Relevance and Speed of Answers: How Can MAS Answering Systems Deal With That?

Albert Trias i Mansilla
Universitat de Girona, TECNIO - Centre
Easy, Agents Research Lab.
Campus de Montilivi
Edifici PIV, Lab 001
+00 34 972 41 84 78
albert.trias@udg.edu

ABSTRACT
Social search has enabled the automation of the village paradigm, allowing users to obtain answers easily from people. However, people can be overwhelmed answering always similar questions so that they can use agents to automate some part of the answering or search process. In this paper we explain the Question Waves, in which the first received answers are more likely to be relevant. Finally, we start the discussion whether agents can use answer time as heuristics for enhanced relevance of answers

Categories and Subject Descriptors

H.1. [**Information Systems**]: Models and principles

General Terms
Algorithms, Experimentation.

Keywords
Automation Social Search, Agents, Multi-agent systems, Question & Answering.

1. INTRODUCTION
Centralized search engines, in the category of the library paradigm [1], are designed to satisfy the needs of the most of users and have been progressively made more personalized and context aware. Although they generally provide good results, centralized search engines are less effective when dealing with atypical searches [2], and the relevance of their results decreases due to the effects of search engine optimization techniques (SEO)[1]. At the same time, public interest in social networking sites has grown. For example, in early 2011 Facebook had 600 million active users, according to its own website. Researchers [1,3,4,5,6] and companies are showing increasing interest in the "village paradigm" [1] or social search versus the "library paradigm". There are new examples of Q&A portals such as Aardvark and Quora, with stronger social or 2.0 features; there are new web browsers such as Rockmelt, which integrates Facebook with the browser, allowing users to ask questions of their online acquaintances or to share relevant information. The most popular online social network sites are interested in question answering, as demonstrated by Facebook's creation of Facebook Questions. Most of the new Q&A portals connect users with their acquaintances, who probably are more motivated to help the user than a complete stranger. In social search, the search problem is reduced to finding people who can cover the information needs. That is why the most important aspect of the village paradigm is finding people with expertise in the question's domain who are willing to answer the question.

In this paper, we will propose a model were users are represented by an agent per user in a p2p social network to automate knowledge exchanges; furthermore we propose a method to obtain the fastest answers that have high probability of being relevant.

This paper is organized as follows: Section 2 contains a brief overview about social search and agents, including few claims and their justification to support our work; then we explain an approach for automated Q&A agent social networks and some related work. In section 4, we explain the question waves model and in section 5 we show a set of simulations of the question waves behavior where the result obtained is that the fastest answers are more likely to be the best from the relevance point of view. Finally, in section 6, we start the discussion about whether agents can use or not answering time to consider answer relevance.

Cite as: Relevance and Speed of Answers: How Can MAS Answering Systems Deal With That. Albert Trias i Mansilla, Proc. of 13th European Agents Systems Summer School (EASSS 2011), July, 11–16, 2011, Girona, Spain, pp. XXX-XXX.

[1] *http://dashes.com/anil/2011/01/threes-a-trend-the-decline-of-google-search-quality.html*

2. SOCIAL SEARCH AND AGENTS
In this section, we will explain the background of our work and we will state the hypothesis our work is built from.

2.1 Social Search
Social Search is a type of search that uses social interactions, implicit or explicit, to obtain results. Chi [3] proposes the following classification for social search engines:

- Social Feedback Systems use social data to sort the results. This information can be obtained directly (ratings, tags or bookmarks) and indirectly (logs).
- Social Answering Systems are systems that use peoples' expertise or opinions to answer questions in a particular domain; the answerers can be friends, colleagues or strangers.

According to Chi's classification, Social Feedback Systems are unable to address new questions when the information is not available. Although Social Answering Systems solve this problem, the experts can be requested to answer a same question multiple times, and the stakeholders do not receive the content immediately.

Within Social Answering Systems, we consider knowledge exchange portals (Q&A). Knowledge and information exchanges in the form of questions and answers have emerged and grown along with the development of the Internet [7]. Likewise, 15% of the queries to the web search engines are completely formulated questions [8], despite the fact that a search using keywords does not always return relevant results.

We base our model in the two following claims:

> **C1: "**The village paradigm (social search) has some advantages in front of the library paradigm"

> **C2:** "Social search can be automated"

C1 is supported by the following statements. First, as Socrates argued in Plato's Phaedrus [9] people can answer new questions while, a text will just keep saying the same thing over and over again. Second, the popular wisdom also suggests that village paradigm has benefits in front of the library paradigm, one example is the Chinese proverb "a library of books does not equal one good teacher". Finally as Howe stated, people are often the best equipped to understand each other's questions and problems and to give an accurate answer [10].

C2 is supported by some examples. The most clear is collaborative filtering (CF) recommender systems. CF is a type of Social Feedback Systems that automatically uses the opinion of other uses to recommend items to a user. Another example is listing FAQs and their answers on web pages or text documents the oldest Q&A automation. FAQs were a popular way to provide answers to common questions [11]. Using FAQs as a way of automating Q&A is not simply uploading an FAQ page to a personal web page. Instead, FAQs as a way of automating of Q&A is like sharing one's knowledge sources, monitoring the questions coming in and being answered, proactively asking for Q&A updates, creating new pairs of Q&As, and much more. These actions are combined with privacy management, as not everything is meant to be public or indexable in its first instance.

However, the village paradigm suffers from a number of drawbacks [8]; the most important is the lack of answers that is caused by not having available the right people to provide the answers, or by having no knowledge available at all. People could be unavailable for several reasons: because they cannot be found, or they are not willing to answer, or they are not accessible due to their small answer bandwidth or the question did not get through to them. We think that some of these drawbacks can be addressed with the automation of the village paradigm.

2.2 Agents and Q&A Automation

In the AgentLink Roadmap [13], Luck et al. claim that agent technology can be considered from three perspectives: as a design metaphor, as a source of technologies or as a simulation. We consider agent technology as a design metaphor, where agents provide a way of structuring the application by means of autonomous and communicative entities. We also base our work on the following claim.

> **C3:** "Agents are a natural approach for social search automation".

Features of the intelligent agents that are relevant for social search are reactivity, sociability, proactivity, and autonomy. Agents' reactivity is defined as their reaction after receiving a question, that they will decide what actions to perform (ignore it, try to answer it, show it to its owner, forward it); agents' sociability is defined based on their capability of asking the question to other agents; agents' proactivity is defined based on their taking initiative without being explicitly asked to have their users maintain updated knowledge bases; and finally, agents' autonomy is defined as the ability of the agents to work independently of any other entity.

Furthermore, agents have enabled support representation, and coordination and cooperation among heterogeneous users and their processes. Internet and software agents enable the construction of information systems from multiple heterogeneous sources and contribute to improving the relationship between suppliers and consumers of knowledge to give the agents better control of the interactions [14]. These agents appear to offer the best approach for the automation of social networks for knowledge exchanges and are a good match for P2P systems.

Most of the works in the state of the art are generating agents with a top-down approach, for citing a few the works of Galitsky et al. [15], Yang [16] and Trojhan et al. [17]. A bottom-up approach is considerably more modular, scalable and synergic with the collective (2.0) approaches that are now at the core of Internet: let every agent be associated with a user to try to collect his/her knowledge and role within a number of communities.

3. AUTOMATED Q&A IN AGENTS SOCIAL NETWORK

In this section we will explain a possible way to automate social network, concretely automating Q&A in an agents social network. Furthermore, we include related work.

3.1 Approach

Our idea to automate a social search is based on using a p2p social network, where each user is represented by an agent, as it is represented in Figure 1. Each agent contains its user's personal FAQ list.

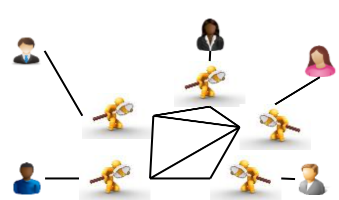

Figure 1: p2p agents social network, on each agent represents a user [2].

---

[2] *Icons by http://dryicons.com*

When a user has a question, she asks it to her agent. Her agent first will check the FAQ list and when there are not good answers on them, the agent can ask the question to a subset of its acquaintances. When an agent receives that question, first check the FAQ list and if the answer is unknown answer, the agent can proactively decide to ask its user to complete the question with further explanations and optionally it could add the question to its user's FAQ; when an agent receives a query that it cannot answer, and the question is not appropriate for its owner, the agent can forward the question to other agents.

With this approach, our attempt is to reuse previous pairs of questions and answers. In a prior study [18], 30% of the time that a query was performed, it had been carried out before by the same user, and 70% of the time it was searched before by an acquaintance of the user.

We proposed the ASKNEXT protocol [22], in which each user is represented by an agent who can forward the question, as in a BFS [2, 5, 21]. If the agent finds that a question is of interest to its user then the agent can show the question to her [6]. In our model, an agent can try to answer using the knowledge it has indexed, or, alternatively, can try to obtain new knowledge from its user, or, finally, can ask acquaintances. The novelty of this protocol is the usage of stop messages, which allow stopping the question-forwarding when any answer is found. In this protocol, for each question the agents can play three roles: the **questioner** is the agent who started the question; a **mediator** is an agent who receives a question and forwards it to another one; and an **answerer** is an agent who answers a question with its knowledge or with its user's knowledge.

3.2 Related Work

Sixearch (6S) [5] proposed to use distributed systems and social networks to address the problem of the web search. In their work, queries are sent through the social network, and the contacts that receive the query can forward the question or answer, as in the breadth first search (BFS) used by Gnutella protocol. Also these researchers use the TTL (Time to Live), which determines when a question can be forwarded. The answer, which consists of a set of bookmarks and documents, follows the inverse path of the question. Walter proposed a recommender system in which agents exploit the social network to obtain information filtered through trust based on a BFS [2], they compute the Trust Path in the graph to rank the answers. In the literature, there are discussions about how to compute the trust transitivity; its usage depends on the scenario and on whether trust can be used in recommender systems with a discounting [2].

MARS (Multi-Agent Referral) [6, 19, 20] is a P2P social network that uses agents to help their users obtain referrals to find experts who might answer the user's question. Agents determine to which contacts to send each question. Also, when an agent receives a question, the agent decides whether to show the question to its owner and whether to provide referrals to the questioning agent.

Michlmayr [21] proposed a model for query routing in P2P networks based on Ant Colony Optimization. In her proposal, an ant represents a query and when it reaches a peer with documents that satisfies the query, then a backward ant, which follows the inverse path, dropping pheromone, represents the answer message. In the case of the selection of the contact using pheromones, the most recent experiences are most important, as pheromones evaporate over time.

4. QUESTION WAVES

In this section we will explain question waves [23], an agent behavior for automated Q&A in agents social networks. In subsection 4.1 we will explain the assumptions that we consider; in section 4.2 we explain the question waves model that motivate us to considering answer speed as a possible heuristic for answer evaluation.

4.1 Assumptions

In this subsection we describe the assumptions of our model.

The first assumption is that agent's behavior has to be based on reciprocity. In [24], the authors explain that social exchange theory assumes that people try to have balanced relationships; people prefer relationships where they give and receive a similar amount of support. We think that agents need to have a give & take behavior. Reference [24] points out that if there is a discrepancy between giving and receiving, then the continuation of the relationship is threatened; furthermore, there are some cases in which there are unbalanced relationships, as can be the case of family or close friends. In this paper we will use the term reciprocity to refer to the effort that an agent makes on answering a question from an acquaintance.

The second assumption is that there are three factors that would affect the result of the task given to the agent. The first factor is the expertise of the answerer (E); the second factor is the answerer's implication (I) and the third factor consists of external factors (R). By implication we mean the amount of effort that an agent makes when performing a task; the agent can be motivated by rewards, by the task itself, or by who asks for the task. The external factors (R) can include considerations such as the personal situation of the answerer, whether she has free time, whether she is in a good or bad mood, whether she has work overload, whether she is tired, etc. As the result of a question will depend on the difficulty of the question, we model the result as a threshold θ for the implication, expertise, and other factors to be considered in relevant answers. The answer relevance is $\vartheta = f(I, E, R, \theta)$, where $\{\vartheta, \theta, I, E, R\} \in [0,1]$. We can express ϑ in logic as $\vartheta = I \wedge E \wedge R > \theta$ that can be implemented by applying a strong conjunction with Łukasiewicz (Eq. 2).

$$\vartheta = F_{\otimes}(I, E, R) = F_{\otimes}\big(F_{\otimes}(I, E), R\big) = max(0, max(0, I + E - 1) + R - 1) \qquad (2)$$

We think that an agent will put its best effort into helping close acquaintances; in the case of convenient acquaintances, the amount of effort that the agent will extend will depend on the benefit that the agent takes from the relationship while trying to maintain its balance. We will compute trust as the benefit that one agent a_1 takes from an agent a_2, meaning the relevance of the answers that a_1 received from a_2 for a_1's benefit.

A third assumption is that agent evaluation and trust is not contextual in our model, it is done to simplify the model allowing a better understanding of it.

Furthermore we assume a network of homogeneous agents, which are benevolent, collaborative, always online, and their answering time is constant.

4.2 Question Waves

Question waves is an agent behavior for automated Q&A in agent based social networks. This behavior is an attempt to offer the answers that are probably the best, first in time.

> **"T1:** Answers relevancy (in village paradigm) is correlated with answer time"

A question wave is an attempt to find an answer to a question. In every attempt, the same question is sent to a subset of acquaintances. The expectancy of finding appropriate answers decays after every attempt. The wave propagates through the network of agents, which amplify or attenuate it. The advantage of the question wave is that multiple agents are committed to finding the answers, with diversified options to find them, resulting in more relevant[3], faster[4], and robust[5] answers.

Question waves try to solve the following problem: in P2P, to request a question to all possible peers is not feasible because it can overload the system. However, reducing the number of recipients too far can provide the worst results. Furthermore, we believe that it is not feasible to request one agent after another when a response (with or without answers) is obtained, because this peer can be disconnected, can ignore the query, or can wait forever for the reply of an acquaintance. Deciding which peers have to receive requests is not any easy task. We think that the agents can decide to request this task to new recipients, as humans do, as a function of the current outcome for a task over time.

In each attempt, the question sender selects the most reliable acquaintances who have not been selected before for this question. Adding new recipients implies that the agent is not sure that it will receive any answer from the current recipients and tries to obtain an answer from less trusted recipients. Furthermore, this approach can be used when useful answers are received, enabling the user to read them in real time (when the search process is not complete) as a heuristic from the most to the least relevant.

This scheme can be implemented, dividing the possible recipients into different groups from most trusted to least trusted and programming the message sending, an agent could decide that a question is solved an no ask it to more agents.

The trust model that we use for this model will be based on the work of Walter [2] and Michlmayr [21]. In Walter the trust is updated with Eq. 1.

$$T_{a_i,a_j}(t+1) = \begin{cases} \gamma T_{a_i,a_j}(t) + (1-\gamma) r_k \, for \; r_k \geq 0 \\ (\gamma - 1) T_{a_i,a_j}(t) + \gamma r_k \, for \; r_k < 0 \end{cases} \quad (1)$$

Where:

- r_k is the experience that the agent a_i has made following the recommendation about object o_k transmitted by a_j.
- $T_{a_i,a_j}(t) \epsilon$ [-1,1] is the trust value of the path, and $T_{a_i,a_j}(0) = 0.5$.
- $\gamma \; \epsilon$ [0,1], indicates the dynamics of the trust. When $\gamma > 0.5$, trust is increased slowly but can decrease fast; it is usually a desired property.

Michlmayr model [21] can be similar to the model proposed by Walter with a $\rho = \gamma = 0.5$, where ρ determines the amount of pheromone evaporated; the difference is that in the model of Walter, the "evaporation" happens when there is a new evaluation, and the decision is done by the acquaintance instead of by the query.

We will updated trust values based on a hybrid between the models proposed by Walter [2] and Michlmayr [21]; the trust updated for each node is about its neighbor as in [21] instead of the path [2], but we used the equations from [2], with a γ=0.8 and we use 0.7 as threshold instead of 0.

5. EVALUATION

With the goal of evaluating T1 we made a set of simulations, that are explained in subsection 5.1, the results of these simulations are shown and discussed on subsection 5.2.

To evaluate if answer time can have correlation with answer relevance we will sort the answers received by different heuristics, and we will check their correlation with the sorting of the answers by relevance.

As answers are qualified by their relevance (ϑ) to evaluate results, we will use the Spearman and Kendall correlation[6] between the set of answers sorted by ϑ and the set of answers sorted by some heuristics explained below:

- Answer Distance (D): Closer answerers will be more motivated to help the questioner.
- Trust of the last sender (Tr): The important concern is the information source that we asked. From where that source takes the information does not matter; it is implicit on the trust of the information source.
- Receiving Order (H): All the contributors are searching first in the most relevant sources of information, so we expect that the best results are found first.
- Answer Distance and Trust (DT): Receiving Order and Trust (HT). As the distance evaluation and the order receiving can have several items with the same value, we decided to break the tie with the trust of the last sender.
- Transitive Trust (TT): We computed trust of the path TT_{a_0,a_n} as $\prod T_{a_i,a_{i+1}}$, where n indicates the size of the path and a_i the agent that will receive the request at distance i, and T_{a_a,a_b} is the trust from agent a_a to a_b.
- Trust of the Last Mediator (TLM): This heuristic means that the important is the trust that the last mediator has on the answerer. The agents should inform with the answer the trust that the last mediator has with the answerer, though it would not be desirable in some cases where privacy should be observed

[3] Relevant in the sense that answers come ranked by trust.

[4] Faster in the sense of reducing the burden of questions, and the agents are less overwhelmed.

[5] Robust in the sense of finding answers persistently.

[6] *We used the java library jsc (Java Statistical Classes) http://www.jsc.nildram.co.uk*

5.1 Simulations

Our simulations are based on the following assumptions:

- Agents are always online.
- All agents follow the question waves behavior.
- Agents share the same criteria evaluating answers.
- The time that an agent needs to generate an answer is constant and has no cost.
- The agents have not any limitation to any kind of messages.
- In each time stamp all agents are executed simultaneously.

Our simulations consist of a set of agents A={a_0 , a_1, …, a_i}, connected in a p2p social network, that, at each simulation step, perform the following algorithm:

Method Step
For each Received Answers
    If Own Question
        Update result and Trust
    Else
        Forward it and update trust
If I have a new Own question
    Select contacts in contact waves;
    Program messages
For each received question
    If I received the same question before
        Ignore it
    Else If I am good enough for answering,
        Generate Answer Value; Send answer
    Else
        Select contacts in contacts waves;
        Program messages
Send programmed messages

It is needed to consider, that a two questions are the same include that they have the same initiation and the same initial time stamp, this information has to be included in the question message.

One agent decides whether it has enough knowledge to answer a question in the function of the sender evaluation $Ev(a_i)$. As part of the reciprocity, the higher the evaluation an acquaintance has, the more the agent will try to give a better answer. The agent will answer if its $e_j > Ev(a_i) - \sigma$. We used $\sigma = 0.1$

In our simulations the implication of agents are computed by the evaluation of the requesting contact as an information source ($Ev(a_i)$), and the distance d from the questioner agent, because we believe that people use less effort helping friends of friends instead of direct friends. We computed ($Ev(a_i)$) as the mean of trust value for each domain (mean) or as the maximum trust value in any domain (max). The implication value is denoted by Eq. 3, where TTL_{MAX} is the maximum distance that a question can reach.

$$I = \frac{Ev(a_i)(TTL_{MAX} - d + 1)}{TTL_{MAX}} \quad (3)$$

In these simulations we compute ϑ with Eq. 4, to give the most importance to some variables over others. ϑ is the answer relevance, α is the weight of the Implication (I), β the weight of the expertise (E), δ the weight of the external factors (R).

$$\vartheta = \alpha I + \beta E + \delta R \quad (4)$$

We modeled the question waves consisting of 4 waves. The 1st wave arrives after 1 simulation step; the 2nd wave arrives after 5 simulation steps, the 3rd after 20 simulation steps and the last after 40 simulation steps. Agents are sorted into the different waves (1st, 2nd, 3rd and 4th) by the trust of the questioner. In our implementation, wave 1 goes in the 1-to-t_1 partition, while wave 2 goes in the t_1-to-t_2 partition, and so forth. The vector T={t_1,t_2,t_3}defines the trust partitions for the different waves. For experimentation, we used the following classifications: T = {[0.8,0.7,0.6], [0.85,0.8,0.7], [0.85,0.75,0.5], [0.85,0.7,0.5]}.

When a mediator receives an answer after it already has forwarded an answer, the mediator can use two strategies: the first strategy is to ignore the new answer as the task is complete for this question; the second strategy consists of forwarding the answer if the evaluation of the answer is better than the previous answer. In this case, the new answers are used to update the trust.

We modeled our simulations using some points in common with [20]:

- The expertise vector E of each agent has dimension 5. The value of $e_j \in [0,1]$ of an expertise vector E = {e_1, e_2, e_3, e_4, e_5} means the expertise level in domain j. E values are set randomly. Although that the model does not consider context, we tested it in a more realistic environment.
- Agent a_i will generate an answer from his expertise vector E when there is a good match between the query and its expertise vector.
- There is a Randomness factor R∈ [0,1] with weight δthat affects the answer relevance.
- The querying agents rate the services from P_i as ϑ'_i and $\vartheta'_i = \vartheta_i$.
- The queries correspond to vectors of length 5 that are 1 in one dimension and 0 in all other dimensions. For each query q, $\exists!\, i \mid q_i = 1$ and $\forall j, j \neq i \mid q_j = 0$.

Also we added the following points:

- At each step, each agent has a question probability of having its own question (Ψ). In our simulations, $\Psi = 0.05$, $\alpha = 0.2$, $\beta = 0.7$ and δ= 0.1
- The interest vector It={i_1, i_2, i_3, i_4, i_5} with the same dimension of vector E denotes the probability that a question will belong to domain j. $\sum_j i_j = 1$.
- The graph that represents the social network is undirected; agents have a mean of 20 acquaintances, all initial trust values are set to 0.75
- We repeated each configuration 10 times, setting as the random seed the values from 0 to 9, with 1,000 simulation steps and 200 agents

5.2 Simulation Results

Table 1 and Table 2 contains the results from the simulations, the first one contains Spearman correlations and the second one Kendall correlations. D and Tr have worst sorting. We suspect that D correlation is related with our implementation of Implication (equation 3) that is correlated with $\boldsymbol{\vartheta}$ (equation 4). Perhaps this poor correlation is due to having used random generation for the social networks in the experiments, where trust relations between peers were not built on the basis of their knowledge needs and, moreover, agents do not have the ability to start new relationships and to cut some of the current relationships. We believe that if agents have the chance of modifying their contact list regarding their needs, there will be an increase in the relevance of these heuristics.

About Tr heuristics, we believe that, at least with these input data, the trust on the answerer is more important than that on the mediator, aspect that is shown with the heuristics TLM and TT.

The heuristics H shows the best performance, specifically with T= [0.8, 0.7, 0.6]. The best answer relevance is obtained with Ev(a) max, because the implication factor will always be greater than with the mean. As in this paper we do not consider the cost of obtaining a good answer, overrating the agents will bring the best results.

Table 1. Simulation Results Spearman Correlation

Ev(a)	T	D	H	DT	HT	Tr	TT	TLM	ϑ
mean	.8,.7,.6	.14	.67	.17	.66	.14	.52	.9	.66
mean	.85,.8,.7	.10	.49	.16	.48	.17	.56	.91	.68
mean	.85,.75,.5	.11	.43	.16	.43	.19	.53	.91	.67
mean	.85,.7,.5	.12	.56	.16	.55	.16	.52	.9	.67
max	.8,.7,.6	.23	.7	.27	.69	.14	.53	.83	.72
max	.85,.8,.7	.13	.62	.2	.61	.2	.57	.87	.73
max	.85,.75,.5	.15	.6	.23	.59	.22	.58	.87	.72
max	.85,.7,.5	.19	.67	.24	.65	.16	.56	.85	.72

Table 2. Simulation Results Kendall Correlation

Ev(a)	T	D	H	DT	HT	Tr	TT	TLM	ϑ
mean	.8,.7,.6	.12	.51	.12	.49	.10	.38	.72	.66
mean	.85,.8,.7	.09	.37	.11	.34	.12	.41	.74	.68
mean	.85,.75,.5	.1	.32	.11	.30	.14	.39	.74	.67
mean	.85,.7,.5	.1	.42	.11	.39	.12	.38	.73	.67
max	.8,.7,.6	.2	.54	.19	.52	.11	.39	.64	.72
max	.85,.8,.7	.12	.47	.15	.44	.15	.41	.68	.73
max	.85,.75,.5	.13	.46	.16	.43	.16	.42	.69	.72
max	.85,.7,.5	.16	.52	.17	.48	.12	.41	.67	.72

We highlight that with our algorithm (H) the best answers are more likely to come up than the other answers. This is a very important claim, which is supported by the correlation of H to the relevance of answers, much higher than the trusted ranking of answers. We can see some relation with H and TT because both of them are based on trust transitivity, H shows better correlation than TT with T= [0.8, 0.7, 0.6].

The TLM shows the best correlation due that the most precise evaluation is the nearest of the answerer, we believe that it is due the homogeneous assumption and that implication has less weight than expertise in answer relevance, in a real situation, due that agents will have different points of view on answer relevance this value should decrease. This heuristic can be seen as a simplification of using reputation, if we use

If we consider reputation, were we get the means of all agents evaluations to each agent in a directory we get correlation values of 0.99 for Spearman's correlation and 0.91 for Kendall's correlation, we think that this value is that high due the same reasons than TLM.

We would like to highlight that best finding of question waves, is that agents and user can start to use the answers as they arrive without the need of resorting them, we believe that this approximation would reduce the amount of answers needed to find relevant answers.

6. DISCUSSION AND FUTURE WORK

Results in section 5 indicate that answer speed could be used as heuristics, or as a part of, for answer relevance. The idea that it is more likely that answers that arrive first are relevant, if this assumption is true, it will allow users checking results at real time, without wasting time.

The usage of the answer speed as a heuristic of answer relevance can have results similar if they were compared with trust transitivity, as all the agents ask the ones with higher trust, one of the advantages is that agents do not need to communicate their trust value to the information source, on the other hand, in our experiments all agents follow the behavior explained with Question Waves, but as autonomous entities they may decide to not use it.

Furthermore in our experiments, the agents do not need to spend time in computing the answers, sometimes adding effort to answer a question implies more time to compute it. In this section we will try to discuss the risks and advantages of using the answer time as answer relevance heuristic, with the idea of knowing if it can be used for agents in knowledge exchanges.

There are several points that can affect the answer velocity, we found the followings:

- Answering time.
- Communication time to find information source.
- Communication time to receive the answer.

Answering time is proportional to the effort put on answering. One example is numerical analysis where the error on the calculus is obtained computing more iterations. Furthermore the effort can be seen as the number of tasks the agent considers to do at a same time, which may decrease its performance (in time - relevance) for each individual task. On the other hand, answer time has relation also with expertise: an expert may need less time to answer a question that a non-expert, because she knows the answer or because she knows faster methods to obtain it. In the case of numerical analysis, not only the time or the number of iterations are important; selecting the best algorithm for the data, and the correct starting values can help to obtain a suitable answer in less time. However, in some cases, rather much expertise can do that the questioner does not understand the results as she does not understand the jargon or the expert considers that it is too obvious and the answer does not need more explanation. Also, computer's performance and computer's overload influence the answer speed.

Answering time also is affected by the information source state, as its information overload. Maybe an expert agent can be overloaded of work; as the agent is overloaded it can decide to spend less time answering each question, while a non-expert, may be less overloaded and can spent more time. Furthermore being overloaded can influence the time that the agent needs before considering to answer the question. Another thing that influences the answering time is the automation, we consider that, in most cases, human answers are better than automated ones; and usually humans needs more time to answering than computers; in section

2 we explained that humans sometimes understand the information need from other users better than a machine

The main lack of our approach is that considering the first answer received as the best, can be exploited by spammers, answering with a spam message without reading or trying to understand the question can bring fast answers. In the inverse case, considering slower answers as the best ones can be also manipulated: it only needs to wait before delivering the answer. We expect that these behaviors can be penalized with trust.

Another issue that influences our heuristics is that the time for receiving an answer does not mean that it is the time needed to compute it or the time that an agent waited to answer to a mediator. It might also happen that the answerer could consider the question only after sometime. Furthermore, some agents are not available 24/7, and this aspect does not make their answer worse.

We think that the findings of section 5 are interesting, it shows that answering time can be used as an heuristics to sort answers by relevance; but it is also needed to consider trust, and filtering the results, furthermore agents will need to try to discover why an answer has that specified in answering time. In section 5 it was only considered that the agents ask first the most trustable agents, without taking into account all the other factors, as information sources availability and different answering times.

7. ACKNOWLEDGMENTS

This research is funded by the EU project Num. 238887 (iSAC6+), the ACC1Ó Catalan Government grant ASKS - Agents for Social Knowledge Search, the IPT-430000-2010-13 project (SAKE), the Universitat de Girona research grant BR09/10 awarded to Albert Trias, and the AGAUR grant for the CSI-ref.2009SGR-1202.

8. REFERENCES

[1] Horowitz, D. and Kamvar, S. D. 2010. The anatomy of a large-scale social search engine, In: Proceedings of the 19th Intl. Conf. on World wide web, 431–440 (ACM, New York, 2010).

[2] Walter, F. E., Battiston, S. and F. Schweitzer, F. 2008. A model of a trust-based recommendation system on a social network, Autonomous Agents and Multi-Agent Systems 16(1)(2008) 57–74.

[3] Chi, E. H. 2009. Information seeking can be social, Computer 42 (3)(2009) 42–46.

[4] Hendler, J. Avoiding another AI winter, IEEE Intelligent Systems 23(2)(2008)2–4.

[5] Wu, L.S., Akavipat, R., Maguitman, A., and Menczer, F. 2007 Adaptive P2P social networks for distributed content based web search, in Social Inf. Retrieval Syst Emergent Tech. and Appl. for Searching the Web Effectively. IGI Global, 2007.

[6] Yu, B. and Singh, M. P. 2002. An agent-based approach to knowledge management, In: CIKM '02: Proceedings of the 11th intl. conf. on Information and knowledge management 642–644. (ACM, New York, 2002).

[7] Gosain, S. 2003 Issues in designing personal knowledge exchanges: First movers analyzed. IT & People, 16(3) 306–325, 2003.

[8] Liljenback, M. E. 2007 ContextQA: Experiments in interactive restricted domain question answering, Master's thesis, San DiegoUniversity, 2007.

[9] Ong. W. J. 1982. Orality and Literacy: The Technologizing of the Word, pp. 78-79.

[10] Howe, J. 2008. Crowdsourcing: Why the power of the crowd is driving the future of business, Crown Business, First Edition, 2008, ISBN: 978-0-307-39620-4

[11] de la Rosa, J. L., Rovira, M., Beer, M., Montaner, M., Givobic, D. 2010. Reducing the Administrative Burden by Online Information and Referral Services. In Citizens and E-Government: Evaluating Policy and Management. 2010.

[12] Malhotra, Y. 2002. Enabling knowledge exchanges for e-business communities. Information Strategy, 18(3), 26-31.

[13] Luck, M., McBurney, P., Shehory, O., Willmott S. 2005 Agent Technology: Computing as Interaction (A Roadmap for Agent Based Computing), AgentLink, 2005.

[14] Dignum, V. 2005 An overview of agents in knowledge management, In: Proceedings of the 16th Intl. Conf. on Applications of Declarative Programming and Knowledge Management 175–189, 2005.

[15] Galitsky, B. and Pampapathi, R. 2005. Can Many Agents Answer Questions Better than One? FirstMonday, 10(1)(2005)

[16] Yang, S.Y. 2009 Developing of an ontological interface agent with template-based linguistic processing technique for FAQ services, Expert Systems with Applications 36 (2009) 4049–4060

[17] Trojahn dos Santos, C., Quaresma, P., Rodrigues, I. and Vieira, R. 2006. A Multi-agent Approach to Question Answering, Lecture Notes in Computer Science, Vol 3960/2006, pp: 131-139

[18] Smyth, B., Balfe, E., Freyne, J., Briggs, P., Coyle, M. and Boydell, O. 2005. Exploiting query repetition and regularity in an adaptive community based web search engine, User Modeling and User-Adapted Interaction 14 (5)(2005) 383–423.

[19] Yu, B., and Singh, M. P. 2003. Searching social networks, In AAMAS '03: Proceedings of the 2nd Intl. joint Conf. on Autonomous agents and multiagent systems 65–72.

[20] Yu, B., Singh, M. P. and Sycara, K. 2004. Developing Trust in Large-Scale Peer-to-Peer Systems. First IEEE Symposium on Multi-Agent Security and Survivability. 1-10 (2004)

[21] Michlmayr, E., A. Pany, A., Kappel, G. 2007. Using Taxonomies for Content-based Routing with Ants, Journal of Computer Networks, 2007.

[22] Trias, A., de la Rosa, J. L., Galitsky, B. and Dobrocsi, G. 2010. Automation of social networks with Q&A agents (extended abstract) In Proceedings of the 9th Intl. Conf. on Autonomous Agents and Multiagent Systems, 1437-1438.

[23] Trias, A., de la Rosa, J. 2011. Propagation of Question Waves by Means of Trust in a Social Network. Proceedings of 9th International Conf. on Flexible Query Answering Systems.

[24] Ikkink, K. K. and van Tilburg, T. 1999. Broken ties: reciprocity and other factors affecting the termination of older adults' relationships. Social Networks 21 (1999) 131-146