 30,000 replies posted every day. Any registered user can ask any type of questions rather freely, and/or can contribute by replying any open question if she is willing to.

The iAsk is a topic-based web-QA system, where any question must be assigned with one and only one topic category when raised. All topic categories are pre-defined in the QA system, and cover more than 45 general topics and nearly 2,000 branch topics during our tracking period. All topics are organized into a hierarchy, as partially illustrated in Figure 1. Questions in the same topic are automatically organized into a question list (in one or multiple pages) in descending order according to their arrival time. The access path (e.g., hyperlinks) from the iAsk's homepage are also pre-defined, usually along the top-down structure of the topic tree. Users typically need multiple clicks to navigate to a specific sub-category.

The question-answering procedure at iAsk typically involves three steps, namely *questioning*, *replying* and *confirmation*, as illustrated in Figure 2. At the beginning, a user (called *QUser* hereafter) posts a question via the standard web interface, i.e., by filling out blanks about certain necessary information such

as question title, detailed description, related topic category, and even a rewarding score. The rewarding score is the virtual points that the QUser will pay the replier who comes up with the best answer. It is an incentive mechanism since users will be ranked according to their earned virtual points.

After the questioning stage, the question enters into the web-QA server with an "OPEN" status and appears in the related topic board as well as some guiding webpages such as the latest questions, the most difficulty questions, the highest rewarding questions, etc., depending how the question is specified. This organizing strategy provides multiple accesses to questions which in return increases the chances a question will be seen by a browsing users (called *BUser* hereafter).

In iAsk, questions reach users in a passive way. That is, only after a BUser requests the corresponding webpage, usually the latest question list in one topic, a series of questions will be seen by the browsing users. It is completely the browsing users' choices to reply or skip them. In our modeling, if a browsing user replies to a question, she will become a replying user (called *RUser* hereafter). Clearly, RUsers are just a small subset of BUsers, as illustrated in Figure 2. All replies will be automatically linked to the question and will be shown together with the question when browsed by later BUsers. Because of this passiveness, a user typically need to view many questions before she/he can locate and reply one.

Finally, the question confirmation procedure is designed to reset the question status. In the current design, a QUser is required to confirm the final status of her question within its lifetime period, e.g., 15 days at iAsk. One can close a question by selecting a best answer out of all the replies or none if no replies are satisfactory. Once a question is closed, it can not be replied any more. However, if the QUser wants to obtain better replies, she/he can re-open and, therefore, prolong the lifetime of the question by improving its rewarding score. If no confirmation process is explicitly performed by the QUser throughout the lifetime of a question, it will be automatically closed without an optimal answer or assigned with one optimal answer by administrators. A closed question will enters the server's database and serve as knowledge based for future queries.

III. Performance Metrics for QA Systems

Before we turn to the detailed measurement, we define three metrics for the QA performance evaluation, namely *Reply-Rate*, *Reply-Number*, and *Reply-Latency*, which we believe to be the most important for a QA system.

- *Reply-Rate.* What a user cares most about a QA system is that how likely his question can be answered. It usually reflects a QA system's credibility and is most crucial to the acceptance of the QA system. We define *Reply-Rate* to be the ratio of the number of questions that have been replied to the total question number. Clearly, the Reply-Rate reflects the QA system's service capability, and the optimal reply rate should be its maximal, i.e., 1.
- *Reply-Number.* The ultimate goal for a QUser is to obtain the correct answer to his question. However, in current web-QA system, there is no rule about the correctness of a reply. The QUser will accept the answer as long as he thinks the reply is satisfactory (not necessarily absolutely correct or complete). Therefore, we define *Reply-Number* to be the average number of replies for all questions. Intuitively, we believe that the more replies provided, the more likely the questioner will find an correct answer.
- *Reply-Latency.* Another factor a QUser cares is the how quickly he can get an answer, i.e., latency between the submission of a question and the notification of the correct answer. Even though most users can tolerate relatively longer latency, we do observe that there are many cases the user wants the answer as soon as possible, such as "Urgent! Wait online...". As pointed out earlier, the identification of the correct answer is up to the QUser, therefore, we only deal with the reply latency between the submission of a question and the notification of the *first* reply, no matter it is the correct one or not. Formally, *Reply-Latency* is defined as the average elapsed time between the submission and the first reply for all replied questions. Obviously, the smaller the reply latency is, the more rapid service a QA system provides.

IV. Measurements and Observations from iAsk

In iAsk, each questioner is assigned a unique user ID upon registration. Each question is also identified by a unique question ID. Our data set comes from all the questions submitted to the system from November 22, 2005 to January 23, 2006, covering a total of about 350,000 questions and 2,000,000 replies by about 220,000 users. For each question, we collect the question ID, questioner ID, question category, questioning time and the reward, as shown in Table I, together with all the corresponding reply information including replier IDs, replying times, and the flag of best answer etc, as shown in Table II. The statistics of our data set is summarized in Table III, where $N_u, N_{qu}, N_{ru}, N_q, N_r$ and N_c represent the total number of users, questioners, repliers, questions, replies and topics, respectively.

Before diving into the detailed analysis and modeling of the iAsk web-QA system's performance, we present some observations on user behavior patterns since they will lay the foundation of some of the assumptions we will use in the modeling phase.

A. Behavior Pattern over Time

We conduct measurements over the question arrival pattern on a weekly, daily, hourly and minutely basis. Due to the space limitation, we only present the results of daily and hourly measurements. In short, the weekly pattern is a low-pass filtered version over the daily pattern is therefore more stable, while the minutely pattern is more dynamic but reveals clear coarse trend as shaped by the hourly pattern.

- *Daily Pattern.* Figure 3 shows the daily number of questions, replies, users with different behaviors throughout our tracking period. We observe that these curves are relative stable at the daily rate about 6 thousand new

TABLE I

QUESTION SAMPLE INFORMATION

Ques	QUser	Category	QTime	Reward
3000930	1406626053	/4/226/230	2005-11-22 0:00:08	0
3000934	1070291357	/1/13/15	2005-11-22 0:01:44	10
3000935	1195577997	/2/161	2005-11-22 0:01:34	100
...	...	...	...	...

TABLE II

REPLY SAMPLE INFORMATION

Ques	RUser	RTime	IsBest
3000930	1085368163	2005-11-22 0:11:38	0
3000930	1454201925	2005-11-26 18:47:14	1
...	...	...	...

TABLE III

STATISTICS OF COLLECTED DATA SET

N_u	N_{qu}	N_{ru}	N_q	N_r	N_c
218,453	122,920	151,316	372,452	1,884,982	1,901

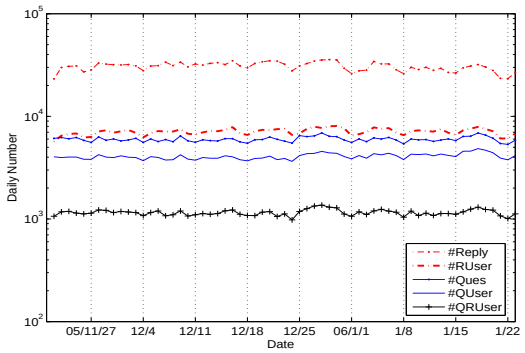


Fig. 3. Daily number of questions, replies, users with different operations across the entire tracking period of 63 days

questions, 30 thousand replies, 4 thousand questioners, 7 thousand repliers, and 1 thousand users that both ask and reply.

- *Hourly Pattern.* We take a closer look at the hourly question arrival pattern. Figure 4 shows a consistent repeated pattern cross all the days during the week from Jan. 1 to Jan. 7, 2006. The number of questions drops gradually during midnight and the early morning (23PM-7AM), and climbs up to a stable high rate stage (9AM-22PM), but drops a little at breaks 12AM-1PM and 18PM-19PM. This pattern reflects the user web access behavior: most users access the web-QA system during the work time and the leisure time (e.g., after supper but before sleep), with slight fluctuations at lunch and dinner time frames. This can be seen more clearly form Figure 5 where we aggregate and plot the question arriving pattern at hourly scale over the whole tracking period.

The above observations across different time scales provide us a big picture of user behaviors, including web-access, questioning and replying behavior in a large scale online web-QA system. It is clear that the user scale varies across

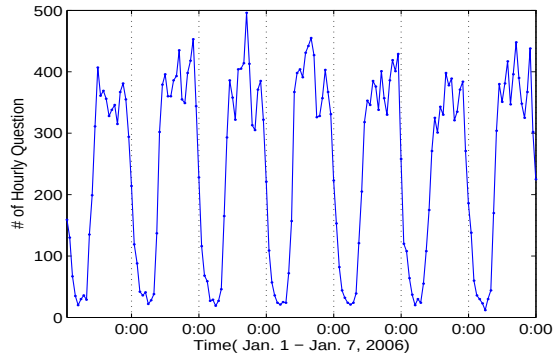


Fig. 4. Hourly distribution of question number during a single week from Jan.1 to Jan. 7, 2006.

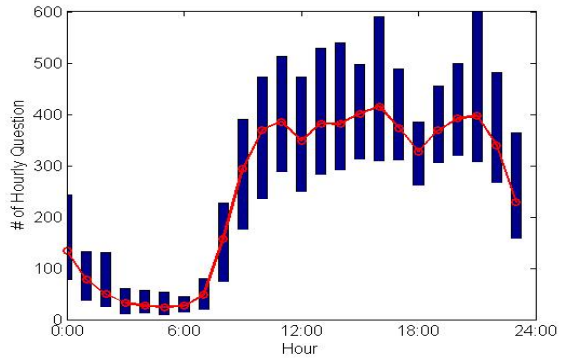


Fig. 5. Aggregated hourly distribution of average and max-min scope across 24 hours of a day throughout the tracking period.

time, especially on the hourly scale, and that the asking and replying operations are positively proportional to the actual user scale. In fact, one important observation from our measurement results is that the ratios of the question number to the reply number and to the user number stay nearly invariant at any time in any time scale, except a slight vibration within the specific period. Figure 6 gives all the ratios across the hourly scale, where each point represents the data in a certain hour. Therefore, in the rest of our study, we only focus on the question number measurement results to investigate QA statistics over different scales of time.

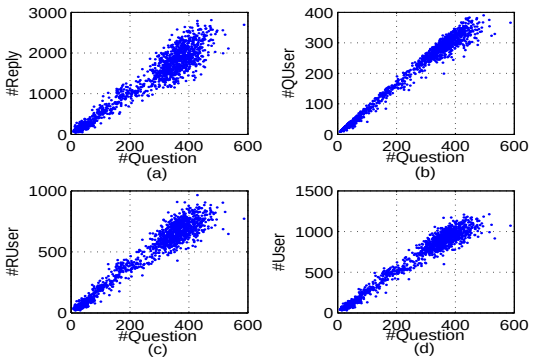


Fig. 6. Hourly reply, RUser, QUser and user number versus question number throughout the tracking period.

Since there is a clear deviation among the hourly question patterns, to better investigate the QA performance under different scales, we divide our data set into three subsets, $S_q(Lo)$, $S_q(Hi)$ and the rest, according to the instants that the questions arrive to the system. Here, $S_q(Lo)$ and $S_q(Hi)$ correspond to the questions that arrive during the low traffic interval (1AM-7AM) and the high traffic interval (9AM-21PM), respectively. The rest question samples can be regarded as a hybrid of these two cases.

B. Behavior Pattern over Topics

As described before, the iAsk is a topic-based QA system where each question must be classified into one and only one topic category. A question topic shows a QUser's interest, and similarly, the question distribution across all topic categories reflects the taste of a QA system.

There exists a hierarchical topic tree with 45 level-1 and total 1901 topics during our tracking period. Figure 7 plots the question number in the major level-1 topic categories, and indicates that most questions at iAsk are related to the following main topic categories: Gaming (20%), Computer, Communication & Internet (14%), Acting, Arts & Entertainment (12%), Family & Life (10%). It is clear that the scale across topics is non-uniform. Actually, the distribution of question number versus topic category is found to be a Zipf-distribution approximation with coefficient $\alpha = -0.82$, as shown in Figure 8 (a).

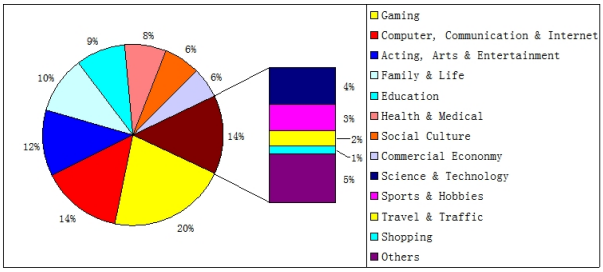


Fig. 7. Question distribution over topic category

Note that the characteristics of a specific topic plays a crucial role in QA performance. Intuitively, how likely a question can be replied depends on how likely it can be viewed by other users and how demanding to users' knowledge to answer the question. To provide more insights, we highlight three characteristics and study their impacts on the QA system's performance.

- *Popularity ($\mathcal{P}$).* A popular topic implies more questioning and replying activities and more participants. Therefore, we measure the topic popularity using the absolute number of questions, replies, QUsers and RUsers. Figure 8(a) plots the probability distribution function (PDF) of question number across topic, and it also validates the Pareto-distribution on the topic popularity. In fact, 80% questions are generated in 10% (nearly 200) topics. Figure 8 (b), (c) and (d), respectively, shows all topic samples of the question number versus the number of reply, QUser and RUser, and illustrates they are basically consistent in representing a topic popularity.
- *Question Proneness ($\mathcal{Q}$).* This characteristic represents the likelihood that a user will ask a question. It is defined as the ratio of number of questions to that of all users (including Q/RUsers, QUsers, RUsers and browsing users) in a specific topic. It generally reflects users interest in a certain topic since the more a user cares about a topic, the more likely he will ask questions.
- *Reply Proneness ($\mathcal{R}$).* Similarly, reply proneness describes the likelihood that a user will reply a question. It is defined as the ratio of the number of replies to that of all users in a specific topic. This characteristic reflects users expertise and altruism in a certain topic. More capable and altruistic participants appears in a topic with high $\mathcal{R}$ to reply others' questions.

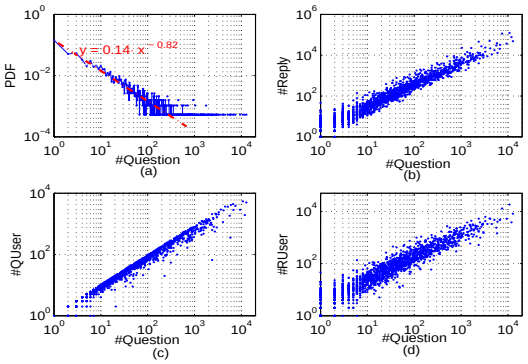


Fig. 8. Topic distribution, reply, QUser and RUser number versus question number over all the topics.

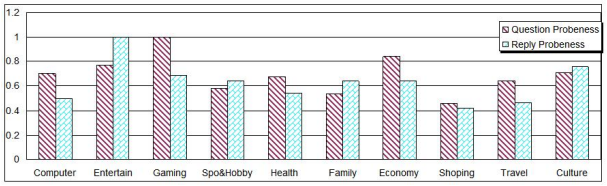


Fig. 9. Topic characteristics over major level-1 topic categories.

Figure 9 illustrates the Question Proneness and Reply Proneness across the major level-1 topic categories, calculated by averaging over all sub-topics. Note that, for better visual effect, each points is normalized by the maximal average value across all level-1 topics. It is observed from Figure 9 that Entertainment is a popular topic where many participants are enthusiastic in replying and the bar for replying is relatively lower. For Gaming, players are very interested in learning some skills or tricks to improve faster and they are more likely to ask questions. All these observations match well with our empirical experiences. In short, characteristics of different topics imply quite different user profiles and therefore play an important rule in our measurement and analysis.

C. Behavior Pattern across Users

As aforementioned, the web-QA system differs from search engines and expertise systems in that it explicitly leverages

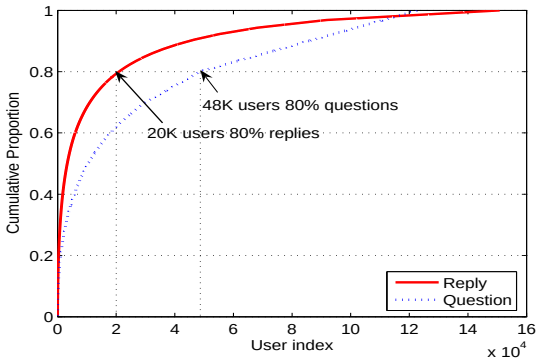


Fig. 10. Cumulative question/reply proportion by users ranked by the question/reply number.

human intelligence to reply questions. Therefore, user behavior have a profound impact on the QA system's performance. A good understanding of user behavior will provide meaningful hints for improving the web-QA system.

Figure 10 plots the cumulative distribution function (CDF) of question (reply) number by those QUsers (RUsers) in a descending order of their question (reply) number. From the figure, we observe an asymmetric questioning and replying pattern where about 9% users (about 20K, 13% of all RUsers) have contributed 80% replies and 80% questions are asked by 22% users (about 48K, 40% of all QUsers). Define the active replier set (aRU) as the set of repliers that contributed to the 80%-reply, and the active questioner set (aQU) as the set for those who contributed to the 80%-question. Then we can classify users into four types: $aRU \cap aQU$ (active in both), $aRU \cap \overline{aQU}$ (active in reply not questioning), $\overline{aRU} \cap aQU$ (active in questioning not reply) and $\overline{aRU} \cap \overline{aQU}$ (inactive in both). Table IV lists the questioning and reply statistics by the above four types of users. Note that there exists an even stronger asymmetry between questioning and replying patterns among active users: we can see a small scale of altruists (4.7%) and a larger scale of free-riders (about 17.7%). Besides, it should be noticed that nearly 73% users performs inactively, as if visitors just occasionally ask or reply questions if they need or will.

TABLE IV

STATISTICS OF QUESTIONS AND REPLIES BY FOUR TYPES OF USERS.

User Set	N_u	N_q	N_r
$aRU \cap aQU$	10,085 (4.6%)	118,000 (31.7%)	893,283 (47.5%)
$aRU \cap \overline{aQU}$	10,333 (4.7%)	***3,248 (0.9%)***	***602,836 (32.1%)***
$\overline{aRU} \cap aQU$	38,659 (17.7%)	***180,276 (48.4%)***	***92,487 (4.9%)***
$\overline{aRU} \cap \overline{aQU}$	159,376 (73.0%)	70,928 (19.0%)	290,509 (15.5%)

Interests, i.e. topics one really cares, is another kind of important user characteristics. Figure 11 shows the PDF of the topic number of QUsers and RUsers. The average topic numbers with questioning or replying behavior are 1.8 and 3.3, respectively. This implies that a user usually cares about a narrow scope of topic categories, and contribute (in both asking and replying) to a small portion of the QA system. This is an interesting finding. It suggests that it is possible that we push the questions to potential repliers without overwhelming them. We will conduct more detailed discussion on this in Section VII.

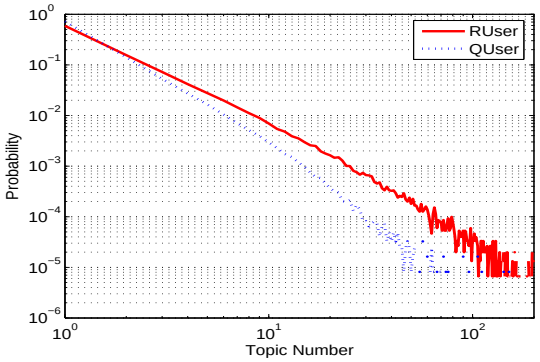


Fig. 11. PDF of user number across interest topic number at iAsk.

D. Behavior Pattern under Incentive Mechanism

We mentioned that iAsk has an incentive mechanism by rewarding the replier who provides the best answer in Section II. As with any incentive mechanism, it is designed to stimulate users to pay more attention and make more rapid responses. Now we present our measurement and finds of user behavior patterns under the incentive mechanism.

First of all, the rewarding score is ranged from 0 to 100 at iAsk. Note that users are allowed to ask questions with zero reward, and this is actually the system default value, therefore any user can ask question. This accounts for why there are so many free-riders in the system.

Now let us examine the question distribution versus rewarding scores, as illustrated by the proportion bar in Figure 12. We find that most questions (about 273K, 73.5%) are scored with zero, and the rest questions are mostly rewarded with scores that are multiples of 5, e.g. $5, 10, 15, 20, ...100$. We plot the curves for Reply-Number and Reply-Latency for the major scored questions (excluding the scored questions with the sample size less than 200) in Figure 12. It is observed that, from the Reply-Number curve, there is only very slight improvement for questions with rewarding scores from 5 to 90. However, the performance for the highest-scored (i.e., 100) questions has been greatly improved with Reply-Latency decreasing to 3 from 10 hours, the average latency of replies for questions rewarded less than 90. Similar observations can be made from the curve for Reply-Number. Therefore, we think the incentive mechanism of iAsk does not work out well. We will provide more discussions on this in Section VII.

E. Overall Performance

Figure 13 shows the performance measurement of Reply-Rate, Reply-Number and Reply-Latency of four sample sets: all topics during a whole day, one popular-topic in $S_q(Hi)$ and $S_q(Lo)$ and one unpopular topic in $S_q(Hi)$.[3]

[3]We do not list the results of the unpopular topic for $S_q(Lo)$ because question samples are too small.

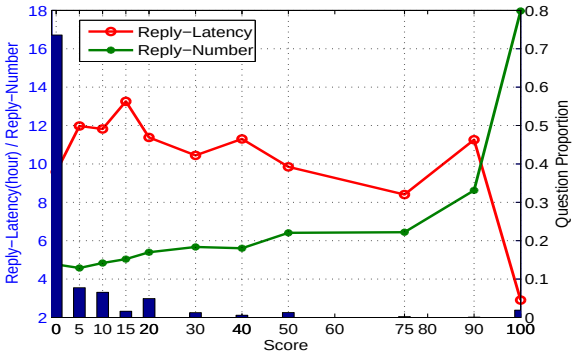


Fig. 12. *Reply-Number* and *Reply-Latency* versus question scores.

From the figure, we see that all the four curves for all the three metrics gradually rise up and become leveled off. Let us brief their stable performance first: the iAsk system provides replies to 99.8% of its questions, about five replies per question, and the average latency is about 10 hours. Within 24 hours, Reply-Rate has arrived at 85%, Reply-Number at 4 (80% of the final one), and Reply-Latency at about 6 hours (60% of the final one). This shows the performance is quite satisfactory except sometimes users need tolerate a relative longer wait, which implies the QoS is not good enough for an urgent request.

The final performance for Reply-Rate and Reply-Number for questions in one topic are nearly the same no matter whether it is raised during $S_q(Hi)$ or $S_q(Lo)$. But for Reply-Latency, questions in $S_q(Lo)$ need to wait a longer time to get feedback because they have a small user scale when raised. The dynamic performance (i.e., the accumulative performance since their arrival) of questions raised in $S_q(Hi)$ or $S_q(Lo)$ are quite different: questions in $S_q(Hi)$ enjoy much better higher Reply-Rate, larger Reply-Number, and shorter Reply-Latency than those raised in $S_q(Lo)$. A popular topic typically enjoys better performance than the unpopular one, as one may expected.

V. Mathematical Models

In this section, we introduce our mathematical framework for QA system based on previous measurement results. As mentioned before, the iAsk is a topic-based QA system. Therefore, we focus on the question and answering framework on *per-topic basis*. Nevertheless, our model is quite flexible to be applied to different topics since the characteristics of different topics are essentially captured by the three main parameters, namely $\mathcal{P}$, $\mathcal{Q}$, $\mathcal{R}$, as stated before. For sake of concise presentation and quick reference, we list all the notations in Table V.

A. Mathematical Definition of Performance Metrics

We have stated three performance metrics of a web-QA system literally in Section III. Now, let us define them mathematically (with symbols defined in Table V) so as to perform quantitative evaluation later on.

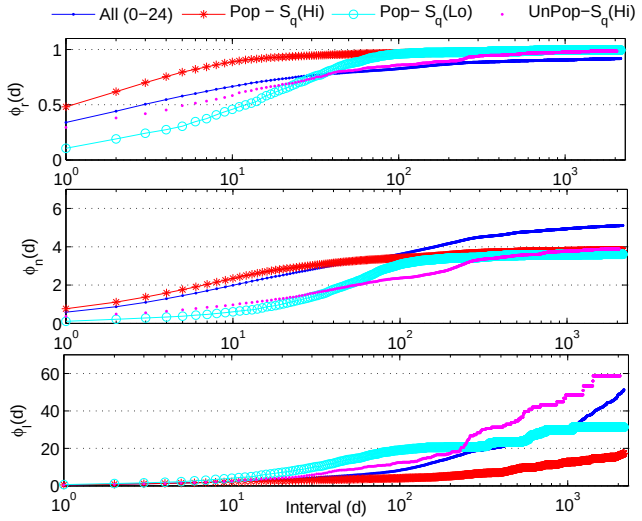


Fig. 13. Performance measurement of different samples: all questions in all topics (whole day), $S_q(Hi)$ and $S_q(Lo)$ in a popular topic and $S_q(Hi)$ in an unpopular topic.

- *Reply-Rate*

$$\phi_r(d) = \frac{1}{N_q} \sum_{q_k \in Q} (N_r(q_k, d) >= 1). \quad (1)$$

Since replying is a dynamic procedure, $N_r(q_k, d)$ actually is a mono-increasing function across interval d, and $\phi_r(d)$ can usually be counted at $d = \infty$ or some specific time point. Let ϕ_r be short for $\phi_r(\infty)$ (similar for the other two metrics).

- *Reply-Number*

$$\phi_n(d) = \frac{1}{N_q} \sum_{q_k \in Q} N_r(q_k, d). \quad (2)$$

- *Reply-Latency*

$$\phi_l(d) = \frac{\sum_{q_k \in Q} \min_{r_l \in R_{q_k}(d)} (\Gamma(r_l) - \Gamma(q_k))}{\sum_{q_k \in Q} (N_r(q_k, d) >= 1)}. \quad (3)$$

B. Questioning Behavior Pattern

Each online user can ask questions in his/her topic category. A questioning model mainly describes when or how often a user may ask a question and what the question looks like. We define a probability function $P_{u,q}$ as the questioning probability of user u, which may be related to topic category (e.g. Hobby-Index), user capability (influencing one's questioning requirement), the questioning history (e.g. the asked question number) or something else. For simplicity, assume that the questioning probability is homogeneous for any user in one single topic, denoted by p_q.

Each online user can ask questions in any topic category. The questioning pattern mainly describes when or how often a user may ask a question and what the question looks like. Let the questioning probability for a specific topic c as $p_q = \frac{1}{N_u(c)} \sum_u p_{q,u}$, where $p_{q,u}$ is the questioning probability of a user u, and $N_u(c)$ is user number of topic c. Since the

TABLE V

DENOTATIONS OF TERMS USED IN OUR MATHEMATICAL FORMULATION

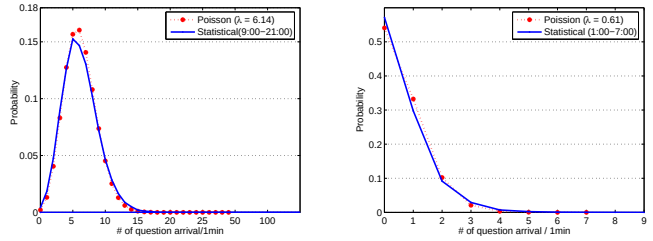
N_u, N_q	the number of users and questions, respectively, for a topic.
$C(\cdot)$	the topic mapping function, e.g. $C(q_k)$ is q_k's topic category.
$\Gamma(\cdot)$	the item's access time point, e.g. $\Gamma(q_k)$ and $\Gamma(r)$ are posting time of question q_k and reply r, resp.
$R_q(d)$	the set of all replies to question q within d intervals since $\Gamma(q)$,
$N_r(q,d)$	the reply number up to d intervals,
$\Delta N_r(q,d)$	the reply number at the d-th interval.
$p_{q,u}$	the questioning probability of user u per interval.
$p_{r,u,q}(d)$	the user u's replying probability to question q after d intervals since $\Gamma(q)$.

questioning probability should be closely related to its topic's Question Proneness, we define a simple linear model with

$$p_q = p_{q0} \cdot \mathcal{Q}(C(q)), \tag{4}$$

where p_{q0} denotes a general questioning probability of the whole QA system.

The question arrival distribution is modeled as a λp_q-Poisson distribution under the general assumption of Poisson-distribution (with λ) for web-access model. Figure 14 compares the statistical distribution of the number of all arriving questions per one minute interval and their Poisson distribution correspondents for all $S_q(Hi)$ and $S_q(Lo)$ in the whole iAsk. The measurement validates the assumption of Poisson-distribution for question arrival. The distribution model holds true for each individual topic category, for which we give the parameters for some major Level-1 topic categories in Table VI.



(a) in 9:00-21PM (b) in 1:00-7AM

Fig. 14. Question arrival per one minute interval distribution. (a) and (b), respectively, describe the arrival distribution for high-arrival-rate question samples $S_q(Hi)$ and low-arrival-rate question samples $S_q(Lo)$.

TABLE VI

QUESTION ARRIVAL RATES FOR DIFFERENT TOPIC CATEGORIES.

Time	All	Game	Comp	Enter	Family	···
9PM-21PM	6.14	1.188	0.852	0.738	0.672	···
1AM-7AM	0.61	0.188	0.076	0.086	0.052	···

C. Reply Behavior Pattern

The replying behavior pattern is closely related to user profile and topic characteristics. For example, it is more likely for an altruist to take more efforts to reply questions; one will be more willing to reply if the question is easy. We expect to model the reply behavior through iAsk's measurements and explain when a user u_j will reply one question q_i. Through extensive reply measurements, we build a simple reply probability model over time, and find it matches the measurement very well.

Let $p_r(t)$ denotes the reply probability to a question at the t-th intervals, which balances all users replying behavior to any question in a certain topic, we have

$$p_r(t) = \frac{1}{N_u N q} \sum_u \sum_q p_{u,q,r}(t). \tag{5}$$

Let $N_u(d)$ denote the user number at the d-th interval since the posting time of question q, $\Gamma(q)$, then the reply number for question q at the d-th interval can be calculated as

$$\begin{aligned} P(\Delta N_r(q,d) = k) &= C^k_{N_u(d)} \cdot p_r(d)^k (1 - p_r(d))^{N_u(d)-k} \\ &= \mathbb{B}(N_u(d), k, p_r(d)) \end{aligned} \tag{6}$$

where $\mathbb{B}(N,k,p) = C^k_N \cdot p^k (1-p)^{N-k}$ represents the k-hit of N-trial in a p-Bernoulli process. For all replies to question q, the probability distribution function at the d-th interval can be computed as

$$\begin{aligned} P_r(\Gamma(r) = d) &= \frac{\sum_k k \cdot P(\Delta N_r(q,d) = k)}{\sum_{t=1}^{\infty} \sum_k k \cdot P(\Delta N_r(q,d) = k)} \\ &= \frac{N_u(d) \cdot p_r(d)}{\sum_{t=1}^{\infty} N_u(t) \cdot p_r(t)}. \end{aligned} \tag{7}$$

In addition, it is observed from the above measurement that the user scale keeps stable for a long period, and thus the above reply distribution over time can be reduced to

$$P_r(\Gamma(r) = d) \sim p_r(d), \tag{8}$$

that is, the reply distribution probability function at time d is approximately isomorphic to the reply probability function at interval d. Figure 15 gives the three example reply distribution for high-arrival-rate questions, based on about 40K, 60K, 40K replies, where topic A, B, C is listed in Table VII. In the figure, each interval unit is 10 minutes and keeps the same in other figures unless otherwise noted. Plenty of measurement illustrates the reply distribution shape for high-arrival-rate questions nearly keeps the same in any topics.

Figure 15 implies that the distribution of early replies (within 600 minutes) can be well captured with an exponentially-decaying function, followed by a relative small but stable tail. It is easy to understand that the reply probability attenuates over time because according to the web layout, the newly generated question is put on the top of the access page which lowers down old questions or even expels them out

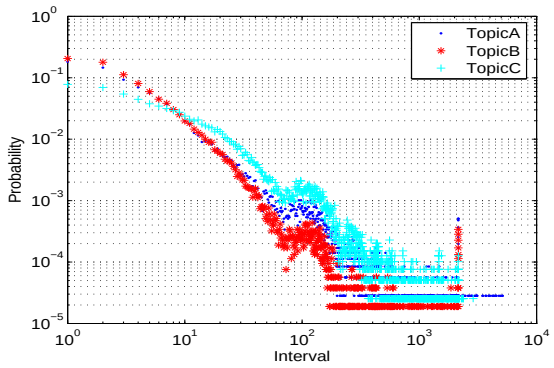


Fig. 15. Example of reply distribution of high-arrival-rate question samples in multiple topics.

of current page. Another reason is that there may have been some replies after a while and it becomes less necessary or attractive for BUsers to reply. Notice that the relative stable tail is generated by the random browsing behavior for old questions. Moreover, there seems to be no big difference in reply pattern for questions that are old enough, i.e., no matter it is three days or a week old. However, we do observe a sudden burst at the end of a question lifetime (the 15-th day or the 2160-th interval). This results from the additional access page specially provided by iAsk for those nearly overdue questions. Since this tail only contributes a very small amount of replies from our measurement (e.g., 10^{-4} after 100 intervals in Figure 15), we simply neglect it and model the reply behavior with the exponentially-decaying model,

$$p_r(d) = p_r \cdot e^{-d/\tau}, \tag{9}$$

where p_r denotes the initial reply probability in this topic, and τ denotes the attenuation factor over time interval. Assume that the replying probability is linearly related to a topic's Reply Proneness, that is, $p_r = p_{r0} \cdot \mathcal{R}(C(q))$, where p_{r0} denotes a general replying probability of the whole QA system, and τ is dependent on the question update rate at the BUser side, i.e., an artifact of the system design such as the webpage layout and the question number in one page, and expressed as $\tau = \frac{\alpha}{\lambda_q}$, where α is a system design constant.

VI. Model Evaluation

In the section, we evaluate the above mathematical models using the actual measurements results of QA performance, and explore the key impact factors to the system's performance.

From the definition, Reply-Rate up to t intervals can be computed by

$$\begin{aligned} \phi_r(t) &= 1 - \prod_{d=1}^{t} P(\Delta N_r(q,d) = 0) \\ &= 1 - \prod_{d=1}^{t} (1 - p_r e^{-d/\tau})^{\lambda(d)}, \end{aligned} \tag{10}$$

where $\lambda(d)$ denotes the user arrival rate at interval d. Note that for questions with high arrival rate, the user arrival rate $\lambda(d)$ may keep stable for a long period. Moreover, $p_r(d)$ is exponentially-decaying with time d (Eqn (9)) and plays little impact on $\phi_r(t)$ when $\lambda(d)$ begins to fluctuate after a long time. So $\phi_r(t)$ can be approximated as

$$\begin{aligned} \phi_r(t) &\simeq 1 - \prod_{d=1}^{t} (1 - p_r e^{-d/\tau})^{\lambda} \\ &\approx 1 - (1 - p_r \sum_{d=1}^{t} e^{-d/\tau})^{\lambda} \text{ (for } p_r \text{ small)} \\ &= 1 - (1 - p_r \frac{e^{-1/\tau} - e^{-(t+1)/\tau}}{1 - e^{-1/\tau}})^{\lambda} \end{aligned} \tag{11}$$

If we measure the system at a stable stage, i.e., $t \to \infty$, τ and λ are large enough (e.g., for a popular topic), using $\frac{e^{-1/\tau}}{1-e^{-1/\tau}} \approx \tau$ and $(1 - c/x)^x = e^{-c}$, Eqn (11) can be simplified as

$$\phi_r = \phi_r(\infty) \approx 1 - e^{-\lambda p_r \tau}. \tag{12}$$

We can find that ϕ_r is monotonously increasing with the product of λ and p_r and τ. That is, more users, higher reply probability, and a slower question update rate (e.g., questions stay longer at the top page) can effectively improve the QA system's Reply-Rate.

Now let us investigate another performance metric, Reply-Number. The average Reply-Number can be expressed as:

$$\begin{aligned} \phi_n(t) &= \sum_{d=1}^{t} p_r(d) \cdot \lambda(d) \\ &\simeq \lambda p_r \cdot \frac{e^{-1/\tau}(1 - e^{-t/\tau})}{1 - e^{-1/\tau}} \\ &\approx \lambda p_r \tau \quad (t \gg \tau \text{ and } \tau \text{ large}) \end{aligned} \tag{13}$$

From the measurement, we found τ is generally greater than 5 for very popular topics and can go beyond 20 for less popular topics (see Table VII). It is now very interesting to find that Reply-Rate and Reply-Number are actually closely related in the above reply framework when λ, τ and p_r are appropriate.

$$\phi_r = 1 - e^{-\phi_n} \tag{14}$$

Furthermore, Figure 16 plots all the actual Reply-Rate measurement, in an ascending order of Reply-Number across all topics, using the reply samples within 40 intervals (approx. 7 hours) for high arrival rate questions, and the theoretical curve for Reply-Rate that is calculated from Eqn (14) with Reply-Number ϕ_n calculated via Eqn (13). The figure clearly demonstrates the good fit of our model to the measurements.

A close look at Figure 16 reveals that iAsk has slow responses to a significant portion of questions. Within 40 intervals, the non-reply proportion reaches more than 20% (i.e., their Reply-Rate is less than 80%, as indicated by the two dash-dotted lines in the figure) for more than half of all topics. Actually, the iAsk's Reply-Latency is measured about 10 hours for all questions, 9.5 hours for high-arrival-rate questions at 9AM-21PM, and 12 hours for low-arrival-rate questions at 1AM-7AM. Users may get disappointed if he expects an quick answer from Today's iAsk. So it is worth

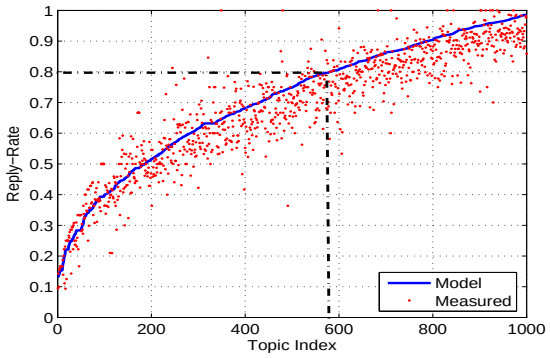


Fig. 16. Comparison of measured and modeled Reply-Rate over topics.

examining the Reply-Latency more thoroughly to identify the key factors. The probability of the first delay at the d-th interval can be written as

$$p_r(fd = d) = (1 - p_{r,d}(0)) \cdot \prod_{k=1}^{d-1} p_{r,k}(0). \quad (15)$$

Hence, the Reply-Latency can be expressed as

$$\begin{aligned}\phi_l(t) &= \sum_{d=1}^{t} d \cdot p_r(fd = d) \\ &= \sum_{d=1}^{t} d(1 - (1 - p_r e^{-d/\tau})^{\lambda}) \prod_{k=1}^{d-1} (1 - p_r e^{-k/\tau})^{\lambda} \quad (16)\end{aligned}$$

When d is large enough, $1 - (1 - p_r e^{-d/\tau})^{\lambda}$ can be reduced to $\lambda p_r e^{-d/\tau}$ via Taylor approximation, and $\prod_{i=1}^{d-1} (1 - p_r e^{-i/\tau})^{\lambda} < 1$, so $\phi_l(t)$ can be upperbounded by $const * \sum d e^{-d}$ and be convergent. In fact, after some mathematical manipulation, ϕ_∞ can be estimated by

$$\phi_l \approx 1 + e^{-\lambda p_r e^{-\frac{1}{\tau}}} + e^{-\lambda p_r \tau} \sum_{d=3}^{\infty} (e^{\lambda p_r \frac{e^{-d/\tau}}{1 - e^{-1/\tau}}} - 1). \quad (17)$$

Although we can not obtain the analytical equation of ϕ_l with respect to λ, p_r and τ, we can still infer that Reply-Latency is mono-decreasing with λ, p_r, and gradually smooths down as τ increases. It becomes insensitive to τ when λp_r is large enough, e.g., any $\tau > 10$ has also no impacts on ϕ_l when λp_r = 1.

Now we start to validate these models using real measurement data. Table VII lists the statistical topic characteristics of the top 10 hot topics using all replying samples during the whole tracking period. We derive their model parameters (assume the QA system parameters $p_{q0} = 0.02$, $p_{r0} = 0.002$ and $\alpha = 10$) for the early reply behavior for high arrival rate samples and compare the theoretical performance results against the actual measurements within 100 intervals in Table VII. Figure 17 shows more detailed comparisons between the measured and modeled performance metrics for topic A, B and C. It is found that our model matches well with the performance observations in hot topics, and indeed captures precisely the early reply behavior distribution. However, we do locate (as bold-faced numbers in the table) few cases where our model overestimates the latency performance since after a long time. The reason is due to small but long replying tail, as we have explained previously, which, when accumulated, leads to the increase of ϕ_l.

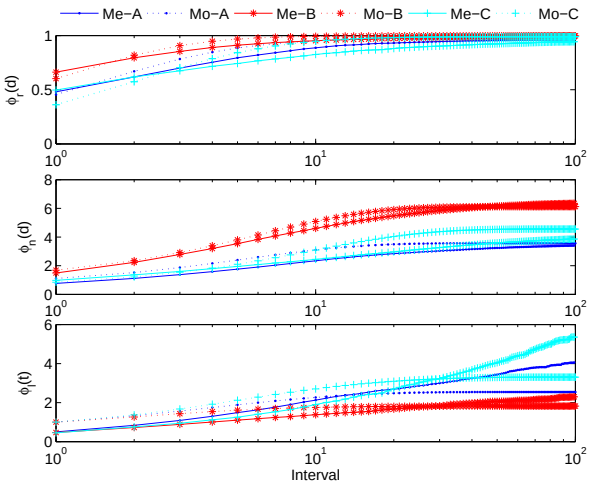


Fig. 17. Comparison of measured and modeled dynamic performance in three typical topics, where Me-* denotes the measured results and Mo-* denotes the modeled ones.

The above performance analysis investigates the impacts of question arrival rate (λ), initial questioning probability (p_r, and system design constant (τ) on the final performance of a web-QA system, see Eqn (12), (13) and (17). However, what users concern more in many situations is the achieved performance within a fixed period, namely, its dynamic performance. To evaluate the dynamic performance of a web-QA system, we define the accumulative performance metric for Reply-Rate and Reply-Number as the proportion of accumulated performance (over time, and up to d intervals) to the final stable performance.[4]

$$\varphi_r(d) = \phi_r(d)/\phi_r, \quad (18)$$
$$\varphi_n(d) = \phi_n(d)/\phi_n. \quad (19)$$

The higher proportion means a better dynamic performance. We can simplify Equ. (18) and (19) in case of large λp_r as below,

$$\frac{1 - \phi_r(t)}{1 - \phi_r} = e^{\lambda p_r \frac{e^{-t/\tau}}{1 - e^{-1/\tau}}} \Rightarrow$$
$$\varphi_r(d) = \frac{1 - e^{\lambda p_r \frac{e^{-t/\tau}}{1 - e^{-1/\tau}}}}{\phi_r} + e^{\lambda p_r \frac{e^{-t/\tau}}{1 - e^{-1/\tau}}} \quad (20)$$
$$\varphi_n(d) = 1 - e^{-t/\tau} \quad (21)$$

Table VIII shows the all performance impact of the parameters of λ, p_r and τ, where the symbol $\uparrow$ represents a positive relation, $\downarrow$ for a negative relation, and — for no relations. It is clear that the increasing λ and p_r can help improve all the final performance and have even more significant impact

[4]It makes no sense to define a similar one for Reply-Latency.

TABLE VII

STATISTICAL & ANALYTICAL RESULTS OF TOPIC CHARACTERISTICS, MODEL PARAMETERS AND REPLY PERFORMANCE FOR $S_q(Hi)$ IN HOT TOPICS.

Category	$\mathcal{P}$	$\mathcal{Q}$	$\mathcal{R}$	λ	τ	p_r	Model-ϕ_s	ϕ_s	Model-ϕ_b	ϕ_b	Model-ϕ_l	ϕ_l
Gaming/Warcraft (**A**)	9,278	1.02	3.90	91	5.38	0.008	0.972	0.966	3.56	3.41	**2.54**	**4.06**
Entertain/Lottery (**B**)	8,061	2.20	14.44	37	6.19	0.029	0.998	0.991	6.12	6.35	1.82	2.30
Family/Love	7,981	0.46	5.28	174	6.25	0.011	1.000	0.990	11.03	10.08	1.27	1.90
Gaming/CrossGate	6,938	1.31	5.23	106	3.60	0.011	0.981	0.985	3.91	3.86	2.96	1.94
Health/Sex	5,509	0.43	6.67	127	9.06	0.013	1.000	0.99	14.15	13.00	1.31	1.81
Education/QualificationTest (**C**)	5,146	0.93	4.37	55	9.70	0.009	0.989	0.943	4.56	3.86	**3.31**	**5.36**
Computer/Software	4,923	0.72	2.25	68	10.14	0.005	0.963	0.922	3.28	2.76	4.58	4.74
Commercial/Stock	4,518	1.49	7.99	30	11.04	0.016	0.994	0.965	5.06	4.78	3.32	4.38
Entertain/Riddle	4,309	3.07	20.23	14	11.58	0.041	0.998	0.992	6.36	5.83	2.74	2.11
Computer/Internet	3,465	0.65	1.57	59	11.79	0.003	0.865	0.849	1.9992	1.8056	**7.3026**	**3.7784**

TABLE VIII

PARAMETER IMPACTS ON QA PERFORMANCE

metric	λ	p_r	τ
ϕ_r	$\uparrow$	$\uparrow$	$\uparrow$
ϕ_n	$\uparrow$	$\uparrow$	$\uparrow$
ϕ_l (small λp_r)	$\downarrow$	$\downarrow$	$\downarrow$ [a]
ϕ_l (large λp_r)	$\downarrow$	$\downarrow$	—
$\varphi_r(d)$ (small d [b])	$\uparrow$	$\uparrow$	$\downarrow$
$\varphi_n(d)$ (small d)	—	—	$\downarrow$

[a] It holds for relative large τ. There could exist $\uparrow$ if τ is too small in extreme examples.

[b] There is little impacts on it for large d since it turns stable. It is the same for $\varphi_n(d)$.

on the dynamic performance, and increasing τ can lead to a better final performance but not the dynamic performance. So there exists a tradeoff for the choice of τ. As explained above, the influence of τ on the final performance is weakened when λp_r increases, and if τ is large, its impact on ϕ_r and ϕ_l is negligible for a large enough λp_r. Moreover, when ϕ_n becomes large that can provide enough replies candidates. So the better comprehensive option for performance improvement should be to enlarge λ, p_r as much as possible, and adjust τ according to the question urgency.

We perform some simulations over the three topics (i.e., **A**, **B**, and **C** in Table VII) for the proposed dynamic performance metrics with different settings for Popularity, Question Proneness and Reply Proneness. The results not only confirms performance impacts of λ, p_r and τ as summarized in Table VIII, but also illustrates how significant their impacts are. However, due to space limit, we are not able to include them into the paper.

VII. MORE DISCUSSIONS

Based on previous measurement and analysis, in this section, we conduct more discussions on possible ways to improve the performance of a web-QA system. More specifically, we seeks ways to improve the three system parameters in the way suggested by Table VIII. Of course, they are many other ways than the four we list here, but in general, any improvement should share the same thought of improving the three system parameters.

A. Active or Push-based Question Delivery

Our measurement and analysis reveals that the early response to one question is crucial to the service quality of the iAsk. The most unsatisfactory performance of the iAsk is actually the long delay for one to get an answer. One primary reason is that, in current web-QA design, the questions reach users passively: each user can update the question list only if he manually refreshes the webpage. But it is so troublesome and unrealistic for potential repliers to refresh or browse the webpage and find questions they can reply. So it not only takes a relative longer time for repliers to notice new questions but also discourages users to participate replying. So, we advocate that the question delivery mechanism should be changed to push the questions to users automatically. A handy solution is to use wide-accepted RSS technology [16]. With question push mechanism, we will not only be able to timely deliver questions, but also increase the user base because users will stick to their client (e.g., let the client run in background all the time) much longer than they stay on a certain webpage. This is actually learnt from our own experience with an RSS client.

To avoid users being overwhelmed and as we observed that users have narrow interest scope, we should narrow down the scope of an RSS channel. For example, we can use an RSS channel for a specific sub-category. In this way, users may be more willing to subscribe a few channels that they are mostly interested in or have expertise on. Furthermore, we can design some intelligent content filters (there are fruitful research results in the information retrieval field) by for example utilizing personal interest and expertise so that we can form dynamic personalized RSS feeds for users. In this way, questions will be more likely routed to appropriate users. This will also increase the reply probability.

B. Better Webpage Layout

We notice the large latency at iAsk is mainly induced by those questions in unpopular topics. However, in the current web-QA design, the question webpage is organized according to the pre-defined topic tree. A user usually needs to click multiple hyperlinks before he can locate the topic he is interested in. The webpage layout has great influence on τ. So one possible way to improve the whole performance by adjusting the topic tree or adding some topic shortcuts to the related

ones, e.g. hyperlinks to the unpopular topics, by learning user preferences and applying cookies. The other possibility is to simply make the interface page more searchable.

C. Better Incentive mechanism

As mentioned in Section IV, the existing incentive mechanism in iAsk does not really work out, except those questions with maximum score. Notice the big difference in performances for questions with 90 rewarding score and those with 100 (maximum) rewarding score, we believe the reason is not the extra 10 rewarding score, but rather, the system provide other access pages to list questions with rewarding scores and questions there are listed in a descending order of their rewarding scores. Naturally, all lower scored questions will be pushed down in the list and miss the opportunities of being catched by potential repliers. It is unfair for median-scored questions. Another observation is that most people do not adopt this incentive mechanism (73% questions are zero-scored). The reason is that it's free to ask question, i.e., no need to pay a minimum points to ask questions.

Although a better webpage layout scheme would alleviate the problem, we believe the web-QA system should come up with a better incentive mechanism to encourage users to pay more attention to median-scored questions. One easy way is to enforce that each question must have a certain rewarding score. In this way, when a user wants to ask questions, he has to find some other questions to earn some points and afford for his questions.

D. Utilizing Power of Social Networks

In one's daily life, people often ask friends when they have questions, and the friends may ask their friends if they can not answer themselves. It turns out that it is quite efficient if we propagate questions along the social networks. In fact, the small diameter and high clustering coefficient [17] ensures the efficiency. There are two ways to utilize the power of social networks: one is to build up user's social information into existing web-QA system. However, it will make the server bulky since it needs to maintain a lot of user status information and there will be privacy issues. The other more proper way is to integrate with instant messengers (e.g. MSN) and utilize the existing social networks there. This is actually the way we are working on.

VIII. Conclusions

Web-QA provides users an natural and effective channels for acquisition of specific information. Different from other QA systems, it explicitly leverage the grassroots' intelligence and collaboration. In this paper, we perform a first measurement study, to the best of knowledge, of a large scale web-QA system - iAsk. We investigated and reported various behavior patterns such as patterns over time, topic, users, incentive mechanisms, etc. We then proposed three performance metrics namely Reply-Rate, Reply-Number, and Reply-Latency, which are most closely related to the QoS of a web-QA system. From our measurement, we found that the overall performance of iAsk is mostly acceptable, but we did observe that it has large response delay (average 10 hours) and the correct answer rate (i.e., questioners are satisfactory with provided answers) is only about 80%. Other interest findings include that the existing incentive mechanism of iAsk does not work out.

Then, based on extensive measurement results, we proposed three mathematical models for the three performance metrics. We conduct evaluations of the three models using the measurement data and found they capture our observation precisely. The models revealed that the QoS of a web-QA system actually heavily depends on three key factors: the user scale, user reply probability and a system design artifact that is directly related to webpage design. We further studied their respective impacts on the system performance and proposed several possible ways through which current web-QA system can be improved.

References

[1] L. Stephen Coles, *An on-line question-answering systems with natural language and pictorial input*, Proceedings of the 23rd ACM national conference, Princeton, ACM, August 1968.

[2] Robert F. Simmons, *Answering english questions by computer: a survey*, Communications of the ACM, Vol. 8, No. 1, pp.:53-70, Jan. 1965.

[3] David Hawking, Ellen Voorhees, Peter Bailey, and Nick Craswell, *Overview of TREC-8 Question Answering Track*, Proceedings of the Eighth Text Retrieval Conference - TREC-8, Nov. 1999.

[4] Mark T. Maybury,*New Directions in Question Answering*, Papers from 2003 AAAI Spring Symposium, Stanford University, Stanford, CA, USA, AAAI Press, ISBN 0-262-63304-3, 2004.

[5] Cody C. T. Kwok, Oren Etzioni and Daniel S. Weld, *Scaling question answering to the Web*, Proceedings of the 10th international conference on World Wide Web, Hong Kong, Hong Kong, 2001, pp.150-161.

[6] Susan Dumais, Michele Banko, Eric Brill and et.al, *Web question answering: is more always better?*, Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, Tampere, Finland, 2002, pp. 291-298.

[7] MIT Start, *http://start.csail.mit.edu*.

[8] Zhiping Zheng, *AnswerBus Question Answering System*, Proceeding of HLT Human Language Technology Conference, San Diego, CA. March 24 - 27, 2002.*http://www.answerbus.com/index.shtml*.

[9] Google Answer, *http://answers.google.com/answers*.

[10] Wondir, *http://www.wondir.com/*.

[11] Naver, *http://www.naver.com*.

[12] iAsk, *http://iask.sina.com.cn*.

[13] Mehul Motani, Vikram Srinivasan and Pavan S. Nuggehalli, *PeopleNet: Engineering A Wireless Virtual Social Network*, Proceedings of the 11th ACM Annual International Conference on Mobile Computing and Networking, Cologne, Germany, Aug.28 - Sep. 2, 2005, ACM Mobicom, pp. 243-257.

[14] Jiwoon Jeon, W. Bruce Croft and Joon Ho Lee, *Finding similar questions in large question and answer archives*, Proceedings of the 14th ACM International conference on Information and Knowledge Management (CIKM), Bremen, Germany, Nov. 2005.

[15] Xiaoyong Liu and W. Bruce Croft and Matthew Koll, *Finding experts in community-based question-answering services*, Proceedings of the 14th ACM international conference on Information and Knowledge Management (CIKM), Bremen, Germany, Nov. 2005, pp. 315-316.

[16] http://www.webreference.com/authoring/languages/xml/rss/intro

[17] Duncan J. Watts, Peter Sheridan Dodds, and M. E. J. Newman, *Identity and search in social networks*, Science, Vol.296, May 2002, pp.1302-1305.